big data and
cognitive computing

Semantic Web Technology and Recommender Systems

Edited by

Konstantinos Kotis and Dimitris Spiliotopoulos

Printed Edition of the Special Issue Published in
Big Data and Cognitive Computing

www.mdpi.com/journal/bdcc

MDPI

Semantic Web Technology and Recommender Systems

Semantic Web Technology and Recommender Systems

Editors

Konstantinos Kotis
Dimitris Spiliotopoulos

MDPI • Basel • Beijing • Wuhan • Barcelona • Belgrade • Manchester • Tokyo • Cluj • Tianjin

Editors

Konstantinos Kotis
Department of Cultural
Technology and
Communication, University
of the Aegean
Mytilene, Greece

Dimitris Spiliotopoulos
Department of Management
Science and Technology,
University of the Peloponnese
Tripoli, Greece

Editorial Office
MDPI
St. Alban-Anlage 66
4052 Basel, Switzerland

This is a reprint of articles from the Special Issue published online in the open access journal *Big Data and Cognitive Computing* (ISSN 2504-2289) (available at: https://www.mdpi.com/journal/BDCC/special_issues/semweb_tech).

For citation purposes, cite each article independently as indicated on the article page online and as indicated below:

LastName, A.A.; LastName, B.B.; LastName, C.C. Article Title. *Journal Name* **Year**, *Volume Number*, Page Range.

ISBN 978-3-0365-7210-9 (Hbk)
ISBN 978-3-0365-7211-6 (PDF)

Contents

About the Editors . . . . **vii**

Preface to "Semantic Web Technology and Recommender Systems" . . . . **ix**

Christos Nikas, Giorgos Kadilierakis, Pavlos Fafalios and Yannis Tzitzikas
Keyword Search over RDF: Is a Single Perspective Enough?
Reprinted from: *Big Data Cogn. Comput.* **2020**, *4*, 22, doi:10.3390/bdcc4030022 . . . . **1**

Christos Makris and Michael Angelos Simos
OTNEL: A Distributed Online Deep Learning Semantic Annotation Methodology
Reprinted from: *Big Data Cogn. Comput.* **2020**, *4*, 31, doi:10.3390/bdcc4040031 . . . . **21**

Dimitris Koryzis, Apostolos Dalas, Dimitris Spiliotopoulos and Fotios Fitsilis
ParlTech: Transformation Framework for the Digital Parliament
Reprinted from: *Big Data Cogn. Comput.* **2021**, *5*, 15, doi:10.3390/bdcc5010015 . . . . **39**

Vasilis Kopsachilis and Michail Vaitis
GeoLOD: A Spatial Linked Data Catalog and Recommender
Reprinted from: *Big Data Cogn. Comput.* **2021**, *5*, 17, doi:10.3390/bdcc5020017 . . . . **55**

Sotiris Angelis, Konstantinos Kotis and Dimitris Spiliotopoulos
Semantic Trajectory Analytics and Recommender Systems in Cultural Spaces
Reprinted from: *Big Data Cogn. Comput.* **2021**, *5*, 80, doi:10.3390/bdcc5040080 . . . . **81**

Muzamil Hussain Syed, Tran Quoc Bao Huy and Sun-Tae Chung
Context-Aware Explainable Recommendation Based on Domain Knowledge Graph
Reprinted from: *Big Data Cogn. Comput.* **2022**, *6*, 11, doi:10.3390/bdcc6010011 . . . . **107**

Maria Trigka, Andreas Kanavos, Elias Dritsas, Gerasimos Vonitsanos and Phivos Mylonas
The Predictive Power of a Twitter User's Profile on Cryptocurrency Popularity
Reprinted from: *Big Data Cogn. Comput.* **2022**, *6*, 59, doi:10.3390/bdcc6020059 . . . . **129**

Vassilis Poulopoulos and Manolis Wallace
Social Media Analytics as a Tool for Cultural Spaces—The Case of Twitter Trending Topics
Reprinted from: *Big Data Cogn. Comput.* **2022**, *6*, 63, doi:10.3390/bdcc6020063 . . . . **143**

Irene Tangkawarow, Riyanarto Sarno and Daniel Siahaan
ID2SBVR: A Method for Extracting Business Vocabulary and Rules from an Informal Document
Reprinted from: *Big Data Cogn. Comput.* **2022**, *6*, 119, doi:10.3390/bdcc6040119 . . . . **161**

Dimos Ntioudis, Panagiota Masa, Anastasios Karakostas, Georgios Meditskos, Stefanos Vrochidis and Ioannis Kompatsiaris
Ontology-Based Personalized Job Recommendation Framework for Migrants and Refugees
Reprinted from: *Big Data Cogn. Comput.* **2022**, *6*, 120, doi:10.3390/bdcc6040120 . . . . **185**

Alireza Javadian Sabet, Mahsa Shekari, Chaofeng Guan, Matteo Rossi, Fabio Schreiber, and Letizia Tanca
THOR: A Hybrid Recommender System for the Personalized Travel Experience
Reprinted from: *Big Data Cogn. Comput.* **2022**, *6*, 131, doi:10.3390/bdcc6040131 . . . . **209**

Reem Qadan Al-Fayez, Marwan Al-Tawil, Bilal Abu-Salih and Zaid Eyadat
GTDOnto: An Ontology for Organizing and Modeling Knowledge about Global Terrorism
Reprinted from: *Big Data Cogn. Comput.* **2023**, *7*, 24, doi:10.3390/bdcc7010024 . . . . **229**

Mohd Anuaruddin Bin Ahmadon, Shingo Yamaguchi, Abd Kadir Mahamad and Sharifah Saon
Refining Preference-Based Recommendation with Associative Rules and Process Mining Using Correlation Distance
Reprinted from: *Big Data Cogn. Comput.* **2023**, *7*, 34, doi:10.3390/bdcc7010034 . . . . **251**

About the Editors

Konstantinos Kotis

Konstantinos Kotis (https://orcid.org/0000-0001-7838-9691) is currently a tenure track assistant professor at the University of the Aegean, Dept. of Cultural Informatics and Communication, i-Lab, and a research associate at the University of Piraeus, Dept. of Digital Systems, AI Lab. His research interests include: Knowledge/Ontology Engineering, Semantic Web technologies, Semantic Data Management, the Semantic Web of Things, and KG-based conversational AI (chatbots). He has published more than 90 papers in peer reviewed international journals and conferences (Google Scholar h-index 18; citations > 1600) and served as reviewer and PC member in several journals and conference events. He has also contributed to several national and European projects from different roles/positions. For more information, please visit: http://i-lab.aegean.gr/kotis.

Dimitris Spiliotopoulos

Dimitris Spiliotopoulos (https://orcid.org/0000-0003-3646-1362) is an assistant professor at the Department of Management Science and Technology, University of the Peloponnese, Greece. His research interests include HCI, data analysis, UX, recommender systems, design thinking, and interaction design. He has been involved in research projects in intelligent communication, cybersecurity, human factor design, and data analysis for more than 10 years. He has more than 100 publications in peer reviewed international journals and conference proceedings.

Preface to ”Semantic Web Technology and Recommender Systems”

Semantic Web (SW) technologies define and analyze Web data linked or not to enable semantic interconnection. Related SW technologies allow data analysts, application designers, and cross-domain experts (linguists, cognitive scientists, machine learning experts and, user interface designers) to uncover, model, and eventually, utilize data semantics to build and work on approaches and ideas that require a deep (human and machine) understanding of the data. Semantics- and data-driven methods in computational intelligence (CI), and especially in recommender systems (RS), analyze single-source big data to identify and select recommendable content for users and applications. Multi-source multi-modal and heterogeneous data are, however, a larger challenge that SW and CI technologies need to face. Such data are of immense value for understanding the users’ expectations and redefining the goals for content recommendation. The challenge is that to semantically integrate and combine data from disparate and heterogeneous sources, and for an undefined or unknown original target, they must go through a layer of data understanding, i.e., a semantic data management layer. Advanced semantic data management and knowledge graphs (KG) are potential means of achieving the interlinking of data from original, social, cognitive, and world sources.

In this book (Volume I), 13 papers have been published on different topics of the wide research areas of Semantic Web and Recommender systems. These papers have been carefully selected based on the peer review of several respectful reviewers organized by MDPI’s *BDCC* journal. This issue has attracted well-known international research teams, who we would like to thank for their work.

Konstantinos Kotis and Dimitris Spiliotopoulos
Editors

big data and cognitive computing

MDPI

Article

Keyword Search over RDF: Is a Single Perspective Enough?

Christos Nikas [1,2], Giorgos Kadilierakis [1,2], Pavlos Fafalios [1] and Yannis Tzitzikas [1,2,*]

[1] Information Systems Laboratory, FORTH-ICS, 70013 Heraklion, Greece; cnikas@ics.forth.gr (C.N.); kadilier@csd.uoc.gr (G.K.); fafalios@ics.forth.gr (P.F.)

[2] Computer Science Department, University of Crete, 70013 Heraklion, Greece

* Correspondence: tzitzik@ics.forth.gr

Received: 6 August 2020; Accepted: 24 August 2020; Published: 27 August 2020

Abstract: Since the task of accessing RDF datasets through structured query languages like SPARQL is rather demanding for ordinary users, there are various approaches that attempt to exploit the simpler and widely used *keyword-based search paradigm*. However this task is challenging since there is no clear unit of retrieval and presentation, the user information needs are in most cases not clearly formulated, the underlying RDF datasets are in most cases incomplete, and there is not a single presentation method appropriate for all kinds of information needs. As a means to alleviate these problems, in this paper we investigate an interaction approach that offers multiple presentation methods of the search results (*multiple-perspectives*), allowing the user to easily switch between these perspectives and thus exploit the added value that each such perspective offers. We focus on a set of *fundamental perspectives*, we discuss the benefits from each one, we compare this approach with related existing systems and report the results of a task-based evaluation with users. The key finding of the task-based evaluation is that users not familiar with RDF (a) managed to complete the information-seeking tasks (with performance very close to that of the experienced users), and (b) they rated positively the approach.

Keywords: keyword search; RDF; interactive information retrieval

1. Introduction

The Web of Data contains thousands of RDF datasets available online (see [1] for a recent survey), including cross-domain knowledge bases (KBs) (e.g., DBpedia and Wikidata), domain specific repositories (e.g., DrugBank [2], ORKG [3], and recently COVID-19 related datasets [4]), as well as Markup data through `schema.org`. These datasets are queried mainly through structured query languages, i.e., SPARQL. Faceted Search is a user-friendlier paradigm for interactive query formulation and exploratory search, however, the systems that support it (see [5] for a survey) also need a keyword search engine as a flexible entry point to the information space. Consequently, and since plain users are acquainted with web search engines, an effective method for keyword search over RDF is indispensable. Moreover, keyword search allows for multiple-word (even paragraph-long) queries that can address many topics, and such information needs could be difficult to formulate even in structured query languages. The results of such queries allow users to detect associations of entities that they were not aware of, thus favoring the discovery of new information.

In general, we could say that *structured queries* (e.g., using SPARQL) and *unstructured queries* (keyword search) are fundamental components of all access methods over RDF. Figure 1 shows the general picture of access services over RDF. Apart from *Structured Query Languages* and *Keyword Search* we can see the category *Interactive Information Access*. That refers to access methods that are beyond the simple "query-and-response" interaction, i.e., methods that offer more interaction options to the user and exploit also the *interaction session*. In this category, we have methods for browsing, methods for faceted search [5], methods for formulating OLAP queries (e.g., [6]), and methods for assistive query building (e.g., [7]). Finally, in the category *natural language interfaces* we have methods for question answering, dialogue systems, and conversational interfaces. As the figure shows, both *interactive information access* and *natural language interfaces* pre-suppose effective and efficient support of structured and unstructured queries.

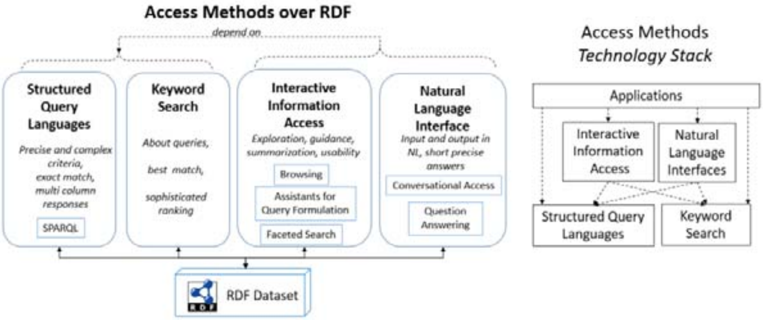

Access Methods over RDF
depend on
Structured Query Languages
Precise and complex criteria, exact match, multi column responses
SPARQL
Keyword Search
About queries, best match, sophisticated ranking
Interactive Information Access
Exploration, guidance, summarization, usability
Browsing
Assistants for Query Formulation
Faceted Search
Natural Language Interface
Input and output in NL, short precise answers
Conversational Access
Question Answering
RDF Dataset
Access Methods Technology Stack
Applications
Interactive Information Access
Natural Language Interfaces
Structured Query Languages
Keyword Search

Figure 1. Access Methods over RDF.

However, keyword search over RDF datasets is a challenging task since (a) in RDF there is no clear unit of retrieval and presentation, (b) it is difficult to understand, from a usually small keyword query, the intent of the user, (c) the data are in most cases incomplete (making the provision of effective retrieval difficult), and (d) there is not a single presentation method appropriate for all kinds of information needs.

To tackle these challenges, in this paper we focus on the value stemming from offering *multiple-perspectives* of the search results, i.e., multiple presentation methods, each presented as a separate *tab*, and allowing the user to easily *switch* between these perspectives, and thus exploit the added value that each such perspective offers. To grasp the idea, Figure 2 shows the search results for the query "El Greco museum", as presented in each of the five currently supported tabs.

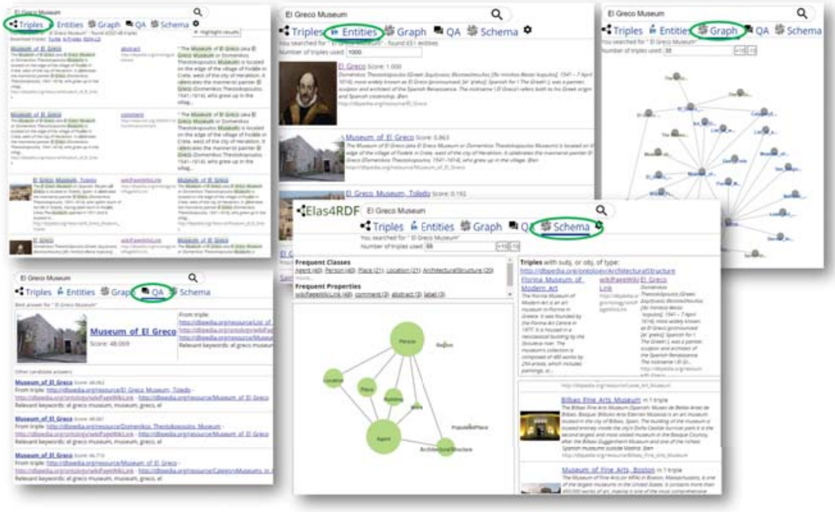


Figure 2. Search results for the query "El Greco museum".

As basic keyword-search retrieval method, we assume the *triple-centered* approach proposed in [8] (which in turn relies on `Elasticsearch`) because it is schema-agnostic (and thus general-purpose), and it offers efficient and scalable retrieval services with effectiveness comparable (as evaluated using DBpedia-Entity test collection for entity search [9]) to the effectiveness of dedicated systems for keyword search over RDF (more in [8]). Over this basic service, in this paper we motivate the provision of certain *fundamental perspectives*, we showcase the benefits from each one, and we evaluate what users can achieve if they have all of them at their disposal.

In comparison to previous works (the demo paper [10]), in this paper: we motivate the multi-perspective approach, we discuss the added value of each perspective, we introduce additional perspectives, we compare the functionality of the implemented system with other systems over DBpedia, and mainly we report the results of a *task-based evaluation with users* that provides interesting insights related to the validation of the main research hypothesis of this paper, i.e., whether the provision of more than one tab is helpful for the users. The key finding is that the success rate of all users was very high even of those users not familiar with RDF.

The rest of this paper is organized as follows: Section 2 discusses the related work, Section 3 provides the motivation for the multi-perspective approach and describes its architecture, while Section 4 describes each individual perspective and the tab-switching interaction approach. Section 5 compares the proposed approach with related (and comparable) systems and presents the results of a task-based evaluation with users. Finally, Section 6 concludes the paper and identifies issues for further research.

2. Related Work

At first we provide some background about RDF (in Section 2.1), then we discuss the existing approaches for keyword search over RDF (in Section 2.2), and finally, we discuss the visualization of RDF search results (in Section 2.3).

2.1. Background: RDF

RDF stands for *Resource Description Framework* and it is a framework for describing resources on the web. Essentially it is a structurally object-oriented model. RDF uses Uniform Resource Identifiers (URIs), or anonymous nodes, to denote resources, and literals to denote constants. Every statement in RDF can be represented as a *triple*. A triple is a statement of the form subject-predicate-object $\langle s, p, o \rangle$, and it is any element of $T = (U \cup B) \times (U) \times (U \cup B \cup L)$, where U, B and L are the sets of URIs, blank nodes and literals, respectively. Any finite subset of T corresponds to an RDF graph (or dataset). We can divide the URIs in three disjoint sets: entities (e.g., http://dbpedia.org/resource/Aristotle), properties (e.g., http://dbpedia.org/property/dateOfBirth) and RDF classes (e.g., http://dbpedia.org/ontology/Philosopher).

2.2. Keyword Search over RDF Datasets

Keyword search over RDF data can be supported either by translating keyword queries to structured (SPARQL) queries (like in [11,12]), or by building or adapting a dedicated information retrieval system using classical IR methods for indexing and retrieval. This paper builds upon approaches that follow the second direction. In general, systems of that kind construct the required indexing structures either from scratch or by employing existing IR engines (like Lucene and Solr), adapt the notion of a virtual document for the RDF data, and rank the results (entities, triples or subgraphs) according to commonly used IR ranking functions. There are various systems that fall in this category, like [13–15]. Most such systems rely on adaptations of the TD-IDF weighting, as in [16] where the keyword query is translated to a logical expression that returns the ids of the matching entities. Another direction is to return ranked subgraphs instead of relevant entity URIs, like in [17], while in [18] the returned subgraphs are computed using statistical language models.

Ranking is usually based on extensions of the BM25 model, e.g., in [19,20]. The work in [21] introduced the TSA+VDP keyword search system, which first builds offline an index of documents over a set of subgraphs via a breadth-first search method, and at query-time, it returns a ranked list of these documents based on a BM25 model. Regarding the retrieval unit, most works return either URIs or subgraphs, except [8,22] that follow a triple-centered approach.

With respect to works that rely on document-centric information retrieval systems, LOTUS [22] makes use of `Elasticsearch` and provides a keyword-search entry point to the Linked Data cloud, focusing on issues of scalability. `Elasticsearch` has been also used for indexing and querying Linked Bibliographic Data in JSON-LD format [23]. Finally, Kadilierakis et al. [8] adapts `Elasticsearch` for supporting keyword search over arbitrary RDF datasets. Through an extensive evaluation, the authors studied questions related to the selection of the triple data to index, the weighting of the indexed fields, and the structuring and ranking of the retrieved results. In our work, we make use of the approach proposed in [8] because it is schema-agnostic and returns ranked lists of triples, which offers us the flexibility to provide different visualizations of the search results.

2.3. Visualization of RDF Search Results

There are several approaches for browsing, exploring and visualizing RDF datasets in general, e.g., see the surveys [24,25]. Regarding the visualization of SPARQL results, there are a few works, however, since the form of the results of such queries is essentially that of a relational table, these approaches provide amenities for the visualization of tabular data, i.e., various plots and charts for analytics [26–28].

As regards the visualization of keyword search results over RDF, which is the main focus of our work, DBpedia Precision Search & Find (http://dbpedia.org/fct/) returns entities and for each one it shows its URI, its title, the URI of the named graph it belongs to, as well as a description with highlighted the query terms. Also, the user can browse on the Linked Data by clicking on the shown resources. The keyword search systems LOTUS [8,22,29] do not focus on presentation and visualization. LOTUS returns triples by providing the full URIs of the resources, while [8] returns triples and/or entities using an API. In general, most works (including [30,31]) do not pay attention to the presentation of results; they focus on the ranking of entities/subgraphs that they compute.

Finally, Stab et al. [32] and Kontiza et al. [33] the exploitation of semantics in the visualization of search results. The work in [32] uses visualization techniques for offering visual feedback about the reasons a set of search results was retrieved and ranked as relevant. In [33] the authors performed an analytical inspection and a user study of the interface offered by two semantic search engines: *Kngine* and *Sig.ma* (both are not active anymore). In particular, the authors investigated if the exploitation of semantics enables a better visualization of search results and thus a better user experience.

To our knowledge, our work is the first that investigates and evaluates (with real users) a multi-perspective interactive approach to present the search results of a keyword search system over RDF.

3. Multi-Perspective Presentation of Search Results: Rationale and Architecture

3.1. Rationale

The rationale for the multi-perspective (and tabs-switching interaction) approach that we propose can be summarized as:

- *No Clear Unit of Retrieval and Presentation.* In RDF data, there is not the notion of document or web page as is the case in web searching. Therefore, the retrieval, presentation and visualization of RDF data is challenging due to the complex, interlinked, and multi-dimensional nature of this data type [25].

- *No Clear Information Need.* The user query is just an attempt to formulate his/her information need. Some user needs require a single fact, others a list of entities or a set of facts, other how a set of entities are connected, other have an exploratory nature, and so on.
- *Incomplete Data.* The underlying dataset is in most cases incomplete [34] (also demonstrated by the number of papers that aim at completing the missing data [35]), therefore the retrieved triples cannot be considered neither complete, nor appropriately ranked. However, the provision of more than one method, each consuming different proportions of the list of top hits (and of their context), increases the probability that one method achieves to return something that is useful for the user's information need.
- *There is not a single presentation method appropriate for all kinds of information needs.* An established method on how to present RDF results for arbitrary query types does not exist yet, and it seems that a single approach cannot suit all possible requirements. Different kinds of information needs need different ways to present the results.

For the above reason we propose a *multi-perspective* approach, where each perspective is presented in a different *tab*, stressing a different aspect (and proportion) of the hits. The user can inspect all tabs and get a better overview and understanding of the search results. The *tabs-switching interaction* that we propose is easy to understand and perform by the user, just like plain Web search engines offer various such tabs (for images, videos, news, etc.). Below, in Section 4, we shall discuss the rationale (added value) of each particular tab and how it is defined. An orthogonal but important challenge is how to provide several such presentation methods at real time, for enabling the user to switch fast between the different perspectives, i.e., the multi-perspective and tab-switching approach should not add a noticeable latency to the responses.

3.2. Architecture

As keyword search service we adopt the approach proposed in [8] because it is schema-agnostic, directly applicable, has good evaluation results, and its triple-centered approach facilitates the multi-perspective approach. Specifically, we exploit the REST API that is offered by that service which accepts keyword queries and returns results in JSON format (code available at https://github.com/SemanticAccessAndRetrieval/Elas4RDF-search). On top of this search service we build the multi-perspective approach.

The full DBpedia 2015-10 dataset has been indexed using 2 approaches (i.e., *baseline* and *extended*, described in [8]). We have used that version of DBpedia because it is the version used in the DBpedia-Entity test collection for entity search [9], which allowed us to get comparable results related the effectiveness of the approach (as detailed in [8]). The number of virtual documents (triples) in both cases is 395,569,688. In our setup and experiments, the average query execution time is around 0.7 s for the baseline method and 1.6 s for the extended, and depends on the query type.

4. The Fundamental Perspectives of Keywords Search Results

Below we describe each individual perspective (for short *tab*) and then (in Section 4.6) we discuss the role of each in tab in the general search process. In the description of each perspective we consider the DBpedia 2015-10 dataset and the query q_{run} = "El Greco museum" as our running example.

4.1. Triples Tab

Rationale: This tab is generally the most useful one since the user can inspect all components of each triple, and understand the reason why that triple is returned. The addition of images help the user to easily understand which triples involve the same entities.
Description: A ranked list of triples is displayed to the user (as fetched from the search service described in Section 3.2), where each triple is shown in a separate row. For visualizing a triple, we create a *snippet* for each triple element (subject, predicate, object). The snippet is composed of:

(i) a title (the text indexed by the baseline method), (ii) a description (the text indexed by the extended index; if any), and (iii) the URI of the resource (if the element is a resource). If the triple element is a resource, its title is displayed as a hyperlink, allowing the user to further explore it. We also retrieve and show an image of this resource (if any). For the query q_{run} = "El Greco museum", more than 4.2 million triples are retrieved. The first two triples are about the Museum of El Greco in Crete, the third about the El Greco Museum in Toledo, the fourth about the entity El Greco, the fifth is a triple about a list of works by El Greco, and so on.

4.2. Entities Tab

Rationale: If the user is interested in entities, and not in particular facts, this view provides the main entities.
Description: Here the retrieved triples are *grouped* based on entities (subject and object URIs), and the entities are *ranked* following the approach described in [8] which considers the weighted gain factor of the ranking order of the triples in which the entities appear. Then, a ranked list of entities is displayed to the user, where each entity is shown in a different row. For visualizing an entity, we create the same snippet like previously. The title is displayed as a hyperlink, since the entities are resources, allowing the user to further explore the entity. For q_{run} the returned entities include "El Greco", the two museums of El Greco (in Crete and Toledo), particular paintings, like "Saint Peter and Saint Paul", the music album "El Greco" by Vangelis, the film "El Greco (2007)", and so on.

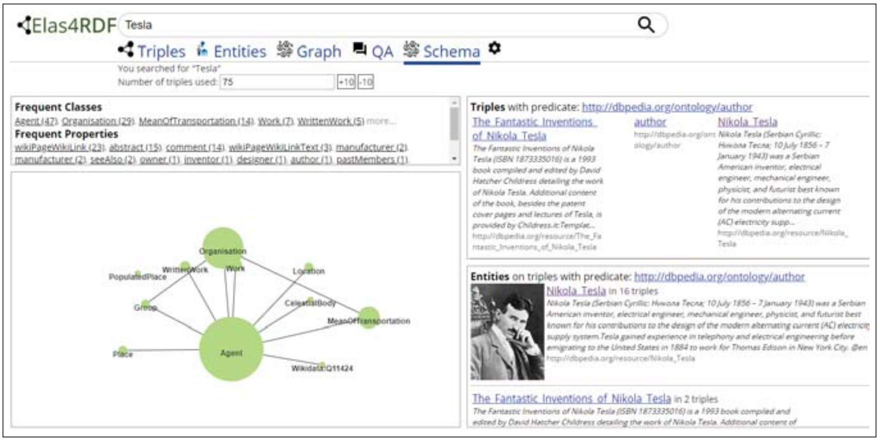
4.3. Graph Tab

Rationale: This tab allows the user to inspect a larger number of triples without having to scroll down. Most importantly, this view reveals the grouping of triples, how they are connected, and whether there is one or more poles and interesting connections.
Description: The retrieved triples are visualized as a graph for stressing how the triples are connected. By default, the graph shows the top-15 triples; however, the user can increase or decrease this number, while the nodes are clickable, pointing to the corresponding resource in DBpedia. In our implementation we use JavaScript InfoVis Toolkit (https://philogb.github.io/jit/). For q_{run} the user can see how the top ranked triples are connected and can easily spot the nodes that have high connectivity.

4.4. Schema Tab

Rationale: The objective is to show which are the more frequent schema elements of the retrieved triples. This is useful for (a) understanding the conceptual context of the hits, (b) for exploring (restricting) interactively the triples or entities of the answer (by filtering with respect to class or property), and (c) for helping an experienced user to inspect which classes and properties occur in the answer, if after the keyword search, the user would like to formulate a SPARQL query (directly or through a faceted search system, or through a query builder in general like [7,36]).
Description: The schema tab is divided in four frames as shown in Figure 3.

Upper Left Frame: It shows the more frequent classes and properties, accompanied by their frequency. Let A be the top-K triples retrieved for the current query, P the properties in A, i.e., $P = \{p \mid (s, p, u) \in A\}$, and C the classes of the URIs in the triples of A, i.e., $C = \{\, c \mid (s, rdf : type, c), s \in SP\}$. For each $c \in C$, its frequency is defined as $freq(c) = |\{\, o \in SP \mid (o, rdf : type, c) \in KB\}|$, while for each $p \in P$, $freq(p) = |\{(s, p, o) \in A\}|$. Through a parameter F we control the number of visible elements, i.e., initially the user can see only the F in number elements of C with the highest frequency, and the F in number elements of P with the highest frequency (however, the user can expand the visible elements to see all of them). By clicking a class or a property the user can see the corresponding triples and entities in the frames at the right side that will be described later.



Tesla

You searched for "Tesla"
Number of triples used: 75

Frequent Classes
Agent (47), Organisation (29), MeanOfTransportation (14), Work (7), WrittenWork (5) more...

Frequent Properties
wikiPageWikiLink (23), abstract (15), comment (14), wikiPageWikiLinkText (3), manufacturer (2), manufacturer (2), seeAlso (2), owner (1), inventor (1), designer (1), author (1), pastMembers (1)

Triples with predicate: http://dbpedia.org/ontology/author

The Fantastic Inventions of Nikola Tesla *The Fantastic Inventions of Nikola Tesla (ISBN 1873335016) is a 1993 book compiled and edited by David Hatcher Childress detailing the work of Nikola Tesla. Additional content of the book, besides the patent cover pages and lectures of Tesla, is provided by Childress.it:Templat...* http://dbpedia.org/resource/The_Fantastic_Inventions_of_Nikola_Tesla	author http://dbpedia.org/ontology/author	Nikola Tesla *Nikola Tesla (Serbian Cyrillic: Никола Тесла; 10 July 1856 – 7 January 1943) was a Serbian American inventor, electrical engineer, mechanical engineer, physicist, and futurist best known for his contributions to the design of the modern alternating current (AC) electricity supp...* http://dbpedia.org/resource/Nikola_Tesla

Entities on triples with predicate: http://dbpedia.org/ontology/author

Nikola Tesla in 16 triples
Nikola Tesla (Serbian Cyrillic: Никола Тесла; 10 July 1856 – 7 January 1943) was a Serbian American inventor, electrical engineer, mechanical engineer, physicist, and futurist best known for his contributions to the design of the modern alternating current (AC) electricity supply system.Tesla gained experience in telephony and electrical engineering before emigrating to the United States in 1884 to work for Thomas Edison in New York City. @en
http://dbpedia.org/resource/Nikola_Tesla

The Fantastic Inventions of Nikola Tesla in 2 triples
The Fantastic Inventions of Nikola Tesla (ISBN 1873335016) is a 1993 book compiled and edited by David Hatcher Childress detailing the work of Nikola Tesla. Additional content of

Figure 3. The Schema Tab (Tesla).

Bottom Left Frame: It shows graphically the more frequent classes and properties. A parameter K (just like in the graph tab) controls the number of triples that feed the schema tab (the user can increase decrease it as she wishes to). In particular, the graph $\Gamma = (Nodes, Edges)$ that is visualized is defined as $Nodes = C$, and $Edges = \{(c, c') \in C \times C' \mid (s, p, o), (s, rdf:type, c), (o, rdf:type, c') \in A\}$, i.e., an edge connects two classes c and c' if there is at least one triple in A that connects an instance of c with one instance of c'. Ideally the graph visualization should make evident the frequencies, i.e., the more frequent classes and properties should be visualized with bigger boxes and arrows. It is not hard to see that the number of edges, i.e., $|E|$, can be higher than the number of distinct properties that occur in A, e.g., if $(s, p, o) \in A$ and s is classified to two classes $c1$ and $c2$, and o to two classes $c3$ and $c4$, then the graph will contain the four edges $\{(n(c1), n(c3)), (n(c1), n(c4)), (n(c2), n(c3))(n(c2), n(c4))\}$. The reverse is also possible, i.e., $|E|$ can be less than the number of distinct properties, e.g., if $(s, p1, o)$ and $(s, p2, o)$ belong to A, and each of s and o is classified to one class, then only one edge will be visible between these two classes. Please note that several variations and extensions are possible from the area of semantic model visualization and summarization.

Right Upper and Right Bottom Frames: These frames show the *triples* and *entities*, related with the user's click. Suppose the user has clicked on a frequent class "`c1(18)`". The triples frame will show all triples $\{(s, p, o) \in A \mid (s, rdf:type, c1) \wedge (o, rdf:type, c1)\}$, and let call this set T. The entities frame will show the more frequent entities that occur in T. If the user clicks on a frequent property "`p2(10)`", the triple frame will show the 10 triples A that have $p2$ as property, let call this set T, and the entity frame will show the more frequent entities of those occurring in T. The above behavior is supported also by the graph, i.e., clicking on a node is interpreted as if the user had clicked on the corresponding frequent class.

Returning to q_{run}, we can see the classes `Person`, `Agent`, `Location`, `Work`, etc. and various properties. The right frames show the triples and entities after having clicked on "`Architectural Structure`", i.e. triples and entities that are related to the query *and* classified under the class "`Architectural Structure`" (we can see information about a museum in Florina, another in Bilbao, etc.).

As another example, for the query "Tesla", the user is getting what is shown in Figure 3, enabling him to focus on the desired triples or entities, i.e., to those related to: Tesla Motors (Organization), Nicola Tesla (Agent), Tesla Model X (Mean of Transportation), Tesla West Virginia (Place). By increasing the number of triples he can also find Tesla Band (Group). By clicking on the property "author" the user can directly see the triple related to works authored by Nicola Tesla. In general, in this tab the user can increase a lot the number of consumed triples: although more classes and properties will appear their number is not high, hence in most cases they will not clutter the diagram (in the example of Figure 3 the schema tab consumes 75 triples).

4.5. Question Answering (QA) Tab

Rationale: Here we attempt to interpret the user's query as a *question* and try to provide *a single compact answer*. The challenge is to retrieve the most relevant triple(s) and then extract natural language answers from them.
Description: QA over structured data is a challenging problem in general (e.g., see [37] for a recent survey), and any QA over KB approach could be applied in this tab. In our current implementation, we only support questions that can be answered by a single triple. We extract a set of terms from the question by applying lemmatization and expansion to multi-word expressions. Then we attempt to retrieve triples where two components (subject, predicate, or object) are similar to terms extracted from the question. To do that, we use `Elasticsearch`'s query Domain Specific Language to search for combinations of terms in the positions of subject, predicate, or object. For example, for the question "`Who developed Skype?`" we find the answer "`Microsoft`" from the triple: http://dbpedia.org/resource/Skype–http://dbpedia.org/ontology/developer–http://dbpedia.org/resource/Microsoft. The system returns the more probable answer accompanied by a score, plus a list of other possible answers. In our running example, this tab returns the Museum of El Greco (in Crete).

4.6. Tabs' Roles and Extra Tabs

There are several other tabs that could be supported and could be useful in certain kinds of information needs, e.g., *image* tab, *geo* tab, *time* tab, etc. Each can be construed as a tool that could aid the user to focus on a particular aspect, based on the task/information need at hand, each enacted by a simple click (therefore the required effort is minimal). One rising question is how to provide an *overview* of these in an effortless manner, and/or how to rank them if that is desired. For reasons of transparency and exploration, it is beneficial to make the user aware of the existence of these, instead of promoting and showing only one, as some Web Search Engines (WSE) do. However, we should mention that it is the task of QA to identify the *question type* and the *expected answer type*, therefore, based on the analysis of the QA perspective, a short answer (presented in the appropriate way), could be promoted (just like WSE do), therefore, one direction for further research is to investigate the applicability of approaches like [38,39] for complex questions.

In this current paper we confine ourselves on the previous five tabs since we believe that they are both *KB-independent* and *task-independent*, hence they can be considered to be fundamental. The added value from each of these basic perspectives is summarized in Figure 4. The diagram also shows some main paths that indicate why a user may decide, in a tab-switching interaction, to move from a tab to another (of course, the user is free to follow any order). Below we provide a few additional examples showcasing the benefits from using more than one tab.

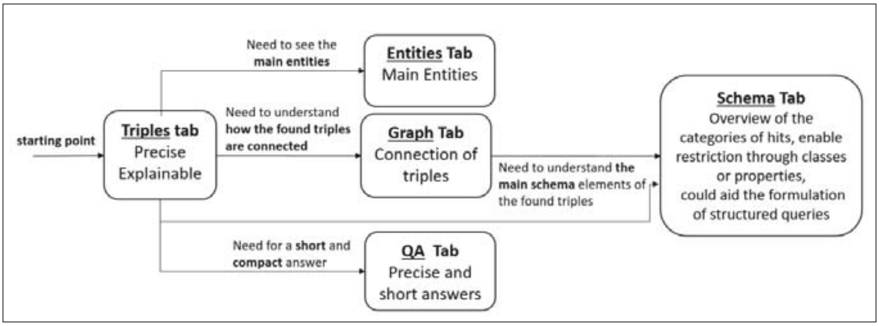

starting point
Triples tab
Precise
Explainable
Need to see the main entities
Entities Tab
Main Entities
Need to understand how the found triples are connected
Graph Tab
Connection of triples
Need to understand the main schema elements of the found triples
Schema Tab
Overview of the categories of hits, enable restriction through classes or properties, could aid the formulation of structured queries
Need for a short and compact answer
QA Tab
Precise and short answers

Figure 4. The Added Value of each Perspective.

For the query *q*="El Greco and Kazantzakis" in the Entities Tab, as shown in Figure 5, the user can find in the first two positions the two main entities of the query, i.e., "El Greco" (the painter), and "Nikos Kazantzakis" (the writer and philosopher), while in the Triples Tab the user can find a triple that connects these two entities. From the Graph Tab the user can see the triples grouped in two poles (one for each entity) and the user can realize that there is only one triple that connect these two poles (in the top-35 triples). Finally, with the Schema Tab the user can refine to Location and find entities whose name is related to the main entities, like "El Greco Apartments" and "Nikos Kazantzakis (municipality)".

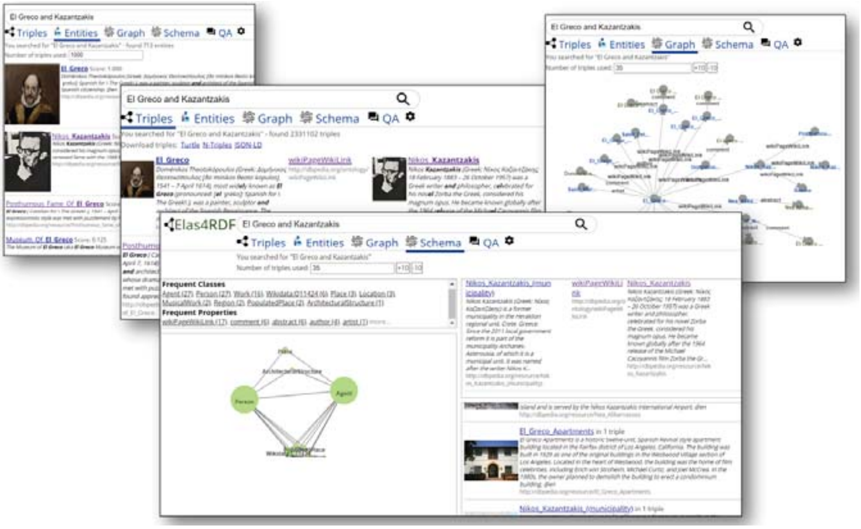
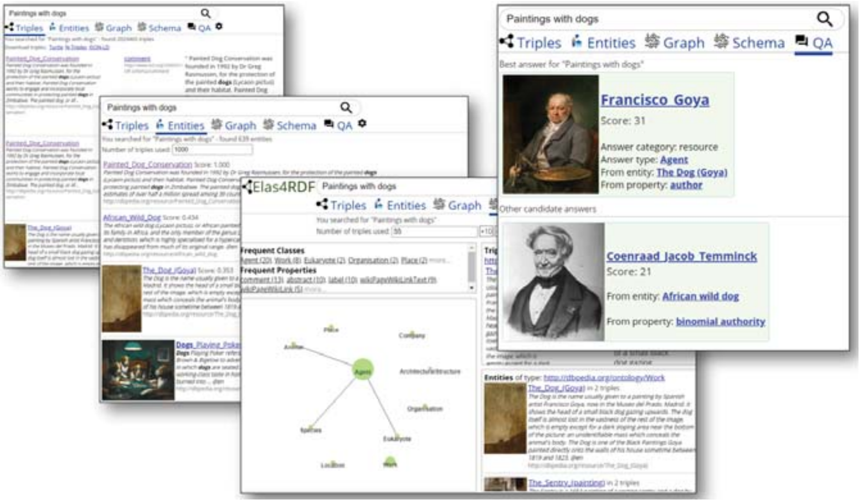


Figure 5. Search results for the query "`El Greco and Kazantzakis`".

As another example, for the query "Paintings with dogs" in the Triples Tab, as shown in Figure 6, the user can find relevant specific information including information about "Painted Dog Conservation" (a non-profit organization for the protection of the painted dog, or African wild dog), information about particular paintings, information about "Greg Rasmussen" the founder of the "Painted Dog Conservation", etc. In the Entities tab the user can find the main entities, including the "Painted Dog Conservation", the species "African Wild Dog", one painting of Goya (The Dog), the "Dogs Playing Poker" (the series of 16 oil paintings by C. M. Coolidge), etc. The Schema Tab shows the classes and properties of the found triples, through which the user can understand that there are related: species, (art) works, locations, etc. Moreover, the user can refine/explore the information space as she wishes to. In Figure 6 the user has refined using the class "Work" and in the right bottom frame he can find various paintings with dogs including: "The Dog (Goya)", "The Sentry (painting)", "The Hunt In The Forest", "Interior With A Young Couple And A Dog" "Portrait Of Charles V With A Dog" etc. Finally, the QA Tab returns two entities "Francisco Goya" (the painter of the painting "The Dog"), and "Coenraad Jacob Temminck" (a Dutch aristocrat, zoologist, and museum director who first described scientifically in 1820 the species African Wild Dog).



Paintings with dogs

Triples Entities Graph Schema QA

Best answer for "Paintings with dogs"

Francisco Goya
Score: 31
Answer category: resource
Answer type: Agent
From entity: The Dog (Goya)
From property: author

Other candidate answers

Coenraad Jacob Temminck
Score: 21
From entity: African wild dog
From property: binomial authority

Elas4RDF Paintings with dogs

You searched for "Paintings with dogs"

Frequent Classes

Frequent Properties

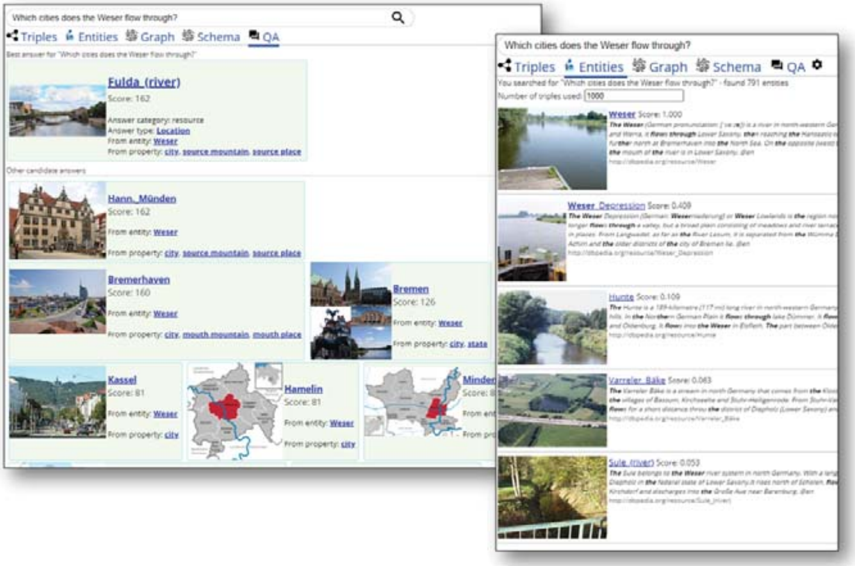
Entities of type: http://dbpedia.org/ontology/Work

The_Dog_(Goya) in 2 triples

The_Sentry_(painting) in 2 triples

Figure 6. Search results for the query "`Paintings with dogs`".

For list questions, i.e., questions with a set of elements as the correct response, like "Which cities does the Weser flow through?" the user may decide to inspect only the QA Tab and the Entities Tab as shown in Figure 7.



Which cities does the Weser flow through?

Triples Entities Graph Schema QA

Best answer for "Which cities does the Weser flow through?"

Fulda (river)
Score: 162
Answer category: resource
Answer type: Location
From entity: Weser
From property: city, source mountain, source place

Other candidate answers

Hann. Münden
Score: 162
From entity: Weser
From property: city, source mountain, source place

Bremerhaven
Score: 160
From entity: Weser
From property: city, mouth mountain, mouth place

Bremen
Score: 126
From entity: Weser
From property: city, state

Kassel
Score: 81
From entity: Weser
From property: city

Hamelin
Score: 81
From entity: Weser
From property: city

Minden

Which cities does the Weser flow through?

Triples Entities Graph Schema QA

You searched for "Which cities does the Weser flow through?" - found 791 entities

Number of triples used: 1000

Weser Score: 1.000

http://dbpedia.org/resource/Weser

Weser Depression Score: 0.409

http://dbpedia.org/resource/Weser_Depression

Hunte Score: 0.109

http://dbpedia.org/resource/Hunte

Varreler Bäke Score: 0.063

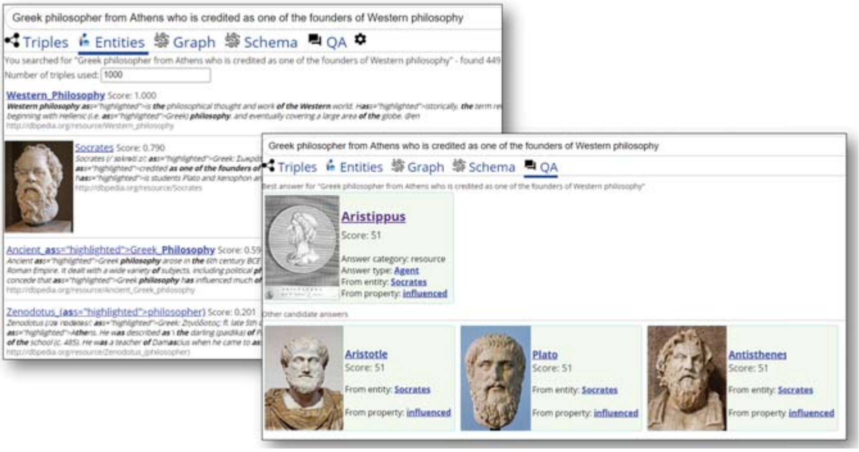
http://dbpedia.org/resource/Varreler_Bäke

Sule (river) Score: 0.053

http://dbpedia.org/resource/Sule_(river)

Figure 7. Search results for the query "`Which cities does the Weser flow through?`".

Longer queries are also possible, for instance for the query "Greek philosopher from Athens who is credited as one of the founders of Western philosophy", from the Entity Tab (as shown in Figure 8) the user we can see that Socrates received the higher score, while from the QA tab the user can see various other philosophers as candidate answers.


Greek philosopher from Athens who is credited as one of the founders of Western philosophy

You searched for "Greek philosopher from Athens who is credited as one of the founders of Western philosophy" - found 449

Number of triples used: 1000

Western_Philosophy Score: 1.000
Western philosophy ass="highlighted">is the philosophical thought and work of the Western world. Hass="highlighted">istorically, the term re
beginning with Hellenic (i.e. ass="highlighted">Greek) philosophy and eventually covering a large area of the globe. @en
http://dbpedia.org/resource/Western_philosophy

Socrates Score: 0.790
Socrates (/ˈsɒkrətiːz/; ass="highlighted">Greek: Σωκράτ
ass="highlighted">credited as one of the founders of
hass="highlighted">is students Plato and Xenophon an
http://dbpedia.org/resource/Socrates

Ancient_ass="highlighted">Greek_Philosophy Score: 0.59
Ancient ass="highlighted">Greek philosophy arose in the 6th century BCE
Roman Empire. It dealt with a wide variety of subjects, including political p
concede that ass="highlighted">Greek philosophy has influenced much of
http://dbpedia.org/resource/Ancient_Greek_philosophy

Zenodotus_(ass="highlighted">philosopher) Score: 0.201
Zenodotus (/zəˈnɒdətəs/; ass="highlighted">Greek: Ζηνόδοτος; fl. late 5th c
ass="highlighted">Athens. He was described as "the darling (paidika) of P
of the school (c. 485). He was a teacher of Damascius when he came to as
http://dbpedia.org/resource/Zenodotus_(philosopher)

Greek philosopher from Athens who is credited as one of the founders of Western philosophy

Best answer for "Greek philosopher from Athens who is credited as one of the founders of Western philosophy"

Aristippus
Score: 51
Answer category: resource
Answer type: Agent
From entity: Socrates
From property: influenced

Other candidate answers

Aristotle
Score: 51
From entity: Socrates
From property: influenced

Plato
Score: 51
From entity: Socrates
From property: influenced

Antisthenes
Score: 51
From entity: Socrates
From property: influenced

Figure 8. Search results for the query "`Greek philosopher from Athens who is credited as one of the founders of Western philosophy`".

5. Evaluation

Below we evaluate the proposed approach by (a) comparing its *functionality* with those of related systems, (b) proving its feasibility by discussing *efficiency*, (c) discussing the retrieval *effectiveness* of the system, and (d) reporting the results of a *task-based evaluation with users* that examines the usefulness of the proposed multi-perspective approach, as well as some results by *log analysis*.

5.1. Comparing the Functionality with Related Systems

Since DBpedia is a core dataset of the Linked Open Data cloud [40], we decided to compare with *interactive systems* (not just APIs) that offer a kind of access/search facility over DBpedia. For this reason, we considered the following systems: LOTUS [22], GraFa [41] (http://grafa.dcc.uchile.cl/), RelFinder [42] (http://www.visualdataweb.org/relfinder.php), DBpedia Search & Find (http://dbpedia.org/fct/), SPARKLIS [43] (http://www.irisa.fr/LIS/ferre/sparklis/), and our system Elas4RDF (https://demos.isl.ics.forth.gr/elas4rdf/).

The results are summarized in Table 1. The table has a column for each of the following features: *triple search*, *entity search*, *graph-view*, *faceted search*, *QA*, *relation finder*, *SPARQL query support*. The last column sums up the number of features each system supports: we count each supported feature with 1, and each partially supported feature with 0.5, as an indicator of the spectrum of the provided access services. We can see that most systems focus on only one or two access methods, while our system offers four, hence it provides a wider spectrum of access services.

Table 1. Search Systems over DBpedia.

System	Triple Retrieval	Entity Search	Graph View	Faceted Search	QA	Relation Finder	SPARQL Support	SUM
LOTUS [22] (no online demo)	Yes	No	No	No	No	No	No	1/7
GraFa [41]	No	No	No	Yes	No	No	No	1/7
RelFinder [42]	No	Partial (through auto completion)	Partial (only of related entities)	No	No	Yes	No	1/7
DBpedia Search & Find	Yes (no images)	No	No	Partial (simple)	No	No	Partial (query display)	2/7
SPARKLIS [43]	No	No	No	Yes (Very Expressive)	No	No	Yes	2/7
Elas4RDF	Yes	Yes	Yes	No	Yes	No	No	4/7

5.2. Efficiency

The efficiency of the back-end search service (i.e., of the ranking service) was evaluated in [8]. Here we focus on the cost for providing the multiple perspectives of the search results. The key point is that the implementation of the perspectives on top of the search service, described in Section 3.2, does not add significant overhead, preserving the real-time interaction. Furthermore, the triples and entities retrieved from the search service are *cached*, further improving load times when the same query is issued on different perspectives.

In Table 2, the average load time of each perspective is displayed (with and without caching), considering 10 queries of varying length from 1 to 8 words and using an instance of the system that runs on a machine with 6 physical cores and maximum memory allocation size set to 8GB. We can see that even without caching all responses are returned in less than 3 seconds, while with caching enabled, the average time is around 150 ms.

Table 2. Average load times for each perspective.

Perspective	Triples	Entities	Graph	Schema	QA
Without caching	980 ms	2582 ms	1018 ms	924 ms	2869 ms
With caching	145 ms	124 ms	91 ms	175 ms	118 ms

5.3. Evaluation of Effectiveness

Another evaluation aspect is the effectiveness of the system, i.e., its capability to fulfill the information needs of the user. Note here that since one can use his own retrieval, ranking or visualization method in any of the fundamental perspectives, evaluating the performance of the method used in each different tab is out of the scope of this paper. As regards the implementation of the tabs in our prototype (described in Section 4), the ranking of the entities in the *entities tab* has been extensively evaluated in [8], demonstrating a high performance. This provides a very positive evidence about the quality of the triples that feed all tabs, in the sense that if triple-ranking were not effective, then it would be hard for the *entities tab* to be effective. More importantly, the results of the user study (that we shall see in Section 5.4) validate the good quality of the results shown in each tab. Specifically, the large majority of users managed to find correct answers for most of the requested tasks. That would be impossible if most of the results in the tabs were irrelevant (more about the user study below in Section 5.4).

5.4. Evaluation with Users

Since there is no dataset that could be used for evaluating the particular multi-perspective interaction we decided to carry out a task-based evaluation with users. Specifically, we wanted to understand how users would use such a system, whether they find useful and/or like the multi-perspective approach, and for collecting general and specific feedback.

5.4.1. Information-Seeking Tasks

Since we are in keyword-search setting (and not in a structured query building process), we selected several tasks that have IR nature, and at the same time are not trivial (some of them are hard to answer, and/or DBpedia has related but not exactly the requested information). We also tried to capture various kinds of information needs, while keeping the list of tasks short for attracting more participants. The selected 11 tasks are shown in Table 3. They include queries of various kinds (entity property queries, entity relation queries, fact checking queries, entity list queries). In total, answering these questions requires at least 30 min.

Table 3. Evaluation Tasks.

ID	Task
T1	Is there any person that is fisherman, writer and poet? Provide at least 3 related names (or URIs).
T2	Is there any writer and astronaut from Russia? Provide related names or URIs.
T3	Find information that relates Albert Einstein with Stephen Hawking.
T4	Find if El Greco was influenced by Michelangelo.
T5	Is there any reference of Freud to the ancient Greece?
T6	How is Mars related to Crete?
T7	Find mathematicians related to Pisa.
T8	Find painters of the Ancient Greece.
T9	Are there drugs that contain aloe?
T10	Which cities does the Weser flow through?
T11	Find at least 5 rivers of Greece.

5.4.2. Participants, Questionnaire and Results

We invited by email various persons to participate in the evaluation voluntarily. The users were asked to carry out the tasks and to fill (anonymously) the prepared questionnaire. No training material was given to them, and the participation to this evaluation was optional (invitation by email). Eventually, 25 persons participated (from 5 May 2020 to 18 May 2020). The number was sufficient for our purposes since, according to [44], 20 evaluators are enough for getting more than 95% of the usability problems of a user interface. In numbers, the participants were 32% female and 68% male, with ages ranging from 20 to 54 years; the distribution is almost uniform, only the age of 23 is the more frequent 20%, as shown in Figure 9.

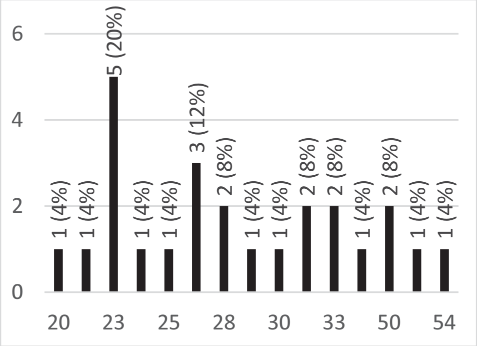


Figure 9. Age distribution of participants.

As regards occupation and skills, all have studied Computer Science, except one Physicist. In detail, 20% were undergraduate students, 15% of them postgraduate computer science students, and the rest computer engineers, professionals and researchers. Students came from at least 3 different universities, while 40% of all the participants have never used DBpedia before. The questionnaire is shown below, enriched with the results of the survey in the form of percentages written in bold:

E1 *How would you rate the Triples tab?*: Very Useful (**40%**), Useful (**44%**), Little Useful (**16%**), Not Useful (**0%**)

E2 *How would you rate the Entities tab?*: Very Useful (**44%**), Useful (**28%**), Little Useful (**24%**), Not Useful (**4%**)

E3 *How would you rate the Graph tab?*: Very Useful (**32%**), Useful (**52%**), Little Useful (**12%**), Not Useful (**4%**)

E4 *How would you rate the Schema tab?*: Very Useful (**16%**), Useful (**40%**), Little Useful (**36%**), Not Useful (**8%**)

E5 *How would you rate the QA tab?*: Very Useful (**16%**), Useful (**36%**), Little Useful (**40%**), Not Useful (**8%**)

E6 *Did you find it useful that the system offers multiple perspective of the search results?*: Very much (**48%**), Fair (**48%**), Not that Useful (**4%**), Not Useful (**0%**)

E7 *Mark the perspective(s) that you think are redundant:* Triples Tab (**0%**), Entities Tab (**8%**), Graph Tab (**8%**), Schema Tab (**40%**) QA Tab (**16%**) All tabs are useful, none is redundant (**44%**)

E8 *Have you used DBpedia before:* Never (**40%**), Only a few times (without using SPARQL) (**16%**), Quite a lot (I have used SPARQL to query it) (**44%**).

E9 *How would you rate the entire system?* Very Useful (**32%**), Useful (**60%**), Little Useful (**8%**), Not Useful (**0%**)

E10 *You can report here errors, problems, or recommendations.* (free text of unlimited length)

5.4.3. Results Analysis and Discussion

User Ratings. As regards *ratings*, most users appreciated the multi-perspective approach (the positive options of E6, Very Much and Fair, sum to 96%). Moreover, all tabs received positive results by some users. By adding the percentages of Very Useful and Useful, the ranked list of *more preferred* tabs is:

$$\langle$$ {GraphTab (84%), TriplesTab (84%)}, EntitiesTab (72%), SchemaTab (56%), QATab(52%) $\rangle$.

The *less preferred* tabs, according to the sum of Little Useful and Not Useful percentages, is:

$\langle$ QATab (48%), SchemaTab (44%), EntitiesTab (28%), {GraphTab (16%), TriplesTab (16%)} $\rangle$.

Please note that these numbers correspond to the percentages of users that *would not be satisfied* if only the corresponding perspective were provided to them.

It is also clear that different users have different preferences for perspectives: there are persons that rated the Schema Tab as Very Useful, while others marked is as Redundant. Probably this depends on the background of the participants: a person with no knowledge of RDF would not be able to understand (and exploit) the notion of schema, and we have seen that 20% of the participants were undergraduate and 40% have never used DBpedia. This is also evident from Figure 10 that depicts the sum of Very Useful and Useful percentages per tab; the black bars correspond to the users that had never used DBpedia, while the white bars correspond to the users that had used DBpedia before.

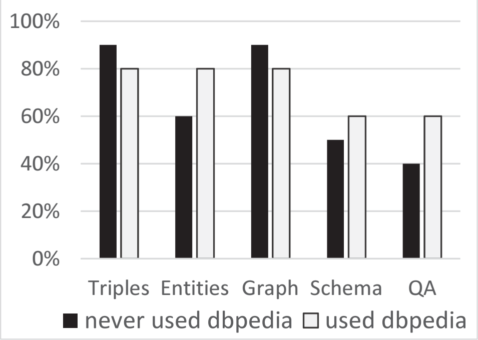


Figure 10. 'Very Useful' and 'Useful' preference percentages per tab and category of users.

By looking at the responses of the questionnaire, we can see that the group of users that had never used DBpedia, preferred the Triples Tab and the Graph Tab (40% found them Very Useful, 50% Useful, and 10% Little Useful, for both tabs), and the least useful tab for them was the Schema Tab (10% Very Useful, 40% Useful, and 50% Little Useful), because a basic understanding of the RDF data model is required to use it. Regarding this user group's opinion of the multi-perspective approach, 30% found it to be Very Useful, and 60% found it Fair. Only one user did not find the approach useful. Also, 50% of these users responded that None of the perspectives are redundant.

Statistical Significance. As regards *statistical significance*, by assuming as *positive* the options Very Useful and Useful, and as *negative* the options Little Useful and Not Useful, the lower bound of *Wilson score confidence interval* shows that with 95% confidence, the percentage of users (of the entire community) that would upvote each perspective would be:

⟨ TriplesTab (65%), GraphTab (65%), EntitiesTab (52%), SchemaTab (37%), QATab (33%) ⟩

Now by considering *all* 4 options quantified as: Very Useful (4), Useful (3), Little Useful (2), Not Useful (1), we can use *Bayesian Approximation* to compute the *expected average rating* for each perspective, in the scale 1 (Worst)–4 (Best), in the entire community of users. These expected ratings are:

⟨ TriplesTab (2.84), GraphTab (2.73), EntitiesTab (2.69), SchemaTab (2.30), QATab (2.27) ⟩

where a perspective with score X means that it will have an average rating greater than X, with 95% confidence.

Task Performance. As regards *task performance*, i.e., the responses to the 11 tasks, from the $11 \times 25 = 275$ responses, 46 (16.7%) reported failure to find the requested information. The failure rate was 20.9% in the (10) users that had never used DBpedia, and 13.9% in the rest (15) users. As shown in Figure 11, the participants faced problems, mainly in T2, T4, T5: T2 is tricky (there is such a space-engineer not astronaut), while T4 and T5 are hard to answer, due to dataset issues (non-existing information, wikiPageWikiLink with no explanation) therefore, these cannot be considered to be failures of the system. Another interesting observation is that for most tasks inexperienced users were almost as successful as experienced ones.

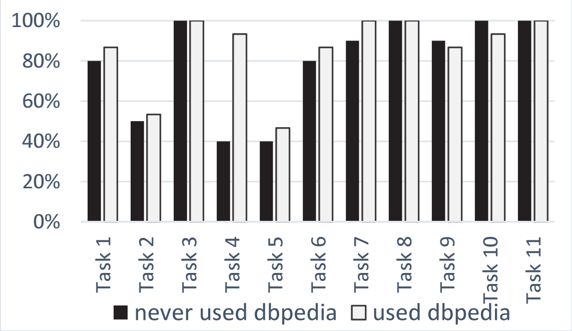


Figure 11. Success rates for experienced and inexperienced users.

Free form Feedback. With respect to the *free form feedback*, 18 of the 25 users provided very interesting and lengthy comments. For reasons of space, here we only summarize the main ones. In general, they (a) spotted problems related to the DBpedia dataset (missing relationships, unexplained `wikiPageWikiLink` relationships, duplicates), and (b) they made suggestions for improving the tabs: Triples Tab (not score with 1.0 a triple if not all query terms are included in that triple, addition of property filters), Schema Tab (add the more frequent labels in the edges of the schema graph, highlight the query words in the hits), Graph Tab (set the size so that all related entities are shown).

General Remarks. Overall, the rating and the feedback that users provided was very positive. Of course, it is not hard to understand that the results depend on the quality of each individual perspective (which in turn depends also on the effectiveness of the underlying search service). Moreover, the *order of tabs* affects the results that concern *user preferences*: in information needs that the first tab(s) provide a satisfying answer, the user will not visit the subsequent tabs (or just a few for verification purposes). That means the harder an information need is, the higher the probability the user visits all tabs. However, our main research hypothesis is not related to the comparison of the individual tabs, but on the usefulness of the multi-perspective approach, and the results of the evaluation provide positive evidence about the value of the multi-perspective approach. Overall, the key finding is that users not familiar with RDF (a) managed to complete the information-seeking tasks (with performance very close to that of the experienced users), and (b) they rated positively the approach.

5.4.4. Log Analysis

Since the system became public and was disseminated in social media on 27 April 2020, below we report some points related to the total traffic of the system; not only from the task-based evaluation with users. More than half of the users (102, in total) have interacted with at least 3 different tabs. The most visited tab is the Triples Tab (35.7% of requests for a tab) which is expected since it is the first tab presented to the user, followed by the Entities Tab (19.1%), the Schema Tab (18.7%), the Graph Tab (16.8%), and the QA tab (9.7%). On average, a user issued 4.6 requests per query (where a request involves: clicking a tab, changing page, adjusting the number of shown triples, or clicking a class or property in the schema tab). Also, a user in average performed 6.7 interactions per query in the schema tab. This is expected since the Schema Tab allows for interactive exploration of the data by clicking on classes and predicates, and adjusting the number of retrieved triples.

5.4.5. Discussion: Related Systems

To our knowledge, the only system that is currently available and offers unrestricted free-text search (which is the focus of our work) is DBpedia Search & Find (http://dbpedia.org/fct/). This system offers a single visualization of the results, in particular it returns *entities*, so it is like using only the Entities Tab provided by our system. The objective of our evaluation is to investigate if a single visualization method is enough, what is answered by the user study; if the Entities Tab were enough, this would be evident in the evaluation results, e.g., in the answers of the questions E1–E7.

6. Concluding Remarks

Keyword search over RDF datasets is a challenging task. To help the user find and explore the requested information, we have investigated a *multi-perspective* approach for keyword search in which multiple perspectives (tabs) are used for the presentation of the search results, each tab stressing a different aspect of the hits. The user can easily inspect all tabs and get a better overview and understanding of the search results. We have focused on five fundamental (i.e., KB and task agnostic) perspectives (triples, entities, graph, schema and QA) and we have implemented this approach over a general keyword search engine over DBpedia.

With respect to related systems that provide keyword access over DBpedia, we could say that the proposed approach is probably the more complete with respect to the access methods that it offers. The task-based evaluation with users has shown that (a) 96% of the users liked the multi-perspective approach (48% Very much, 48% Fair), (b) the success rate of all users was very high (even of those not familiar with RDF), (c) users seem to have quite different preferences on perspectives.

There are several issues that are worth further work and research. We plan to advance the QA tab, to improve the Graph Tab, and to add additional tabs. Moreover, we would like to investigate how to exploit the equivalence (`owl:sameAs`) relationships. The system is available to all at https://demos.isl.ics.forth.gr/elas4rdf/.

Author Contributions: Conceptualization, Y.T.; methodology, Y.T., C.N. and P.F.; software, C.N. and G.K.; validation, Y.T., C.N., P.F.; writing—original draft preparation, Y.T., C.N. and P.F.; writing—review and editing, Y.T., C.N. and P.F.; supervision, Y.T.; project administration, Y.T.; funding acquisition, Y.T. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Acknowledgments: We thank all participants of the user study for dedicating time on the evaluation and providing valuable feedback.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

API	Application Programming Interface
TF-IDF	Term Frequency - Inverse Document Frequency
KB	Knowledge Base
OLAP	Online Analytical Processing
QA	Question Answering
RDF	Resource Description Framework
REST	Representational State Transfer
SPARQL	SPARQL Protocol and RDF Query Language

References

1. Mountantonakis, M.; Tzitzikas, Y. Large-scale Semantic Integration of Linked Data: A Survey. *ACM Comput. Surv. (CSUR)* **2019**, *52*, 103. [CrossRef]
2. Wishart, D.S.; Feunang, Y.D.; Guo, A.C.; Lo, E.J.; Marcu, A.; Grant, J.R.; Sajed, T.; Johnson, D.; Li, C.; Sayeeda, Z.; et al. DrugBank 5.0: A major update to the DrugBank database for 2018. *Nucleic Acids Res.* **2018**, *46*, D1074–D1082. [CrossRef] [PubMed]

3. Jaradeh, M.Y.; Oelen, A.; Farfar, K.E.; Prinz, M.; D'Souza, J.; Kismihók, G.; Stocker, M.; Auer, S. Open Research Knowledge Graph: Next Generation Infrastructure for Semantic Scholarly Knowledge. In Proceedings of the 10th International Conference on Knowledge Capture, Marina del Rey, CA, USA, 19–22 November 2019; pp. 243–246.
4. Dimitrov, D.; Baran, E.; Fafalios, P.; Yu, R.; Zhu, X.; Zloch, M.; Dietze, S. TweetsCOV19–A Knowledge Base of Semantically Annotated Tweets about the COVID-19 Pandemic. In Proceedings of the 29th ACM International Conference on Information and Knowledge Management (CIKM 2020), Virtual Event, Ireland, 19–23 October 2020.
5. Tzitzikas, Y.; Manolis, N.; Papadakos, P. Faceted exploration of RDF/S datasets: A survey. *J. Intell. Inform. Syst.* **2017**, *48*, 329–364. [CrossRef]
6. Papadaki, M.E.; Tzitzikas, Y.; Spyratos, N. Analytics over RDF Graphs. In Proceedings of the International Workshop on Information Search, Integration, and Personalization, Heraklion, Greece, 9–10 May 2019; pp. 37–52.
7. Kritsotakis, V.; Roussakis, Y.; Patkos, T.; Theodoridou, M. Assistive Query Building for Semantic Data. In Proceedings of the SEMANTICS Posters&Demos, Vienna, Austria, 10–13 September 2018.
8. Kadilierakis, G.; Fafalios, P.; Papadakos, P.; Tzitzikas, Y. Keyword Search over RDF using Document-centric Information Retrieval Systems. In Proceedings of the Extended Semantic Web Conference (ESWC'2020), Heraklion, Crete, Greece, 31 May–4 June 2020.
9. Hasibi, F.; Nikolaev, F.; Xiong, C.; Balog, K.; Bratsberg, S.E.; Kotov, A.; Callan, J. DBpedia-Entity V2: A Test Collection for Entity Search. In Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, Tokyo, Japan, 7–11 August 2017; pp. 1265–1268.
10. Kadilierakis, G.; Nikas, C.; Fafalios, P.; Papadakos, P.; Tzitzikas, Y. Elas4RDF: Multi-perspective Triple-centered Keyword Search over RDF using Elasticsearch. In Proceedings of the Extended Semantic Web Conference (ESWC'2020), Heraklion, Crete, Greece, 31 May–4 June 2020.
11. Elbassuoni, S.; Ramanath, M.; Schenkel, R.; Weikum, G. Searching RDF graphs with SPARQL and keywords. *IEEE Data Eng. Bull.* **2010**, *33*, 16–24.
12. Lin, X.; Zhang, F.; Wang, D. RDF Keyword Search Using Multiple Indexes. *Filomat* **2018**, *32*, 1861–1873. [CrossRef]
13. Cheng, G.; Qu, Y. Searching linked objects with falcons: Approach, implementation and evaluation. *Int. J. Semant. Web Inform. Syst. (IJSWIS)* **2009**, *5*, 49–70. [CrossRef]
14. Delbru, R.; Rakhmawati, N.A.; Tummarello, G. Sindice at semsearch 2010. In Proceedings of the 19th International World Wide Web Conference, Aleigh, NC, USA, 26–30 April 2010.
15. Liu, X.; Fang, H. A study of entity search in semantic search workshop. In Proceedings of the 3rd International Semantic Search Workshop, Raleigh, NC, USA, 26–30 April 2010.
16. Delbru, R.; Campinas, S.; Tummarello, G. Searching web data: An entity retrieval and high-performance indexing model. *J. Web Semant.* **2012**, *10*, 33–58. [CrossRef]
17. Ouksili, H.; Kedad, Z.; Lopes, S.; Nugier, S. Using Patterns for Keyword Search in RDF Graphs. In Proceedings of the EDBT/ICDT Workshops, Venice, Italy, 21–24 March 2017.
18. Elbassuoni, S.; Blanco, R. Keyword search over RDF graphs. In Proceedings of the 20th ACM international Conference on Information and Knowledge Management ACM, Glasgow, UK, 19–23 October 2011; pp. 237–242.
19. Blanco, R.; Mika, P.; Vigna, S. Effective and efficient entity search in RDF data. In Proceedings of the International Semantic Web Conference, Bonn, Germany, 23–27 October 2011; pp. 83–97.
20. Pérez-Agüera, J.R.; Arroyo, J.; Greenberg, J.; Iglesias, J.P.; Fresno, V. Using BM25F for semantic search. In Proceedings of the 3rd International Semantic Search Workshop ACM, Raleigh, NC, USA, April 2010; p. 2. Available online: https://dl.acm.org/doi/10.1145/1863879.1863881 (accessed on 27 August 2020).
21. Dosso, D.; Silvello, G. A Scalable Virtual Document-Based Keyword Search System for RDF Datasets. In Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, Paris, France, 21–25 July 2019; pp. 965–968.
22. Ilievski, F.; Beek, W.; van Erp, M.; Rietveld, L.; Schlobach, S. LOTUS: Adaptive text search for big linked data. In Proceedings of the European Semantic Web Conference, Crete, Greece, 29 May–2 June 2016; pp. 470–485.
23. Johnson, T. Indexing linked bibliographic data with JSON-LD, BibJSON and Elasticsearch. *Code4lib J.* **2013**, *19*, 1–11.

24. Bikakis, N.; Sellis, T. Exploration and visualization in the web of big linked data: A survey of the state of the art. *arXiv* **2016**, arXiv:1601.08059.
25. Dadzie, A.S.; Pietriga, E. Visualisation of linked data–reprise. *Semant. Web* **2017**, *8*, 1–21. [CrossRef]
26. Skjæveland, M.G. Sgvizler: A javascript wrapper for easy visualization of sparql result sets. In Proceedings of the Extended Semantic Web Conference, Crete, Greece, 27–31 May 2012; pp. 361–365.
27. Leskinen, P.; Miyakita, G.; Koho, M.; Hyvönen, E. Combining Faceted Search with Data-analytic Visualizations on Top of a SPARQL Endpoint. In Proceedings of the CEUR Workshop, Bolzano, Italy, 20–22 September 2018; pp. 53–63.
28. Vargas, H.; Buil-Aranda, C.; Hogan, A.; López, C. RDF Explorer: A Visual SPARQL Query Builder. In Proceedings of the International Semantic Web Conference, Auckland, New Zealand, 26–30 October 2019; pp. 647–663.
29. Ilievski, F.; Beek, W.; Van Erp, M.; Rietveld, L.; Schlobach, S. LOTUS: Linked Open Text UnleaShed. In Proceedings of the 6th International Workshop on Consuming Linked Data, Bethlehem, PN, USA, 12 October 2015; p. 6.
30. Rihany, M.; Kedad, Z.; Lopes, S. Keyword Search Over RDF Graphs Using WordNet. In Proceedings of the 1st International Conference on Big Data and Cyber-Security Intelligence BDCSIntell 2018, Hadath, Lebanon, 13–15 December 2018; pp. 75–82.
31. Dosso, D.; Silvello, G. Search Text to Retrieve Graphs: A Scalable RDF Keyword-Based Search System. *IEEE Access* **2020**, *8*, 14089–14111. [CrossRef]
32. Stab, C.; Nazemi, K.; Breyer, M.; Burkhardt, D.; Kohlhammer, J. Semantics visualization for fostering search result comprehension. In Proceedings of the Extended Semantic Web Conference, Crete, Greece, 27–31 May 2012; pp. 633–646.
33. Kontiza, K.; Bikakis, A. Web Search Results Visualization: Evaluation of Two Semantic Search Engines. In Proceedings of the International Conference on Web Intelligence, Mining and Semantics (WIMS'14), Thessaloniki, Greece, 2–4 June 2014; pp. 1–12.
34. Mountantonakis, M.; Tzitzikas, Y. LODsyndesis: Global scale knowledge services. *Heritage* **2018**, *1*, 23. [CrossRef]
35. Belth, C.; Zheng, X.; Vreeken, J.; Koutra, D. What is Normal, What is Strange, and What is Missing in a Knowledge Graph: Unified Characterization via Inductive Summarization. In Proceedings of the Web Conference, Ljubljana, Slovenia, 20–24 April 2020; pp. 1115–1126.
36. Oldman, D.; Tanase, D. Reshaping the Knowledge Graph by connecting researchers, data and practices in ResearchSpace. In Proceedings of the International Semantic Web Conference, Monterey, CA, USA, 8–12 October 2018; pp. 325–340.
37. Dimitrakis, E.; Sgontzos, K.; Tzitzikas, Y. A survey on question answering systems over linked data and documents. *J. Intell. Inform. Syst.* **2019**, *55*, 1–27. [CrossRef]
38. Cui, W.; Xiao, Y.; Wang, H.; Song, Y.; Hwang, S.W.; Wang, W. KBQA: Learning Question Answering over QA Corpora and Knowledge Bases. *Proc. VLDB Endow.* **2017**, *10*, 565–576. [CrossRef]
39. Lu, X.; Pramanik, S.; Saha Roy, R.; Abujabal, A.; Wang, Y.; Weikum, G. Answering Complex Questions by Joining Multi-Document Evidence with Quasi Knowledge Graphs. In Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, Paris, France, 21–25 July 2019; pp. 105–114.
40. Auer, S.; Bizer, C.; Kobilarov, G.; Lehmann, J.; Cyganiak, R.; Ives, Z. DBpedia: A nucleus for a web of open data. In *The Semantic Web*; Springer: Berlin, Germany, 2007; pp. 722–735.
41. Moreno-Vega, J.; Hogan, A. GraFa: Scalable faceted browsing for RDF graphs. In *International Semantic Web Conference*; Springer: Berlin, Germany, 2018; pp. 301–317.
42. Heim, P.; Hellmann, S.; Lehmann, J.; Lohmann, S.; Stegemann, T. RelFinder: Revealing Relationships in RDF Knowledge Bases. In *Semantic Multimedia*; Springer: Berlin/Heidelberg, Germany, 2009; Volume 5887; pp. 182–187.
43. Ferré, S. Sparklis: An expressive query builder for SPARQL endpoints with guidance in natural language. *Semant. Web* **2017**, *8*, 405–418. [CrossRef]
44. Faulkner, L. Beyond the five-user assumption: Benefits of increased sample sizes in usability testing. *Behav. Res. Methods Instrum. Comput.* **2003**, *35*, 379–383. [CrossRef] [PubMed]

big data and cognitive computing

Article

OTNEL: A Distributed Online Deep Learning Semantic Annotation Methodology

Christos Makris * and Michael Angelos Simos *

Department of Computer Engineering and Informatics, University of Patras, 26504 Patras, Greece
* Correspondence: makri@ceid.upatras.gr (C.M.); asimos@ceid.upatras.gr (M.A.S.);
Tel.: +30-2610-996-968 (C.M.)

Received: 15 September 2020; Accepted: 27 October 2020; Published: 29 October 2020

Abstract: Semantic representation of unstructured text is crucial in modern artificial intelligence and information retrieval applications. The semantic information extraction process from an unstructured text fragment to a corresponding representation from a concept ontology is known as named entity disambiguation. In this work, we introduce a distributed, supervised deep learning methodology employing a long short-term memory-based deep learning architecture model for entity linking with Wikipedia. In the context of a frequently changing online world, we introduce and study the domain of online training named entity disambiguation, featuring on-the-fly adaptation to underlying knowledge changes. Our novel methodology evaluates polysemous anchor mentions with sense compatibility based on thematic segmentation of the Wikipedia knowledge graph representation. We aim at both robust performance and high entity-linking accuracy results. The introduced modeling process efficiently addresses conceptualization, formalization, and computational challenges for the online training entity-linking task. The novel online training concept can be exploited for wider adoption, as it is considerably beneficial for targeted topic, online global context consensus for entity disambiguation.

Keywords: named entity disambiguation; text annotation; word sense disambiguation; ontologies; Wikification; neural networks; machine learning

1. Introduction and Motivation

Named entity disambiguation (NED) is a process involving textual mention resolution and assignment to predefined concepts from a knowledge base or concept ontology. The deterministic identification and linking of semantically dominant entity mentions, based on contextual information available, is not trivial in most cases; ambiguity is common on unstructured corpora, as homonymy and polysemy phenomena are inherent to natural languages.

Advances in the domains of artificial intelligence, information retrieval, and natural language processing, outlining the requisition of semantic knowledge input, such as common paradigms like the bag of words representation, are proven inefficient for deeper knowledge extraction and, hence, higher accuracy. As a result, NED is a common step for many relevant tasks including information retrieval [1], data mining [2,3], and web and semantic search [4–7], consequently being a vital component of the artificial intelligence (AI), internet and information industries.

One of the basal challenges in NED involves the maintenance of knowledge resources, especially as new domains and concepts arise or change dynamically over time. In recent years, Wikipedia has been leveraged as a knowledge base and concept universe due to its online nature. The introduction of Wikipedia in the domain derived noteworthy leaps in classic challenges such as knowledge acquisition and adversarial knowledge resolution, as its articles tend to summarize widespread and commonly accepted concepts.

Deep learning architectures have recently been established in several scientific fields including machine translation, computer vision, medical image analysis, speech recognition, audio recognition, social networks, bioinformatics, and material inspection. As such, methodologies successfully modeled high-level abstraction patterns, leveraging deep multilevel transformations, and several approaches successfully addressed the NED problem. However, a series of factors constitute a challenging background for the task.

The engagement of deep learning architectures preconditions the dimensionality projection to lower-dimension spaces for the training input as computational challenges with large-scale training datasets arise. Consequently, the training input entropy is abstracted during a dimensionality reduction process aiming at fitting sparse input spanning plenteous domains to predefined sized dimension spaces, mainly for computational feasibility purposes. As a result of this computational complexity and accuracy trade-off, the inference of semantics deviant from the dominant patterns is burdensome. In addition, the extensive training process required in the scope of a general-purpose NED application is demanding from a computational complexity perspective as outlined in [8]. Our introduced methodology employs an alternative modeling and dimensionality reduction approach method, detailed in Section 3.

Another fundamental adversity for the task resides in knowledge acquisition, including adversarial knowledge resolution and the impact of noise in the training input. Successful deep learning applications require vast training sets. As the task is based on facts for semantic acquisition of pertinent sense representation associations in the available context, the intricacy of semantic attainment, defined as *knowledge acquisition bottleneck* in [9], is dominant. Consequently, the attainment of high-quality data at a scale for wide semantic coverage is not trivial. Similar works as detailed in [8] often rely on diffusive sources ranging from structured ontologies to unstructured corpora, for example, by inducting context with unsupervised techniques for the inference of co-reference information. The impact of noise in the training input is critical for attaining high accuracy at scale. On the contrary, uncontrolled data cleansing approaches aiming at eliminating anomalies on the input training sets could result in substantial information loss for the resolution of more intricate and less frequent senses of a polysemous anchor.

In this work, we propose a novel approach for efficient NED. In particular, by employing divergent thinking on the main task impediments described above, we propose a model for dimensionality reduction according to topical confinement in the context of online training. We focus on minimizing the impact of input data loss and simplifying the task by leveraging topical inference using a semantic ontology information network representation of Wikipedia.

The remainder of this manuscript is organized as follows: the necessary background and related work are presented in Section 2. Our methodology and implementation details are presented in Section 3. The experimental process is described and assessed in Section 4. Our final conclusions are presented in Section 5, along with potential improvements and future work.

2. Background and Related Work

The NED task requisites a knowledge base or concept ontology as its foundation for the identification of named entities, to resolve text segments to a predefined concept or sense universe. Human domain experts also need such a knowledge ontology for identifying the appropriate sense of a polysemous mention within a context. As the creation of knowledge resources by human annotators is an expensive and time-consuming task, facing implications as new concepts or domains emerge or change eventually, the knowledge acquisition issue has been pervasive in the field. The maintainability, coverage, and knowledge acquisition challenges have been outlined on several manually created ontologies applied to the NED task. As a result, attempts for unifying such ontologies emerged; however, they encountered accuracy issues throughout the unification process.

As Wikipedia is an online crowdsourcing encyclopedia with millions of articles, it constitutes one of the largest online open repositories of general knowledge. Wikipedia articles are created and maintained by a multitudinous and highly active community of editors. As a result, widespread and commonly accepted textual descriptions are created as derivatives of a diverse knowledge convergence process in real time. Each article can be interpreted as a knowledge entity. As Wikipedia's online nature inherits the main principles of the web in a wide and highly active user base, named entity linking with Wikipedia is among the most popular approaches in several similar works. The rich contextual and structured link information available in Wikipedia along with its online nature and wide conceptual coverage can be leveraged toward successful high-performance named entity linking applications.

2.1. Named Entity Disambiguation Approximations

Among natural language processing domain tasks, NED and word sense disambiguation (WSD) are acknowledged as challenging for a diversity of aspects. WSD was defined as AI-complete in [8]. AI-completeness is defined by analogy to the nondeterministic polynomial completeness (NP-completeness) concept in complexity theory.

Several formalization approaches have been applied at entity linking coarseness scopes ranging from specific sense ontological entities to generic domains or topics. The disambiguation coverage spans from the disambiguation of one to several senses per sentence. Domain confinement assumptions may also be present on the entity universe.

According to [8], WSD and, hence, NED approaches may be broadly classified into two major categories:

Supervised machine-learning methodologies are used for inferring candidate mentions on the basis of knowledge inference from labeled training sets, usually via classification techniques.

Unsupervised methodologies are based on unstructured corpora for the inference of semantic context via unsupervised machine-learning techniques.

A second level further distinction according to knowledge sources involved can be made as follows:

- knowledge-based, also known as knowledge-rich, relying on lexical resources such as ontologies, machine-readable dictionaries, or thesauri;
- corpus-based, also known as knowledge-poor, which do not employ sense-labeled knowledge sources.

Supervised knowledge-based NED methodologies are in the limelight of current research focus. Wikipedia is commonly employed for underlying knowledge base representation as an entity linking ontology.

2.2. Early Approaches

The pioneering works on the NED problem using Wikipedia for the entity linking approach were [9–11]. The works proposed methods for semantic entity linking to Wikipedia. Those early methods clearly captured the technical impediments of the task, while proposing some effective early solutions. Foundations for future work were placed by the establishment of the commonness feature value for the task.

In [12,13], a more sophisticated approach to the task led to the introduction of relatedness among Wikipedia articles as an invaluable measure of semantic compatibility. Relatedness was defined as the inbound link overlap between Wikipedia articles. The coherence of input text anchors disambiguated with unambiguous mentions of the input was used as the core of the introduced models. Specifically, ambiguous mentions were ranked on the basis of a global score formula involving statistics, relatedness, and coherence for the final selection.

The segmentation of the ranking scores to local and global resulted in further improvements in [14]. Local scores were leveraged for the contribution representation of contextual content surrounding an anchor being processed. The consensus among every input anchor disambiguation within the full frame of the input was modeled as a global score. The problem was formalized as a ranking selection and a quadratic assignment problem, aiming at the approximation of an entity mention for each anchor on the basis of a linear summation of local and global scores.

Another suite with a focus on accuracy and computational efficiency of short input was introduced in [15]. The work is particularly popular and established as a baseline to date. Relatedness, commonness, and other Wikipedia statistics were combined in a voting schema for the selection of the top scoring candidate annotation for a second-step evaluation and selection process.

An alternate modeling approximation was used by [16,17]. An undirected weighed graph was used for the knowledge base representation. The graph nodes were used to model entity annotations or candidate entities. The weighted edges among entities of the graph were used for representing relatedness. Weighted edges among mentions and entities of the graph were used to model contextual similarities. These representations were referred to as the mention-entity graph in [16], and a dense subgraph search approximation was used for the selection of a subgraph of anchor nodes, each containing a unique mention-entity edge. In [17], the representation was referred to as a referent graph, and the methodology employed was based on the PageRank algorithm.

In [18], some innovative approaches for text annotation and entity linking were contributed. Voting schema approximations were introduced, along with a novel method inspired by the human interpretation process on polysemous contexts. An iterative method approach was employed for the modeling process. The iteration converged to proposed annotations while balancing high accuracy with performance, leveraging established metrics derived from the Wikipedia graph representation.

A graph knowledge base representation was employed by [19], and a Hyperlink-Induced Topic Search (HITS) algorithm variant using a breadth first search traversal was evaluated with the candidate entities of the input text anchors as initial seed nodes. Coreference resolution heuristics, extension of surface forms, and normalization contributions to the system constituted the core of this work.

The architecture of [15] was refined and redesigned in WAT [20], as several methodology variants were introduced for experimental assessment. The three-component architecture was revisited by some PageRank and HITS algorithm-based approaches. The main components were thoroughly assessed, and results for a series of methodologies were contributed to the community.

2.3. Recent Deep Learning Approaches

A leading deep learning approximation for the problem was presented in [21]. A vector space representation was used for modeling entities, context, and mentions. The core methodology architecture consisted of a convolutional neural network, in various context windows, for the projection of anchors on the continuous vector space. Finally, a computationally demanding methodology employing a tensor network introduced context and mention interaction information. A similar vector space representation approach of mentions and entities was also employed in [22]. The core disambiguation methodology extended the skip gram model using a knowledge base graph. At a second level, the vector space proximity optimization of vectors representing coherent anchors and entities was used for concluding the process

The authors of [23] introduced a suite combining established approaches, such as graph representation and knowledge base statistics, with deep learning benefits, involving an ensemble consensus disambiguation step. Specifically, variable sized context windows were used by a "neural attention mechanism" with an entity embedding representation.

As most systems rely on heuristics or knowledge-based approaches for conducting semantic relation evaluations, such as coreference, relatedness, or commonness for the conceptual compatibility assessment, the authors of [24] followed a neural entity linking approach, modeling relations as latent variables. Specifically, they extracted semantic relations in an unsupervised manner using an end-to-end optimization methodology for selecting the optimal mentions. The proposed multi-relational model exhibited high performance throughout an experimental evaluation process.

The problem was also addressed in [25] by leveraging a knowledge graph representation. This work was based on the observation that the link density on the representation graph plays a key role as the vertex degree had a major impact to the selection of strongly coherent nodes. To that end, their methodology induced a density enhancement step on the graph representation on the basis of cooccurrence statistics from an unstructured text for relational inference. A training step of entity embeddings was employed for extracting similarity results for the linking step. As anticipated, the system presented exceptional results for the less densely interconnected concepts on the input, resulting in high performance throughout the experimental assessment through a simple, yet effective method.

The authors of [26] attempted to address weaknesses in previous global models. Specifically, by filtering inconsistent candidate entity annotations, they successfully simplified their proposed model while reducing noise on data input. The task was treated as a sequence decision problem, as a sequential approach of exploiting disambiguated mentions during the disambiguation of subsequent anchors was applied. Finally, a reinforcement learning model was used, factoring in a global context. The experimental results outlined accuracy and high generalization performance.

2.4. Conclusions and Current Limitations

Following the success of deep learning methodologies on AI tasks, several similar research endeavors approached the NED, using deep neural network architectures, furthering the outstanding research works outlined above. However, the input dimensionality challenges placed considerable impediments of production-ready, computationally efficient methodologies as outlined in [27]. The complexity of recent approximations employing deep learning architectures led to several recent works, including [28] which queried whether deep neural network NED methodologies are currently applicable for industry-level big data AI applications compared to simpler and more scalable approaches. Current methods focus more and more on accuracy instead of run-time performance, neglecting the options for complexity reduction in many cases, by focusing on input dimensionality for complexity reduction. To that end, systems, like RedW employ a performance-oriented approach relying on graph and statistical analysis features, questioning deep neural network approaches at scale. As deep learning methodologies have been established in terms of knowledge inference and enhanced modeling capabilities, a computationally efficient approach bridging complexity and performance would be propitious for wide, industry adoption.

3. Materials and Methods

3.1. Notations and Terminology

For readability enhancement purposes, this section presents a terminology and notation summary. The terminology in use is aligned with widely adopted previous works in the domain.

- The Wikipedia articles are also referred to as Wikipedia entities, denoted as p.
- A text hyperlink to a Wikipedia page is denoted as a *mention.*
- Text hyperlink anchors within Wikipedia pointing to another page or article are referred to as anchors and denoted as a. Indices are used for referral to specific items in the anchor sequence as follows: a_0 is the first anchor, $i + 1$ and so on. The number of anchors of a text input is cited as m.
- The notation p_a refers to one of the candidate Wikipedia page senses of the anchor a.
- The set of linkable Wikipedia entities to an anchor a is denoted as *Pg(a).*
- The ensemble of inbound links to a given Wikipedia entity p is represented using *in(p).*
- The size of the Wikipedia entities ensemble is cited as $|W|$.
- *link(a)* refers to the cardinality of the count of an anchor's indices as a mention.
- *freq(a)* denotes the total occurrence count of an anchor text within a corpus, including free text and hyperlinks.
- *lp* denotes the link probability of a text segment.

A formal definition of the task on the basis of the above notation can be summarized as the selection of the best fit mention to a p_a from *Pg(a)* for each anchor a_i from the set of identified anchors of a text input.

3.2. Knowledge Extraction

Knowledge is fundamental in an NED task. The current work relies on semantic information from hyperlink anchors to Wikipedia entities. Our methodology supports knowledge acquisition by incorporating any Wikipedia annotated corpus as a training set. In the scope of this work, we leverage the corpus of Wikipedia and the annotated inter-wiki hyperlinks for composing the mention universe ensemble. This ensemble of potential anchor senses grows in parallel with Wikipedia's corpus link structure and its semantic coverage, and it can be considered a sound foundation for our knowledge acquisition process, due to the collaborative online nature of the encyclopedia. Wikipedia entities are widely adopted as an underlying representation ontology for the task due to their commonly accepted textual descriptions.

The population of the mention universe requires the ensemble of Wikipedia pages in MediaWiki article namespace, i.e., pages in namespace with identifier 0. Redirect page unification is carried out for the inclusion of the redirect link context. This involves following the redirect chains and accordingly updating Wikipedia hyperlinks to the corresponding direct link entity IDs. As in [10,11,14,15,20], a preprocessing of a Wikipedia snapshot can be used for the initial extraction of the mention universe, which can remain up to date in syndication with the Wikimedia Update Service [29]. The process involves harvesting the following:

- *Anchor ID*: by keeping an identifier encoding for each text segment encountered as a hyperlink on the processed Wikipedia snapshot.
- *Mention entity ID*: the Wikipedia ID pointed to by a mention. Maintaining this information is necessary for deriving relatedness and commonness statistics.
- *Source article ID*: the Wikipedia article ID where an individual mention is encountered. This is necessary for relatedness calculations.

The above structures constitute the core of the knowledge acquisition for the extraction, transformation, and loading of our training dataset universe, effectively composing a graph representation of Wikipedia. In addition, an *anchor occurrence count* dictionary was extracted and maintained for link probability calculations via a second parse of the corpus for implementation simplicity. An appropriate indexing mechanism can be implemented for avoiding this second parse.

The mitigation of noise impact during the knowledge acquisition phase is crucial to the success of our NED methodology and any deep learning model. In the first stage, following an approach inspired by [25], we performed a mention dictionary density enhancement, by incorporating redirect title information. Specifically, page and redirect titles were treated as hyperlinks in a special source article ID. In the next step, unlike many recent deep learning approaches employing a coarser approximation, we applied filtering rules for ignoring low-frequency mention anomalies, with a relative threshold up to 0.5%, at a minimum of one occurrence. Common preprocessing rules for discarding stop-words, punctuation, single characters, and special symbols were also applied to the extracted mention vocabulary, as established in similar works [12–20]. The knowledge extraction phase was straightforward for both the initial loading and the online syndication of the mention universe, as real-time updates were performed in the structures outlined above.

As an outcome from the knowledge extraction process, Wikipedia was encoded in a mention database, enabling the next steps.

3.3. Methodology

The focus of this work was oriented toward the named entity disambiguation task. The task prerequired anchor extraction. The entity universe to be linked by our system was derived as detailed in the previous section for creating a mention database. In the first step, an extraction, transformation, and loading process was carried out on the unstructured text input for disambiguation (Section 3.3.1). As a next step, we applied a topical coherence-based pruning technique for the confinement of the entity linking scope to coherent topics in a given context (Section 3.3.2). Then, we employed a novel deep learning model for the selection of candidate mentions of an anchor on a local context window, modeling the problem as a classification problem (Section 3.3.3). Finally, a quantification of uncertainty scoring step followed for the confidence evaluation of outcome predictions (Section 3.3.4). Figure 1 outlines our methodology. In the remainder of this manuscript, our methodology is referred to as OTNEL (Online Training Named Entity Disambiguation).

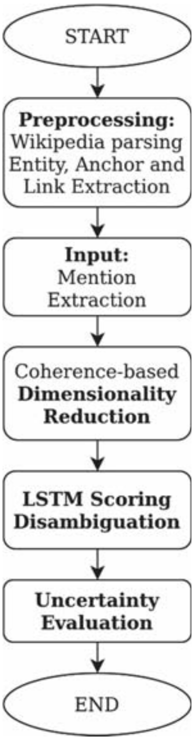


Figure 1. OTNEL (Online Training Named Entity Disambiguation) methodology flowchart.

3.3.1. Extraction, Transformation, and Loading

In the first stage, the unstructured text input was parsed for extracting candidate anchors along with their candidate mentions for further evaluation in the following steps. The input underwent a tokenization process for composing candidate n-grams, with n sized from 1–6. The candidate n-grams were initially evaluated for existence in our mention database as in similar works [10,13,15,20]. A n-gram present in the database could be identified as an anchor for annotation. However, the case of overlapping n-grams needed further examination. The *link probability* of a mention as outlined below by Equation (1) was basial in this examination process.

$$lp(a) = \frac{link(a)}{freq(a)}. \tag{1}$$

Link probability expresses the probability of a word or text fragment occurring as a hyperlink within a corpus. As expressed above, *link(a)* denotes the number of occurrences of anchor *a* as a link. The notation *freq(a)* depicts the occurrence frequency of *a* in a corpus. To preserve meaningful mentions and filter semantically meaningless mentions, *n*-grams with link probability less than 0.001 were pruned similarly with the corresponding knowledge extraction process. As link probability indicates link worthiness, in cases of overlap, the *n*-gram with the highest link probability was selected. Stop-words, punctuation, special symbols, and characters were ignored as they did not return matches in the mention database not being present in the mention universe due to the relevant filtering during the knowledge extraction phase. The specific *n*-gram length was selected in accordance with the maximum size of links on our dataset. Larger *n*-gram lengths would have no effect. Smaller lengths would confine the maximum token length of detected anchors. After this step, unstructured text segments were converted to sequences of semantically significant anchors.

3.3.2. Coherence-Based Dimensionality Reduction

Generic deep learning approximations to the problem face feasibility intricacies at scale for the NED task. On the other hand, in the similar task of named entity recognition, the problem space is limited for the NED task. Specifically, the problem space dimension spans to the mention universe registered on the underlying knowledge base. A perusal of the Wikipedia knowledge graph representation delineates relevant topic coherence with a high degree of reciprocity that can be exploited for discarding incoherent entity mentions from further evaluation in a given context.

As the current work constitutes online semantic annotation, we applied topical confinement for our predictions in terms of the online training process. Specifically, in the first stage, the knowledge graph was pruned to the candidate mentions set of identified input anchors. In the next step, we recalled a training set consisting of mentions within that specific subgraph. This process could be iterated until a wider subgraph was covered, forming a clique from the knowledge graph. As our aim was a vast reduction in the dimensionality space involved, enabling on-the-fly training, a single iteration was performed. The trained model for the specific topical confinement could be persisted for future predictions or on-the-fly training expansion as training input becomes available. As a result, our methodology's dynamic adaptation to rapid underlying knowledge changes as twofold. Firstly, as described in Section 3.2, our mention universe and knowledge graph remained up to date in near real time in syndication with the Wikimedia Update Service [29]. Secondly, the online training process enabled eventual adaptation on potential knowledge changes or updates within the scope of the topical confinement.

It is worth noting that existing approaches to the problem are entrenched by focusing on the generality in a shared vector space with a unified model across topics. However, our approach efficiently confines the training scope required to explicitly generate coherent topical context. The application of similar works following unified approaches would be prohibitive in the online scope of the NED task.

3.3.3. Named Entity Disambiguation

For unambiguous anchors identified by the extraction step, a single entity annotation is available and can be used for linking, i.e., $|Pg(a)| = 1$. For cases where $|Pg(a)|$ has more than one entry, a compatible mention evaluation is needed. Polysemous anchors may have several candidate entities for linking derived from the relevant mention ensemble of the knowledge base. However, using the knowledge base entity dimension as our output dimension would place exorbitant barriers on performance. For named entity disambiguation, we modeled the process as a feature-based classification problem, leveraging an architecture of long short-term memory (LSTM) cells in an artificial recurrent neural network (RNN). Specifically, the problem of selecting a coherent mention for a polysemous anchor in a context was modeled as a binary classification problem as described below.

For every candidate mention of an anchor, we evaluated the classification to the following complementary classes:

- class 1: the compatibility of the mention in the given context;
- class 0: the incompatibility of that mention in the given context.

In the next phase, we could utilize the penultimate deep learning model layer scoring for depicting class predictions probabilities. We selected the highest scoring class 1, i.e., compatible mention, as the disambiguation result.

For maintaining a low input dimension in our model, in the performance context of the online scope of the problem, we provided a set of three features at the input layer, summarizing the gist of topical semantic information. Those features were as follows:

An *inter-wiki Jaccard index average*, as shown in Equation (2). This formula expresses the reciprocity of inbound mentions. The feature of Jaccard similarity was established in [20] as a strong Wikipedia entity coherence measure.

$$avg\ interwiki\ Jaccard\ index(a_i) = \sum_{k=0}^{k=i-1} \frac{\left|in\left(p_{a_i}\right) \cap in\left(p_{a_k}\right)\right|}{\left|in\left(p_{a_i}\right) \cup in\left(p_{a_k}\right)\right|} / \mathrm{m} + \sum_{k=i+1}^{k=m} \frac{\left|in\left(p_{a_i}\right) \cap in\left(p_{a_k}\right)\right|}{\left|in\left(p_{a_i}\right) \cup in\left(p_{a_k}\right)\right|} / \mathrm{m}. \quad (2)$$

Relatedness is an established measure of semantic entity relatedness. The feature has been used as a core disambiguation primitive in several works [13–15]. In this case, we applied an *average relatedness* feature as depicted by Equation (3).

$$avg\ relatedness(a_i) = \sum_{k \in \{p_{a_0} \ldots p_{a_m}\} - \{p_{a_i}\}} \frac{\log\left(\max\left(\left|in\left(p_{a_i}\right)\right|, \left|in\left(p_{a_k}\right)\right|\right)\right) - \log\left(\left|in\left(p_{a_i}\right) \cap in\left(p_{a_k}\right)\right|\right)}{\log(|W|) - \log\left(\min\left(\left|in\left(p_{a_i}\right)\right|, \left|in\left(p_{a_k}\right)\right|\right)\right)} / \mathrm{m}. \quad (3)$$

Commonness as defined by Equation (4) is the prior probability of an anchor pointing to a specific Wikipedia entity. Commonness was broadly used in similar works, contributing significant statistical information to the model.

$$Commonness(p_k, a_i) = P(p_k | a_i). \quad (4)$$

Figure 2 presents the deep learning layers of our classifier's distributed architecture. The classifier received a three-dimensional vector of feature scores as input, summarizing the contextual compatibility of an evaluated candidate mention for an anchor. This evaluation was derived as a classification score for the binary output compatible/incompatible classes. As more than one or (in rare cases) even none of the candidate mentions were classified as compatible in a context, we exploited the penultimate layer score for deriving a relative prediction to select the most coherent mention.

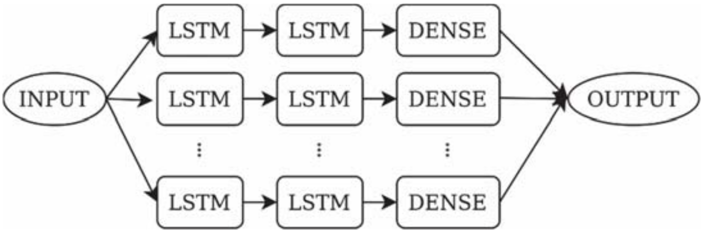


Figure 2. The proposed methodology classifier layer architecture.

The model's input dimensionality was intentionally maintained low through the employment of established context features. Leveraging Keras' [30] and Tensorflow's [31] distributed execution, the input was equally split and fed to the distributed deep learning pipelines. Each individual output was combined to form our classifier output. The complexity was maintained as simple as possible for computational performance reasons.

LSTM was established for addressing RNN's limitations [32]. General RNNs exhibit the "vanishing gradient problem", resulting in a declining effect of the contribution of previous training inputs. The LSTM layer building blocks comprise cells featuring typical input and output gates, along with a forget gate enabling a connected cell state and output recurring feedback from previous states. The *sigmoid* function is commonly adopted as an activation function in LSTM architectures [33] for the input, forget, and output gates, efficiently exhibiting the characteristics and mathematical assets of an effective activation function. The *Tanh* function, a hyperbolic tangent function, is commonly applied as the LSTM candidate and output hidden state.

Our model consisted of two stacked LSTM cell layers after the input followed by a dense layer, producing the output summary. In our model implementation, the LSTM cell size used for the first and second layer was 3, thereby maintaining a low training complexity, yet a high degree of nonlinear feature relation adaptivity. The *MSE* loss function and *Adam* optimizer were used during the model training phase. The *Tanh* activation function and *sigmoid* recurrent activation were employed for the LSTM layers parameters.

The intuition behind the specific multilevel LSTM layer architecture was to involve enhanced semantic relation persistence from the topically confined training sequence. In addition to a clear architecture, simplicity, and modeling and computational efficacy, the methodology enables enhanced prediction strength via exploiting a rich set of both positive and negative linking training examples.

As depicted in the comparative results on a different domain by [33], several activation function options may be explored and compared, contributing intriguing results even in the case of simple classification problems and LSTM architectures. However, in the scope of the current work, we focused on the general approach of a model for the problem, applying established activation functions that experimentally exhibit efficient results for a range of relevant domains. Further exploration of tunning options for our deep learning architectural approach in the specific domain is among our plans for future work.

3.3.4. Quantification of Uncertainty

The NED task is quite demanding, with several cases of variant semantics, insufficient underlying information, or highly ambiguous context. Absolute accuracy may be considered unattainable even for humans on the task. As a result, the confidence evaluation of a named entity prediction is momentous for the development of successful applications. At the pre-output layer of our deep learning model architecture, we could fruitfully exploit the output score as a quality indication for the predicted positive linking compatibility class outcome.

For an anchor a, the candidate mention set size is $|Pg(a)| = k$. Let *compatibility score(m)* denote the compatibility score of mention m. Let the candidate mentions set for anchor a in *Pg(a)* be denoted as $\{m_1, m_2 \ldots\ m_k\}$.

Hence, the uncertainty quantification formula can be defined as follows:

$$U(a, m_a) = \frac{compatibility\ score(m_i) -\ compatibility\ score\big(m_j\big)}{compatibility\ score(m_i)}\ ,$$
$$m_i : max(compatibility\ score(m_i \in \{m_1,\ m_2 \ldots m_k\})),$$
$$m_j :\ max(compatibility\ score(m_j \in \{m_1,\ m_2 \ldots m_k\} - \{m_i\})\,.) \tag{5}$$

Equation (5) is an expression of the semantic distance between the selected mention for an anchor annotation and the next most coherent available mention for that annotation in a specific context. This metric was proven as a good uncertainty indication throughout our experimental evaluation.

3.4. Evaluation Process

The evaluation analysis focused on the entity linking disambiguation process, delineating the benefits of our novel methodology. The uncertainty score introduced for our methodology was thoroughly validated as a confidence indicator for the outcome prediction. For performance comparison, a classic precision, recall, and F1 measure assessment was carried out. Specifically, we evaluated precision by depicting the ratio of valid anchor mention annotations over the size of the identified mention ensemble.

$$\text{Precision} = \frac{TP}{TP + FP}. \tag{6}$$

We evaluated recall on the basis of the number of correctly annotated predictions divided by the total predictions made.

$$\text{Recall} = \frac{TP}{|mentions|}. \tag{7}$$

Lastly, the F1 score outlined a harmonic mean between recall and precision.

$$\text{F1} = 2 * \frac{Precision \times Recall}{Precision + Recall}. \tag{8}$$

The wiki-disamb30 dataset, introduced by [15], was utilized by several works in the domain, including [15,19,20,30], and it is generally accepted as a baseline for the task. Our methodology evaluation process was based on segments of the wiki-disamb30 dataset for a thorough performance analysis. This dataset contains approximately 1.4 million short input texts up to 30 words, incorporating at least one ambiguous anchor hyperlink along with its corresponding Wikipedia entity. As the dataset target entity links correspond to an old instance of Wikipedia, some processing is required for updating the references to the current changes. The dataset in use features extensive context variability, as the text segments cover a wide range of topics, making it ample for a thorough assessment.

As this work introduces and studies a specific NED problem, namely, online training NED, our main focus was the evaluation of our methodology, using precision, recall, and F1 measures. However, the established baseline methodologies from [15] along with the systems proposed in [34] were included for an incommensurate yet indicative performance comparison, outlining the performance of our methodology under a common evaluation dataset.

The first baseline, TAGME [15], is a particularly popular and relatively simple to implement methodology featuring computational efficiency. Relatedness, commonness, and other Wikipedia statistics were combined in a voting schema for the selection of the top scoring candidate annotation for a second-step evaluation and selection process. We aimed to extract insights from a comparison with classic baseline high-performance approaches. The second baseline employed was the Clauset–Newman–Moore (CNM) methodology from [34]. This approach introduced community detection algorithms for semantic segmentation of the Wikipedia knowledge graph into densely connected subgraphs, achieving high accuracy. A classification model approach was employed for the task, using established features along with community coherence information derived by the Clauset–Newman–Moore algorithm.

4. Results

Our methodology assessment was performed using Wikipedia snapshots for the knowledge extraction process. Specifically, the enwiki-20191220-pages-articles dump of pages from 20 December 2019 [35] was employed for an extraction, transformation, and loading process in the WikiText [36] format using the Apache Spark framework [37]. Big data techniques were eventful for the process, deriving more than 37,500,000 individual anchors and over 19,300,000 entities, composing a graph with over 1,110,000,000 edges. The distributed architecture and processing model of our implementation can handle a far larger scale, and its scalability capabilities allow following the growth rates of Wikipedia datasets. For the NED disambiguation process implementation, we used Keras [30] and TensorFlow [31]. Our experiments employed a distributed 192vCPU and 720 GB memory Google Cloud Platform setup.

4.1. Experimental Analysis Discussion

Our OTNEL implementation was experimentally evaluated as dominant for high precision performance, as outlined in the comparative results of Figure 3 and Table 1. Specifically, the methodology indicatively outperformed the baseline methodologies at full recall by 7%. The inclination of precision–recall in similar works in the generic NED scope was impetuous at recall levels above 0.6. This fact was interpreted as the inadequacy of those methodologies to fit a generic model for low-frequency or poorly linked entities in the knowledge graph. Conversely, in the case of our methodology, we observed a more gradual decline in precision to the point of 0.9 recall levels. This not only justified the overall precision of our methodology but also the high-performance certainty evaluation metric employed for recall adjustment, along with the improved modeling capabilities of OTNEL, due to its topical online training. The recall area (0.9, 1] of our method evaluation framed an elevated negative inclination as anticipated toward absolute recall. This was mainly interpreted as knowledge deficiency and a latent modeling approximation of deviating cases.

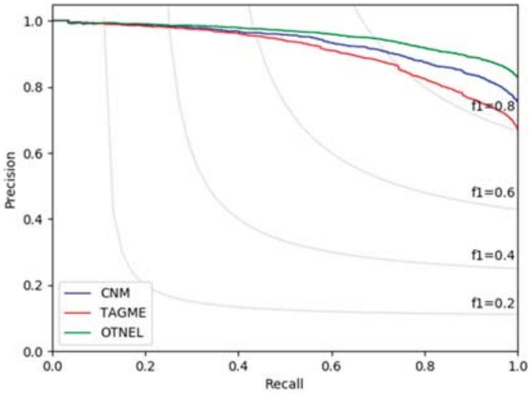


Figure 3. Precision–recall of OTNEL method, compared with Clauset–Newman–Moore (CNM) and TAGME baselines.

Table 1. OTNEL, TAGME, and CNM precision and F1 scores at varying recall levels.

Recall	1.0	0.9	0.8	0.6	0.4	0.2
CNM [1] (Precision)	0.7554	0.8362	0.8738	0.9337	0.9678	0.9866
CNM [1] (F1)	0.8606	0.8667	0.8351	0.7287	0.5659	0.3325
TAGME (Precision)	0.6720	0.7640	0.8242	0.9101	0.9619	0.9832
TAGME (F1)	0.8038	0.8264	0.8118	0.7222	0.5650	0.3323
OTNEL [2] (Precision)	0.8290	0.8897	0.9180	0.9589	0.9789	0.9896
OTNEL [2] (F1)	0.9065	0.8948	0.8548	0.7381	0.5678	0.3327

[1] WSD methodology based on Clauset-Newman-Moore Community detection; [2] Online Training Named Entity Linking.

Overall, our deep learning architecture consisted of a multilevel LSTM network. The recurrent learning selection was driven through a delayed reward with a global context within the topic confinement. Furthermore, the utilization of our penultimate layer score apparently yielded considerable insights into the success of certainty scoring, contributing to a progressive precision recall inclination. Again, we could observe high precision even at high recall over 0.9. For a dataset featuring such context variability, as training was conducted using Wikipedia, the extraordinary performance and potential of our approximation is profound.

The F1 score had a local maximum of approximately 91% of recall for the OTNEL model, as shown in Figure 4. The influence of topical segmentation introduced by our online training methodology, in conjunction with the high-performance indicator of linking certainty in the big data scale of the evaluation, emphasizes a consistently high performance, as clearly illustrated for the area over 0.8 of the recall axis. The value of our modeling approach is emphasized by the impressive accuracy even at high recall levels.

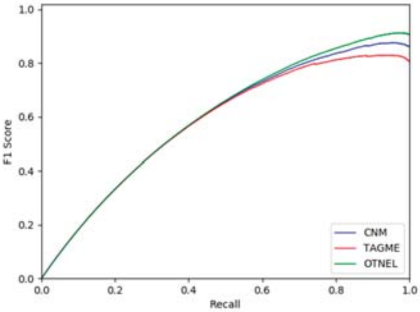


Figure 4. F1 score of OTNEL method, compared with CNM and TAGME baselines.

4.2. Quantification of Certainty Evaluation

Certainty was modeled as a measure of confidence for a mention selection. The correlation of certainty score and prediction score are outlined in the two-dimensional (2D) histograms in Figures 5 and 6. In Figure 5, we can observe a dense distribution of high certainty scores and a strong correlation of high prediction scores with high certainty. On the contrary, Figure 6 presents a less dense distribution of low certainty, in the areas of certainty below 0.6 and prediction score below 0.5. This intriguing observation can be interpreted as a knowledge deficit in the knowledge acquisition process, probably due to the coverage of our training set. Another reading of Figure 6 could delineate the presence of outliers; however, the apparent correlation of low certainty with low prediction score clearly indicates our model's advanced capabilities. At this point, it is worth noting that the analogy of valid (true positive) and invalid (false positive) entity link predictions was highly inclined toward true positives, as outlined on Figures 3 and 4, and the visualization of certainty and prediction scores validates our intuitions. Overall, the certainty metric performance as a measure of confidence for the validity of a linked entity was outstanding.

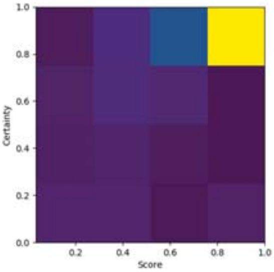


Figure 5. Prediction score–certainty score two-dimensional (2D) histogram: true positive distribution.

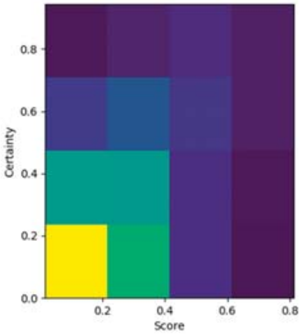


Figure 6. Prediction score–certainty score 2D histogram: false positive distribution.

5. Conclusions and Future Work

The current work proposed an innovative methodology featuring a deep learning classification model for the NED task, introducing the novel concept of online topical training for attaining high performance whilst maintaining rich semantic input specificity. This work introduced and studied the domain of online training NED. Moreover, to the best of our knowledge, this is the first approximation of Wikification and NED leveraging online topical training, introducing a stacked LSTM deep learning architecture model for the task. Our thorough experimental assessment revealed astounding performance in terms of precision, at moderate computational requirements, due to our simplicity-oriented dimensionality and modeling approach.

Our overall deep learning architecture permeates nonlinear input relation modeling, as the LSTM layers involved enable the exploitation of a dynamically changing contextual window over the input sequence history during the online topical training process. As a result, the use of a limited set of established features from works in the domain was adequate for attaining superior deep semantic inference capabilities with a topical focus, successfully addressing high-dimensional-space performance difficulties on a challenging task.

Among our plans for future enhancement of the current work's promising results, further analysis and experimentation in the quest for a more accurate architecture will be considered. A noteworthy advantage of the proposed neural network architecture is its understandability and neural network opacity via a simple model for delineating the benefits of the topical confinement concept in the online training NED task. As entity linking and NED tasks are based on knowledge, their underlying adversity is discerned in the absence of semantically linked corpora, namely, the knowledge acquisition bottleneck. An unsupervised machine learning knowledge expansion approximation could lead to more accurate results and, thus, knowledge acquisition closure from both structured and unstructured knowledge sources and corpora. The incorporation of an unstructured knowledge source via an ensemble learning approach for mitigating the impact of superinducing noise in the knowledge acquisition phase is among our plans.

In this article, our primary focus was the evaluation of new concepts for lowering the computational feasibility barrier to employing deep learning architectures in the NED task, while maintaining input semantic entropy by avoiding vast input transformations and granularity loss. Our extensive experimentation revealed propitious results, placing the introduced methodology in the limelight for further study and broad adoption.

Author Contributions: Conceptualization, C.M. and M.A.S.; methodology, M.A.S.; software, M.A.S.; validation, M.A.S.; formal analysis, M.A.S.; investigation, M.A.S.; resources, C.M.; data curation, M.A.S.; writing—original draft preparation, M.A.S.; writing—review and editing, C.M. and M.A.S.; visualization, M.A.S.; supervision, M.A.S.; project administration, C.M.; funding acquisition, C.M. All authors have read and agreed to the published version of the manuscript.

Funding: Christos Makris was co-financed by the European Union and the Greek national funds through the Regional Operational Program "Western Greece 2014–2020", under the Call "Regional Research and Innovation Strategies for Smart Specialization—RIS3 in Information and Communication Technologies" (project: 5038701 entitled "Reviews Manager: Hotel Reviews Intelligent Impact Assessment Platform").

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Khalid, M.A.; Jijkoun, V.; de Rijke, M. The impact of named entity normalization on information retrieval for question answering. In *Advances in Information Retrieval*; Macdonald, C., Ounis, I., Eds.; Springer: Berlin, Germany, 2008; Volume 4956, pp. 705–710.
2. Chang, A.X.; Valentin, I.S.; Christopher, D.M.; Eneko, A. A comparison of Named-Entity Disambiguation and Word Sense Disambiguation. In Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16), Portoroz, Slovenia, 23–28 May 2016; pp. 860–867.
3. Dorssers, F.; de Vries, A.P.; Alink, W. Ranking Triples using Entity Links in a Large Web Crawl—The Chicory Triple Scorer at WSDM Cup 2017. Available online: https://arxiv.org/abs/1712.08355 (accessed on 28 August 2020).
4. Artiles, J.; Amigó, E.; Gonzalo, J. The role of named entities in web people search. In Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, Singapore, 6–7 August 2009; Volume 2, pp. 534–542.
5. Blanco, R.; Ottaviano, G.; Meij, E. Fast and Space-Efficient Entity Linking for Queries. In Proceedings of the Eighth ACM International Conference on Web Search and Data Mining (WSDM'15), Shanghai, China, 31 January–6 February 2015; pp. 179–188.
6. Dietz, L.; Kotov, A.; Meij, E. Utilizing Knowledge Graphs in Text-centric Information Retrieval. In Proceedings of the Tenth ACM International Conference on Web Search and Data Mining (WSDM'17), Cambridge, UK, 6–10 February 2017; pp. 815–816.
7. Chair-Carterette, B.G.; Chair-Diaz, F.G.; Chair-Castillo, C.P.; Chair-Metzler, D.P. Entity linking and retrieval for semantic search. In Proceedings of the 7th ACM International Conference on Web Search and Data Mining (WSDM'14), New York, NY, USA, 24–28 February 2014; pp. 683–684.
8. Navigli, R. Word sense disambiguation. *ACM Comput. Surv.* **2009**, *41*, 1–69. [CrossRef]
9. Gale, W.A.; Church, K.W.; Yarowsky, D. A method for disambiguating word senses in a large corpus. *Lang. Resour. Eval.* **1992**, *26*, 415–439. [CrossRef]
10. Mihalcea, R.; Csomai, A. Wikify! Linking Documents to Encyclopedic Knowledge. In Proceedings of the Sixteenth ACM Conference on Conference on Information and Knowledge Management, Lisbon, Portugal, 6–10 November 2007; pp. 233–242.
11. Silviu, C. Large-Scale Named Entity Disambiguation Based on Wikipedia Data. In Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, Prague, Czech Republic, 28–30 June 2007; pp. 708–716.
12. Milne, D.N.; Witten, I.H. Learning to link with wikipedia. In Proceedings of the 17th ACM Conference on Information and Knowledge Management (CIKM'08), Hong Kong, China, 2–6 November 2008; pp. 509–518.
13. Milne, D.; Witten, I.H. An Effective, Low-Cost Measure of Semantic Relatedness obtained from Wikipedia Links. In Proceedings of the AAAI Workshop on Wikipedia and Artificial Intelligence (WIKIAI), Chicago, IL, USA, 13 July 2008; pp. 25–30.
14. Sayali, K.; Amit, S.; Ganesh, R.; Soumen, C. Collective annotation of Wikipedia entities in web text. In Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'09), Paris, France, 28 June–1 July 2009; pp. 457–466.

15. Paolo, F.; Ugo, S. TAGME: On-the-fly annotation of short text fragments (by wikipedia entities). In Proceedings of the 19th ACM International Conference on Information and Knowledge Management (CIKM'10), Toronto, Canada, 26–30 October 2010; pp. 1625–1628.
16. Johannes, H.; Mohamed, A.Y.; Ilaria, B.; Hagen, F.; Manfred, P.; Marc, S.; Bilyana, T.; Stefan, T.; Gerhard, W. Robust Disambiguation of Named Entities in Text. In Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing (EMNLP 2011), Edinburgh, UK, 27–31 July 2011; pp. 782–792.
17. Han, X.; Sun, L.; Zhao, J. Collective entity linking in web text: A graph-based method. In Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'11), Beijing, China, 25–29 July 2011; pp. 765–774.
18. Makris, C.; Simos, M.A. Novel Techniques for Text Annotation with Wikipedia Entities. In Proceedings of the Artificial Intelligence Applications and Innovations Evaluation—AIAI 2014, Rhodes, Greece, 19–21 September 2014.
19. Ricardo, U.; Axel-Cyrille, N.N.; Michael, R.; Daniel, G.; Sandro, A.C.; Sören, A.; Andreas, B. AGDISTIS—Agnostic Disambiguation of Named Entities Using Linked Open Data. In Proceedings of the Twenty-first European Conference on Artificial Intelligence, Prague, Czech Republic, 18–24 August 2014; pp. 1113–1114.
20. Piccinno, F.; Ferragina, P. From TagME to WAT: A new entity annotator. In Proceedings of the First International Workshop on Entity Recognition & Disambiguation (ERD'14), Gold Coast, Queensland, Australia, 11 July 2014; pp. 55–62.
21. Sun, Y.; Lin, L.; Tang, D.; Yang, N.; Ji, Z.; Wang, X. Modeling mention, context and entity with neural networks for entity disambiguation. In Proceedings of the 24th International Conference on Artificial Intelligence (IJCAI'15), Buenos Aires, Argentina, 25–31 July 2015; pp. 1333–1339.
22. Ikuya, Y.; Hiroyuki, S.; Hideaki, T.; Yoshiyasu, T. Joint Learning of the Embedding of Words and Entities for Named Entity Disambiguation. In Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning, Berlin, Germany, 11–12 August 2016; pp. 250–259.
23. Ganea, O.-E.; Hofmann, T. Deep joint entity disambiguation with local neural attention. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, Copenhagen, Denmark, 7–11 September 2017.
24. Ivan, T.; Phong, L. Improving Entity Linking by Modeling Latent Relations between Mentions. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, Melbourne, Australia, 15–20 July 2018; Volume 1, pp. 1595–1604.
25. Priya, R.; Partha, T.; Vasudeva, V. ELDEN: Improved entity linking using densified knowledge graphs. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, New Orleans, LA, USA, 1–6 June 2018; Volume 1, pp. 1844–1853.
26. Fang, Z.; Cao, Y.; Li, Q.; Zhang, D.; Zhang, Z.; Liu, Y. Joint Entity Linking with Deep Reinforcement Learning. In Proceedings of the World Wide Web Conference (WWW'19), San Francisco, CA, USA, 13–17 May 2019; pp. 438–447.
27. Avirup, S.; Gourab, K.; Radu, F.; Wael, H. Neural Cross-Lingual Entity Linking. In Proceedings of The Thirty-Second AAAI Conference on Artificial Intelligence (AAAI-18), New Orleans, LA, USA, 2–7 February 2018; pp. 5464–5472.
28. Ilya, S.; Liat, E.-D.; Yosi, M.; Alon, H.; Benjamin, S.; Artem, S.; Yoav, K.; Dafna, S.; Ranit, A.; Noam, S. Fast End-to-End Wikification. Available online: https://arxiv.org/abs/1908.06785 (accessed on 28 August 2020).
29. Wikimedia Update Feed Service. Available online: https://meta.wikimedia.org/wiki/Wikimedia_update_feed_service (accessed on 28 August 2020).
30. Keras: The Python Deep Learning API. Available online: https://keras.io (accessed on 28 August 2020).
31. Abadi, M.; Barham, P.; Chen, J.; Chen, Z.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Irving, G.; Isard, M.; et al. TensorFlow: A system for large-scale machine learning. In Proceedings of the 12th USENIX conference on Operating Systems Design and Implementation (OSDI'16), Savannah, GA, USA, 2–4 November 2016; pp. 265–283.
32. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef] [PubMed]

33. Farzad, A.; Mashayekhi, H.; Hassanpour, H. A comparative performance analysis of different activation functions in LSTM networks for classification. *Neural Comput. Appl.* **2017**, *31*, 2507–2521. [CrossRef]
34. Christos, M.; Georgios, P.; Michael, A.S. Text Semantic Annotation: A Distributed Methodology Based on Community Coherence. *Algorithms* **2020**, *13*, 160. [CrossRef]
35. Index of /Enwiki/. Available online: https://dumps.wikimedia.org/enwiki (accessed on 28 August 2020).
36. Specs/wikitext/1.0.0 MediaWiki. Available online: https://www.mediawiki.org/wiki/Specs/wikitext/1.0.0 (accessed on 28 August 2020).
37. Matei, Z.; Mosharaf, C.; Michael, J.F.; Scott, S.; Ion, S. Spark: Cluster computing with working sets. In Proceedings of the 2nd USENIX Conference on Hot Topics in Cloud Computing (HotCloud'10), Boston, MA, USA, 22 June 2010.

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

big data and cognitive computing

MDPI

Article

ParlTech: Transformation Framework for the Digital Parliament

Dimitris Koryzis [1], Apostolos Dalas [2], Dimitris Spiliotopoulos [3] and Fotios Fitsilis [4,*]

[1] Strategic Planning and Administrative Functions Re-Engineering Unit, Hellenic Parliament, 10671 Athens, Greece; dkoryzis@parliament.gr
[2] Hellenic OCR Team, 15343 Agia Paraskevi, Greece; info@hellenicocrteam.gr
[3] Department of Management Science and Technology, University of the Peloponnese, 22100 Tripoli, Greece; dspiliot@uop.gr
[4] Scientific Service, Hellenic Parliament, 10671 Athens, Greece
* Correspondence: fitsilisf@parliament.gr

Abstract: Societies are entering the age of technological disruption, which also impacts governance institutions such as parliamentary organizations. Thus, parliaments need to adjust swiftly by incorporating innovative methods into their organizational culture and novel technologies into their working procedures. Inter-Parliamentary Union World e-Parliament Reports capture digital transformation trends towards open data production, standardized and knowledge-driven business processes, and the implementation of inclusive and participatory schemes. Nevertheless, there is still a limited consensus on how these trends will materialize into specific tools, products, and services, with added value for parliamentary and societal stakeholders. This article outlines the rapid evolution of the digital parliament from the user perspective. In doing so, it describes a transformational framework based on the evaluation of empirical data by an expert survey of parliamentarians and parliamentary administrators. Basic sets of tools and technologies that are perceived as vital for future parliamentary use by intra-parliamentary stakeholders, such as systems and processes for information and knowledge sharing, are analyzed. Moreover, boundary conditions for development and implementation of parliamentary technologies are set and highlighted. Concluding recommendations regarding the expected investments, interdisciplinary research, and cross-sector collaboration within the defined framework are presented.

Keywords: digital parliament; digital transformation; legal tech; disruptive technologies; technology framework; parliamentary administrators; ParlTech; knowledge-driven processes; parliamentary hype cycle; semantic web

Citation: Koryzis, D.; Dalas, A.; Spiliotopoulos, D.; Fitsilis, F. ParlTech: Transformation Framework for the Digital Parliament. *Big Data Cogn. Comput.* **2021**, *5*, 15. https://doi.org/10.3390/bdcc5010015

Received: 13 February 2021
Accepted: 11 March 2021
Published: 15 March 2021

1. Introduction

Organizations such as parliaments are complex systems that can be considered an ensemble of five different elements: Process, people, culture, structure, and information systems [1]. These entail the need for an organizational transformation framework that exploits the potential of information communication technology (ICT) [2]. Over the past two decades, the evolutionary use of workplace technologies in organizations has hybridized their use with human activities [3], forming a more complex environment [4] and an emergent human-AI hybrid digital assistant [5] or meta-human configurations as new forms of socio-technical systems [6]. ICT has the potential to impact all of these elements and involves the emergence of several digital/human configurations [3], reflecting the assembly of digital features with human intent and their performance within a complex organization, as in the case of parliaments.

However, even if the demand on ICT to design and implement changes within the parliamentary institution has been documented in previous decades [7,8], it is still unclear how and under which conditions this digital transformation takes place [9]. Within governance, in particular, ICT was found to skew the balance towards efficiency rather than innovation, despite organizations expressing a need towards the latter [10]. Therefore, a two-step ICT-embedded organizational transformation, i.e., technical and social, can be hypothesized. Mergel et al. [11] (p. 12) go a step further and use the term digital transformation "to emphasise the cultural, organizational, and relational changes . . . to differentiate better between different forms of outcomes".

The introduction of ICT is often combined with the transformation of an organization as a whole. Naturally, the technical elements of an organization, e.g., data and information systems, are impacted most. For this reason, in relation to digitalization, researchers have called for "digital ambidexterity", which is the capability to dynamically balance the digital initiatives in terms of efficiency and innovation [12]. In contrast, the social system (culture and structure) appears to be less affected by digital transformation [13]. Generally, digital innovation for value creation in organizations such as parliaments unbundles and recombines linkages among existing resources or simply generates new ones. In situations like these, when changes are radical, digital disruption may emerge, with considerable effects for different actors [14].

Regarding parliaments, in recent years, a small number of studies have investigated their role as organizations that are managing new technologies [8,15]. Nonetheless, the e-Parliament concept is not new [16]. Since the early 2000s, several attempts, projects, and concepts have indicated that citizens can, and in fact should, be included and engaged in decision-making processes through tools, products, platforms, and integrated IT services that enable them to actively participate in interaction with policy makers [17]. Indeed, it appears that the use of digital technologies by traditional organizations such as parliaments is highly diverse, albeit existing studies mostly refer to a variety of tools that allow for the engagement of citizens [18]. It must be noted though that such concepts are still far from the manifestation of parliament as a digital democracy hub for online engagement, communication, cooperation, and interaction among citizens and legislators. Such a digital collaborative platform that is operated in a transparent manner could be a useful tool, particularly within the legislative process.

To date, little attention has been placed on the development of a theoretical framework for the transformation of a traditional public organization, such as parliament, into a modern digitally ambidextrous organization, because there are several pathologies [10]. A basic approach has been made with the technology acceptance model framework by Davis [19], which is currently in version 3 (TAM3). This is premised on the theory that the model helps explain a specific behavior, which in this case is usage, ease of use, or perceived usefulness, towards a specific target, using technology, and within a specific context (e.g., public administration, parliaments). Scarce literature is complemented by a small number of published digital strategies in parliaments that attempt to incorporate organizational transformation elements with digital technologies in a layered structure.

Ongoing development, e.g., within the ISA^2 European Interoperability Framework, is providing guidance towards interoperable digital public services [20]. Effectively, system architecture is leading to an ecosystem of tools and services with accurately defined functions and interfaces. Within this multi-stakeholder environment, a user-centric approach is favored, using agile and lean ICT methodologies for interoperable and secure systems that constitute legal data hubs, which are accessible and inclusive for all stakeholders. Nonetheless, for the parliamentary workspace, more than a simple platform with integrated tools and software applications is needed, especially in the policy formulation stage, where a large number of users, e.g., parliamentary actors and/or stakeholders, are typically involved. However, state-of-the-art intuitive integrated tools of the likes of e-participation services, social media campaigns, visualizations, and linguistic analysis have the potential to advance digital transformation of the policy cycle [21].

The emergence of disruptive technologies might complicate a linear evolutionary approach for the digital parliament. At the same time, they have the potential to strengthen parliamentary institutions and bridge the informational and processing gap towards the executive. Taking into consideration digital tools and solutions for the digital evolution of parliaments [15,17], comparative reports for aspects of future parliaments [22], the World e-Parliament Report 2018 [23], and the evolution of digital technologies from the 2020 Gartner Hype Cycle for Emerging Technologies [24], this study describes a novel approach to the digital transformation of parliamentary organizations, both from a holistic perspective and the user's point of view. It does so by refining existing digital parliament concepts and discussing organizational vis-à-vis digital transformation. Moreover, an innovative digital framework is developed bottom-up using the findings from a survey of parliamentary experts, who constitute users of parliaments' digital systems. Finally, based on acknowledged technology trends, the definition of a "parliamentary hype cycle" identifies promising technologies of parliamentary relevance that could shape future e-Parliament systems.

The next part (Section 2) defines the methodology of research and the approach taken to create the survey on which the study is based, as well as the selection of a representative sample of intra-parliamentary actors. It is followed by Section 3, where the main findings are shown and discussed. These are used to define a framework for the digital parliament (Section 4), based on which a concise discussion of the most promising technologies is made based on the survey key findings (Section 5). The article concludes with the most interesting aspects in a parliamentary context and a brief outlook for further research (Section 6).

2. Approach and Methodology

The authors have opted to use a user-initiated approach to define the framework of a digital parliament. To obtain data related to the nature and attributes of the framework, a structured expert survey has been developed and sent to a carefully selected set of parliamentary actors/users/stakeholders [25]. An expert survey is preferable, since the object of scholarly inquiry is novel and complex, yet it directly affects the users as actors and actuators. Therefore, it is "more likely to find reliable information in experts' judgements rather than in documentary sources" [26] (p. 274). In expert surveys, i.e., special and limited populations, the sample size is small by design, and no representative sampling framework is required. Instead, for this study, purposeful sampling was utilized for data collection and their predominantly qualitative interpretation [27]. The main criteria, according to which the subjects have been selected as parliamentary experts, were: Expertise in parliamentary development, scholarly engagement, and international cooperation. Further selection criteria were applied to ensure the geographic and gender diversity of the sample.

The survey builds upon IPU's definition of the digital parliament, the drivers and barriers for its digital and organizational transformation [13], and Gartner's hype cycle [24]. They were used to create a set of questions designed to capture the user's perception of the digital parliament. The resulting survey contains 15 questions, which can be divided in five basic blocks:

1. Demographics (country, sector, scientific background).
2. Digitalization process (level, transformation, priorities, relevance).
3. Barriers and drivers of transformation (organizational, digital).
4. E-parliament trends (significance and importance).
5. Emerging digital technologies in parliaments (applicability, maturity, usefulness, and sustainability).

Next to general demographics questions, block 2 attempts to redefine the digital parliament. The perceived level of digitalization from the users' point of view is measured and linked to the organizational transformation. Priorities and themes of relevance are captured. The barriers and drivers of organizational transformation and their link with digital transformation is assessed within block 3. Having the 2018 IPU World e-Parliament report [23] as point of reference, block 4 then re-defines trends and key aspects of the digital parliament and introduces tools and services in the parliamentary context. The final block of questions (block 5) estimates the applicability, maturity, usefulness, and sustainability of digital emerging technologies. On the detailed definition of these user experience (UX) terms, see [28,29].

The questions were carefully designed to facilitate the understanding of the parameters and the building blocks of an evolutionary framework for the digital parliament. Both language and terminology were adapted to the parliamentary context. Technology foresight, especially for niche parliamentary technology, or ParlTech, is a particularly difficult task. ParlTech goes beyond what is considered state-of- the-art and is based on emerging technologies that fully or partially automate or even advance processes of parliamentary nature. As such, it is to be differentiated from standard technology aiming to provide solutions to administrative/organizational issues.

The technologies that have been selected to be included in the survey, which eventually led to the creation of a parliamentary hype cycle, have been extracted from 2020 Gartner hype cycles (emerging technologies, legal and compliance technologies, and internet of things) and constitute direct projections of emerging technologies in the parliamentary workspace. In the course of an internal workshop, 13 specific technologies have been identified as promising ParlTech and included in the survey. Table 1 matches the technologies from Gartner hype cycles with the ones that are relevant for parliaments.

Table 1. Technologies for the parliamentary workspace.

#	Technology in Gartner	Corresponding ParlTech [1]
1	Adaptive ML	Recommender systems
2	AI assisted design	AI-assisted legal drafting/policy making (*Legal AI*)
3	AI augmented development	*Virtual parliament*
4	Authenticated provenance	Smart contracts, *smart legislation*
5	Bring Your Own Identity	Identity as a Service for parliament apps (*IDaaS*)
6	Citizen twin	*Digital twin* of parliamentary infrastructure
7	Composable Enterprise	*Rapid digital* and operational *transformation*
8	Decentralized (semantic) web	*Linked open data* and advanced legal services
9	Embedded AI	*Machine learning* solutions
10	Internet of Things (Services)	*Internet of Parliamentary Things*
11	Legal & compliance analytics	*Interoperability solutions*, integrated tools & services (legal informatics)
12	Ontologies and Graphs	*Ontological representation* of parliamentary entities and procedures
13	Social data	*Social media analytics*

[1] Short names appear in italics.

In the course of the article, the authors attempt to approach three particular research goals around the digital parliament:

- To redefine the main factors of digital transformation in parliament.
- To explore the possibility to create a parliamentary hype cycle.
- To specify the challenges and preconditions of an evolutionary framework.

The above survey design methodology constitutes a valid instrument to evaluate responses on the prerequisites and conditions of the digital parliament. While quantitative data have been collected (i.e., a Likert scale is used for quantitative evaluation), focus is placed on the qualitative evaluation of findings, in order to come up with a tangible approach for a digital parliament framework.

The survey has been sent to 53 MPs and parliamentary professionals, collectively referred to as parliamentary experts, covering 36 countries. A total of 32 persons from 25 countries responded, a response rate of 60.4%, which the authors consider particularly high, given the complexity of the survey. The high response rate may be also an indication that these usually busy, high-level parliamentary experts considered the survey favorably. The responders originate from 25 different parliaments, which means that some parliaments are represented by more than one expert. For methodological reasons, even in countries with bicameral systems, experts were selected from a single chamber. Hence, the number of parliaments coincides with the number of their countries of origin. As a result, in the context of the study, the terms country and parliament can be used interchangeably. Based on the original survey design, a wide geographic distribution across five continents can be observed. The findings are comparable across countries due to the common criteria used for expert selection, i.e., parliamentary experts or MPs who meet the above conditions are more likely to provide comparable information on the digital parliament and its development than random parliamentary professionals. A significant part of survey respondents (around one third, i.e., 31.2%) are female. Basic sample demographics are presented in Table A1 (Appendix A).

Upon request, the participants received online support and technical guidance to complete the survey. Most of the queries referred to the last survey block related to emerging digital technologies. This was anticipated as, unsurprisingly for parliamentary experts, a dominant majority of the respondents have a social science background, i.e., more than one third owns a degree in law, with only 15.6% having a degree in informatics or engineering. The experts work in different sectors of parliament, e.g., in parliamentary committees, library and/or research service, and international relations. The broad distribution in parliamentary sectors is important because it provides for a holistic approach to the research topic.

Processing and presentation of the findings ensured anonymity and confidentiality of the individual contributions. The survey, as well as the raw data set, has been placed on an open platform (Figshare) for cross-analysis and further elaboration [30]. The survey results have been assessed for reliability using the Cronbach coefficient (α) for each of the blocks of questions, i.e., $\alpha = 0.88$ for block 2, $\alpha = 0.76$ for block 3, $\alpha = 0.89$ for block 4, and $\alpha = 0.95$ for block 5.

3. Findings and Evaluation

Different blocks in the survey cater for gradual approximation of a digital parliament framework, starting with the perception of digitalization. There is a significant number of participants (46.9%) stating that the level of digitalization of their parliament is high or very high, while 37.5% rate it as moderate. The resulting average score in the seven-point Likert scale is 4.37 with a standard deviation of 1.31, i.e., 4.37 ($\sigma = 1.31$), and gives an overall positive view of the level of digitalization in parliament. The extraordinary high values, i.e., approximately 85%, show that the users report that the level of digitalization is at least moderate. This could be a temporal effect, and can be partially explained through the overall positive effects of the pandemic to the digitalization of parliaments [31]. Nevertheless, it can be considered as a strong foundation for further digitalization efforts. Moreover, this overall positive perception allows the authors to assume that the subjects also have the necessary technological affinity to assess the fitness of a broad list of technologies in the parliamentary workspace.

From the organizational perspective, findings show that digitalization has mostly transformed processes (78.1%), data (75%), people (65.6%), and systems (62.5%), with similar Likert scores (1–5 scale): Data is 3.94 ($\sigma = 0.88$), processes is 3.69 ($\sigma = 0.86$), people is 3.66 (0.83), and systems is 3.88 ($\sigma = 0.98$). The widespread perception that ongoing digitization is transforming data, information systems, and processes is not unexpected. It has already been the outcome of existing investigations (Tangi et al., 2020). However, one needs to consider whether the measured lower values in the digitalization effect on structure and culture, 3.31 ($\sigma = 0.93$) and 3.25 ($\sigma = 0.80$), respectively, are attributed to a certain cause. Regarding the current progress of the parliaments from digitalization, the aspects that the progress applies to (processes, people, culture, structure, systems, and data) were all found to be non-independent, when examined in pairs (chi-square, $p < 0.001$, for all pairs). The authors believe that these parameters are still decoupled from the effects of digitalization, hence the observed difference. As a matter of fact, overall high acceptance values and inter-dependence seem to confirm that digitalization tends to holistically affect parliamentary organizations.

At the same time, findings show that the organizational transformation process that goes along with digitalization is significantly hindered by a number of factors, the most recognized being bureaucratic culture (65.6%) and resistance to change (62.5%). Likert (1–5) scores for bureaucratic culture and resistance to change are 3.63 ($\sigma = 1.03$) and 3.53 ($\sigma = 0.95$), respectively, which is similar to earlier findings [13]. This is an interesting result that is linked to the wider perception of parliaments as "traditional" organizations. The fact that digital transformation efforts have been acknowledged, be it as a response to the COVID-19 pandemic or not, shows that even high intrinsic barriers can be overcome, given the proper incentive or when reaching out to a greater objective. It is worth mentioning that experts are differentiated when it comes to roadmaps and planning, i.e., "only" 50% agree with the statement, with 3.50 ($\sigma = 1.02$), a result that can be associated with findings from the 2018 IPU report [23]. This partially interprets the observed lack of digital strategies in parliaments. Moreover, the survey participants reported that the fear of innovation was the most serious condition that affected the level of digitalization of their parliament (Spearman's Rank Correlation, $p = 0.021$, $\rho(30) = 0.406$). In the context of the institutional future, the greater objective is no less than to correspond to a digital societal shift while maintaining the institutional equilibrium.

Two thirds expressed that the organizational transformation process that goes along with digitalization is pushed by expected benefits for the main stakeholders, i.e., 3.72 ($\sigma = 0.81$) in Likert scale (1–5), and strong top-down leadership, i.e., 3.47 ($\sigma = 0.88$). Both are expected drivers, as the effects of benefits and incentives in public service are well documented (for a systematic review of the relevant literature, see [32]), as well as the positive impact of leadership [33].

Furthermore, parliamentary experts were asked to assess a series of digital trends and aspects from the 2018 IPU World e-Parliament Report [23]. Two years after the IPU report, the user evaluation can reveal an understanding of the transformation of former trends in today's tangible systems and processes within parliaments. Additionally, it can serve as a qualitative indicator for the validity of evaluation of emerging ParlTech.

A huge majority of the experts (87.5%) perceive open and transparent legal data, as well as openness, accountability, and accessibility, as significant components for digital parliament. However, the exact degree of correlation between the production of legal data, for instance Big Open Legal Data (BOLD), and an increase in institutional accountability is unclear, and almost certainly depends on the individual organization. Yet, this finding is in-line with recent developments in legal informatics and the development of legal documents standards that are utilized by dedicated legislative drafting tools [34,35].

When recording priorities for the digital parliament, for most of the occasions (>90%), processes, data, and people are the experts' preferences. System architecture is a high priority for roughly two thirds (65%) of the experts. For all of them, high Likert (1–5) scores (>4.3) are recorded, where data display extremely high Likert values 4.47 ($\sigma = 0.57$). Furthermore, the identification of people as a priority is relevant to society representation, openness, inclusiveness, accessibility, accountability, communication, and engagement with citizens. At the same time, process is relevant with accountability, and system architecture is relevant with business process collaboration. Notably, these priorities coincide with preferable components of digitalization from an organizational perspective.

When describing the use of digital tools, services, and products in a parliament, the experts highly favored accessibility and openness (87.5%) as well as communication with citizens (84.3%) as relevant attributes. These preferences are highly ranked in (L)ikert (1–5), i.e., $L > 4.2$, with $0.68 < \sigma < 0.90$. While these results are in-line with the aforementioned findings regarding digital parliament as a whole, it is worth highlighting the interaction between citizens and systems, through careful and efficient design and implementation of digital components, tools, products, services [21]. These results confirm once more the IPU suggestions for an open, accessible digital parliament that communicates interactively with citizens.

The 2018 IPU report indicated that digital broadcasting and video streaming will gradually overtake traditional broadcasting, a finding that is supported by the majority of the experts from this study (78.1%). Other important IPU trends, such as inter-parliamentary support and political commitment to use digital technologies, are also confirmed by the present survey. Additional enabling factors, such as training and skills, earned similar high scores. Acquiring new digital skills is deemed necessary for public administrators to be able to participate in the design and operation of ParlTech. For this, novel training approaches are necessary that may involve national schools of government [36] and/or more unified schemes, such as the Interoperability Academy in the framework of the European interoperability framework.

When using digital tools, knowledge of how parliaments work seems to be a high barrier for greater citizen engagement for 68.7% of the respondents. Citizen engagement can be facilitated by parliaments through the use of social media (L = 3.50, σ = 0.92). One could derive that parliaments use social media mainly to report on parliamentary business, interacting with citizens only marginally [37]. Even further, there are several attempts to use innovative ICT tools for social media analysis, without limited impact [38,39]. Table A2 (Appendix B) presents the aggregated results of the above study parameters in the form of average scores on the five-point Likert scale (L), along with the respective standard deviation (σ).

The use of disruptive technologies derived from Gartner hype cycle for Emerging Technologies constitutes a pragmatic approach to define a first set of applicable technologies for parliamentary use. This has been demonstrated already for the broader e-governance sector [40]. The majority of experts identified linked open data and advanced legal services as the most promising technologies (59.3%), immediately followed by social media analytics and the virtual parliament (53.1%). Linked open data, when efficiently produced and distributed, is certainly a direct manifestation of the broader call for institutional openness and can lead, as seen above, to increased accountability of parliamentary actors. Virtual parliament is not a single, but rather a combination of technologies around virtual, augmented, mixed, and extended reality [41] and can be associated with widespread hype around these technologies. Nonetheless, one should not underestimate the marketing-effect in relation to the introduction of such technologies. An adequate marketing wrap could be an efficient passport into the parliamentary sphere for the discussed technologies.

On the other hand, it stands out that a significant number of the questioned users do not identify machine learning solutions and artificial intelligence (AI)-assisted legal drafting and policy making as particularly relevant for parliaments. In recent years, significant applications of AI technologies have found their way into governance. In particular, machine learning, as an expression of AI, is considered a mature technology with broad applications in GovTech (Government Technology) [42], albeit future utilization needs to be based upon responsiveness, efficiency, and fairness [43]. Thus, negative opinions may be related with technological maturity or the lack of relevant pilot/demonstrator applications. Indeed, survey users rated AI as well as blockchain-assisted technologies as less mature than others.

Regarding usefulness of technologies, virtual parliament, linked open data, and advanced legal services stand out for 68.7% of the experts. Additionally, social media analytics (59.3%) and rapid digital and operational transformation (53.1%) seem to be rather useful. Digital Twins represent "digital replications of living as well as non-living entities that enable data to be seamlessly transmitted between the physical and virtual worlds" [44] (p. 87). In parliaments, the concept, boosted by machine intelligence and cloud computing, could be used to monitor and optimize institutional functions and operations.

However, less than a third (31.2%) of the experts do not perceive the usefulness of digital twin infrastructure.

Sustainability of these technologies is a central issue. After all, lack thereof would be a major indicator to question investment in technologies below a certain threshold. Regarding sustainability of technologies that the users indicated as useful, almost all experts (96.9%) stated that these will provide added value to professional parliamentary work, while roughly eight out of ten (78.1%) believe that these will help provide usable and more interesting services to strengthen the democratic appreciation of citizens. Special mention is deserved for the option for empowerment of civic stakeholders favored by 71.9% of the experts, which practically confirms the finding that digital communication, e.g., through social media, can potentially re-link citizens and parliaments.

4. Parliamentary Hype Cycle

The findings from the evaluation of the maturity, usefulness, and applicability parameters of emerging technologies were used to develop a parliamentary hype cycle that is based on the Gartner hype cycle concept [45,46]. Conceptually, the Gartner hype cycle depicts the expectations hype for new and emerging technologies versus time until they are adopted and have passed on to production. Based on the original Gartner plot, the following assumptions were made to assess the necessary parameters and create a respective chart for ParlTech:

- Maturity was matched to the Time parameter.
- Usefulness was matched to the Expectations parameter.
- Applicability was matched to the time scale that a technology is expected to reach the productivity plateau.

For the technologies as per Table 1 (short names are used; classified from lower to higher maturity), Table 2 shows the mean Likert scores (1–5) for these three parameters. The methodology was to create an XY chart of Maturity (Time) versus Usefulness (Expectations), with references to distinct stages of the hype cycle as defined by Gartner. Similar to the original plot, a third dimension (time to productivity plateau) was added for each technology data point via color code. Analysis of survey results led to the definition of two basic time frames for the parliamentary hype cycle:

- Mean Likert (1–5) score L $\geq$ 3.00: Medium to high applicability.
- Mean Likert (1–5) score L < 3.00: Low to medium applicability.

Table 2. Maturity, usefulness, and applicability of ParlTech.

ParlTech [1]	(M)aturity [Mean L(1–5)]	(U)sefulness [Mean L(1–5)]	(A)pplicability [Mean L(1–5)]
Ontological representation	2.72	3.00	2.84
Legal AI	2.75	3.34	3.09
Smart legislation	2.75	3.22	3.19
Recommender systems	2.78	2.84	2.84
Digital twin	2.81	2.81	2.97
Machine learning solutions	2.94	3.31	2.97
Internet of Parliamentary Things	2.94	3.31	3.09
IDaaS	3.00	3.31	3.28
Rapid digital transformation	3.09	3.59	3.31
Interoperability solutions	3.22	3.59	3.50
Linked open data	3.25	3.81	3.56
Virtual parliament	3.31	3.72	3.41
Social media analytics	3.47	3.53	3.34

[1] Standard deviation, M: 0.75 $\leq \sigma \leq$ 1.02; U: 0.85 $\leq \sigma \leq$ 1.06; A: 0.85 $\leq \sigma \leq$ 1.13.

The chart depicts Maturity (X-axis) versus Usefulness (Y-axis), and it was based on their mean Likert (1–5) values (see Figure 1). The 'noisy' early part, attributed to the overall low grading of the maturity parameter, has been normalized, an offset has been added, and the slope of the curve has been exaggerated to match the characteristic Gartner hype cycle form. Consequently, it results in a qualitative plot which depicts technology hype as perceived by the experts. Three characteristic stages of the Gartner plot, already in this form, are visible. The sharp rising part of the curve matches the "innovation trigger" followed by the "peak of inflated expectations", i.e., the highest point in the curve. The curve then enters the decreasing slope of the "trough of disillusionment".

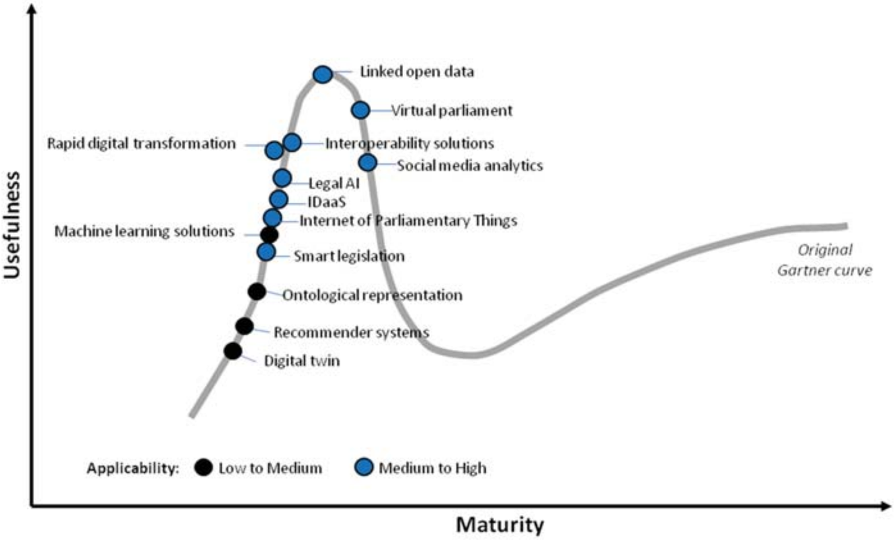


Figure 1. ParlTech hype cycle for year 2020.

Visibly, most ParlTech finds itself on the "innovation trigger" (potential breakthrough that might kick things off). It is noted that technology early in the hype cycle is perceived to be less applicable compared to technologies higher in the cycle. Digital twins, recommender systems, and ontological representation belong to this category. At the "peak of inflated expectations", one finds linked open data and advanced legal services. This is the technology that enjoys the biggest hype, yet it is perceived to not be mature enough for entering production status. While moving further right on the maturity axis, but maybe still within the limits of the "peak of inflated expectations", the virtual parliament along with social media analytics can be found. It can be observed that a similarity between Gartner and the parliamentary hype cycle lies in the fact that most technologies are located in the first two stages of the curve, where excitement and expectations are high. On the other hand, in contrast to Gartner, the here referenced ParlTech appears not to have reached the "trough of disillusionment" stage. In general, ParlTech is found to be delayed in terms of maturity and expectations compared to Gartner emerging technologies.

It is, however, worth noting differences when considering responses from digitally advanced parliaments (based on responses for the level of digitization with mean L(1–7) $\geq$ 5.50), namely Austria, Brazil, France, Israel, Libya, and Spain. Overall, higher maturity and usefulness scores are reached for respective technologies. All applicability scores are in the higher tier. Finally, specific ParlTech like digital twins seem to receive considerably higher scores. The assessment of survey results, combined with existing knowledge from previous studies, offer significant insights for the development of a digital parliament framework.

5. The Digital Parliament Transformation Framework

Considering all the above, a broad framework for digital parliament can be set-up. This framework will give the opportunity for parliamentary organizations to level the path towards advanced digital transformation stages. There have been earlier attempts to define such frameworks from other perspectives, e.g., in the form of organizational restructuring concepts or digital national plans. These frequently and solely rely on elaborated parliamentary strategies, as in the cases of the UK, Australia, Greece, and France. These attempts again have led to specific operational plans and actions within the digital environment [47,48].

Matt et al. [49] presented five general principles, i.e., strategy, operations, functions, technologies, and transformation, upon which such a framework can be developed. Gimpel et al. [50] provide six fields of actions for digitalization, i.e., user, data, value, organization, operations, and transformation. Additionally, Nwaiwu [51] compared 10 conceptual and theoretical frameworks for digital transformation, which primarily deal with organizational issues rather than user behavioral aspects and technological adoption. Nwaiwu [51] concludes that the parameters to be considered when choosing a model for digital transformation are corporate strategy, vision, and mission.

In the light of the above, in a balanced act between strategy and technology, a robust yet adjustable structure for a parliamentary digital framework is defined. The framework consists of four distinct components that roughly correlate to the principles by Matt et al. [49] when unifying functions with operations. This becomes possible because in legislatures, parliamentary functions closely match primary working processes. Hence, the following components may constitute the basis of an advanced digital framework for parliament:

1. Strategy: An integrated strategy with a clear definition of a digital parliament vision and goals.
2. Operations: Digitalization of parliamentary operations.
3. Technology: Adaption to emerging technologies for digital growth using the parliamentary hype cycle.
4. Digital transformation: Develop and align the enablers of digital transformation.

An integrated parliamentary digital strategy is the main pillar of this framework that contains the organization vision, values, scope, and goals, with a clear definition of digitalization in the parliamentary context (e.g., openness, transparency, accountability, and societal representation). Significant attributes of the latter are provided in the evaluation part of this article with high correlation with people (users) as priorities of digitalization. The next step is an operational stage that is related to identification and planning of digitalization actions. Here, as highlighted by parliamentary experts, actions that are related to inclusive governance could be prioritized. These could include, for instance, parliamentary functions that strengthen citizen’s engagement. Emerging technologies, as an expression of digital evolution, constitute a natural compound of any digital framework. Survey findings led to the creation of a parliamentary hype cycle, based on state-of-the-art and disruptive technologies adapted to parliamentary context.

Parliaments, depending on their level of digitalization and willingness for innovation, could screen the hype cycle to determine technologies appropriate for further utilization. An overview of necessary digital (and organizational) transformation enablers is suggested above and includes, among others, strong leadership, digital skills, and potential benefits for users. Figure 2 presents the proposed Digital Parliament Transformation Framework, based on the reported priorities (people, culture, structure, data, processes, systems) and the identified attributes that the ParlTech adoption is expected to enhance. However, there are a series of boundary conditions under which this framework can be useful for parliament. For instance, there might be an overlap with existing digital strategies or commitments, as is the case of Open Government Partnership. In such cases, parliament may opt to reassess its relevant digital plans under the light of the proposed framework.

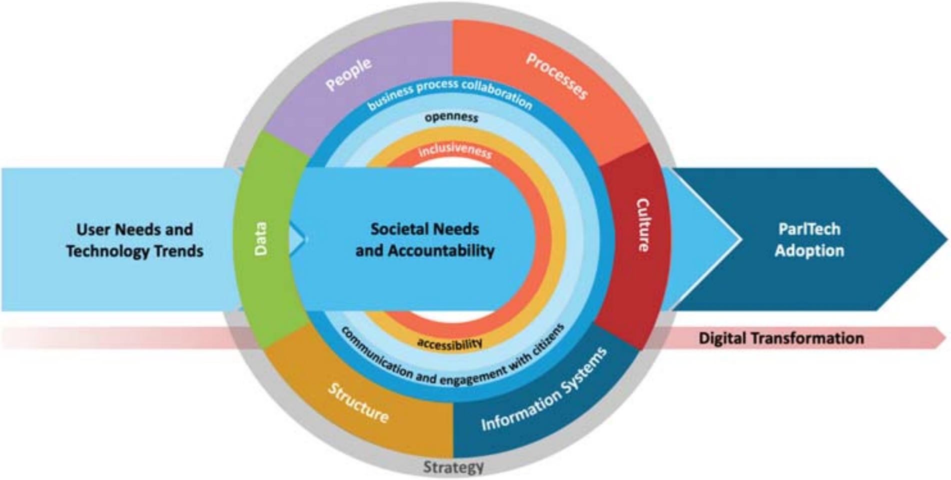

People
Processes
business process collaboration
openness
inclusiveness
User Needs and Technology Trends
Data
Societal Needs and Accountability
Culture
ParlTech Adoption
Digital Transformation
accessibility
communication and engagement with citizens
Structure
Information Systems
Strategy

Figure 2. Digital parliament transformation framework.

The above framework is more than a mere thought experiment. It relies on established knowledge and trustworthy data from a structured expert survey. Therefore, it can serve as a point of reference or an inspiration for parliament actors when planning digital strategies and action plans. However, there are several technology parameters that are yet to be defined with precision. At the same time, the authors are aware that the proposed framework may appear inexplicit, e.g., when defining the underlying principles or in the justification of a basic set of technologies. It needs to be noted that this was the intended purpose, since an overall too-stiff concept in the era of disruptive technology would risk being overturned all too soon.

6. Conclusions and Outlook

Parliaments are complex representative institutions that can benefit from on-going digitalization, particularly through the use of emerging technologies. The authors evaluated the results from a structured expert survey directed to internal parliamentary actors, parliamentary professionals and MPs, who constitute users of the tools and services of the Digital Parliament. Data, people and, unsurprisingly, information systems are still top priorities for parliament digitalization, thus confirming IPU's 2018 report [23]. On the other hand, societal barriers, such as culture and change, and lack of tangible strategies and plans, may hinder digitalization, even if there is no lack of resources. This is why stakeholders in parliaments play a significant role in organizational transformation, something which is also positively correlated with parliament digitalization (ANOVA $p = 0.006$, $F(3, 28) = 5.064$). Open, transparent legal data, which are inherently linked to increased accountability and accessibility, are also of significance. This, again, leads to the determination of parameters such as openness, accessibility, and communication with citizens as particularly relevant contexts for the digital parliament.

In terms of applicability, maturity, and usefulness, the evaluation of expert preferences pointed toward a number of technologies particularly interesting for parliamentary use, such as legal informatics, integrated tools and services, virtual parliament, social media analytics, and rapid digital and operational transformation. However, significant development efforts are necessary for them to be adapted, modified, and customized for use within parliaments. Parliamentary experts stated that these technologies will bestow added value to parliamentary professional work (internal environment). In addition, there are indications that such tools and services will strengthen the democratic appreciation of citizens (external environment) by empowering and improving relationships between parliament and its civic stakeholders.

By combining quantitative findings with the qualitative approach of Gartner's depiction, a parliamentary hype cycle has been created. Indeed, Gartner proved to provide solid guidance to assess emerging ParlTech. According to the developed parliamentary hype cycle, technologies can be screened for suitability in the institutional workspace. Overall, an analogy to the original hype cycle can be observed, yet responses are concentrated in the prism of parliament use.

Nonetheless, the introduction of emerging technologies should be performed within a wider digital framework. The findings from this study enable the construction of a rigid framework for the digital parliament out of four components, i.e., strategy, operations, technology, and transformation, with specific boundary conditions for the utilization of novel parliamentary technologies. Within this framework, the user plays a central role in its design and implementation, having digitalization as an ultimate scope. For any given parliament, democratic tradition is deeply embedded in its organizational culture. Though indirectly accounted for when discussing ParlTech attributes (e.g., people and culture), the study of related deeper political, societal, and organizational perceptions, interrelations, and ethical structures is well outside the scope of this article, and further research is needed to cover this field.

The evaluation results from the survey produced comprehensive insight, whose detailed presentation goes well beyond the scope of a single publication. The authors will continue the study of the data to come up with novel insights that further elucidate the framework and individual components of the digital parliament. They also point at the online dataset made available to the research community and call for further interdisciplinary studies on the ParlTech field. As new digital technologies emerge at a high rate, increasing investments and cross-sectors collaboration within the defined technological and organizational framework are necessary for them to be efficiently deployed in the parliamentary environment. In addition, a more detailed view of individual technologies appears to be advantageous, possibly prioritizing the ones that are built on artificial intelligence background; for instance, recommender systems (for their use in parliaments see [52]).

Ultimately, under the light of the digital (r)evolution, one has to verify once again the very notion of the digital parliament. This study suggests that the parliament of the future will be more a mere aggregation of tools and technologies. This new parliament will still have strong social and procedural components (see also [11]). It is in the people's interest that the intra-parliamentary actors do not develop negative-biased perceptions for emerging technologies that have the potential to shape the future or legislatures. Tangible digital strategies and targeted re-education of personnel and parliamentarians to develop essential digital skills, a notion that is labelled as 'training' in the 2018 IPU e-Parliament report [23], seem to be inevitable steps towards the digital future of representative institutions.

Author Contributions: All authors conceived, designed and performed the experiments; analyzed the data and wrote the paper. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data presented in this study are openly available in Figshare at doi: 10.6084/m9.figshare.13604030.v3.

Acknowledgments: The authors would like to thank the MPs and parliamentary professionals who participated to the expert survey.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

Table A1. Basic sample demographics.

Continent	Europe	Asia	Africa	S. America	Oceania				
Country total	10	8	4	2	1				
Per cent	40.0	32.0	16.0	8.0	4.0				
Working sector	Committees	Library-Research	Int'l Relations	Legislative-Oversight	IT	Leadership	Training	MP	Transparency
Sector total	7	7	6	6	4	4	3	3	1
Per cent [1]	21.9	21.9	18.8	18.8	12.5	12.5	9.4	9.4	3.1
Academic background	Legal	Political science	Public Admin.	Information Science	Other [2]				
Total	11	7	5	5	4				
Per cent	34.4	21.9	15.6	15.6	12.4				

[1] More than one selection was possible; hence, the total percentage exceeds 100%. [2] Other: Economics, History, Higher Education Policy, and Urban Geography, each represented once.

Appendix B

Table A2. Expert study aggregated results.

Digitalization											
Priority	Level of Digitization	People	Process	Architecture	Data						
Mean L (1–5)	3.10	4.31	4.34	3.81	4.47						
σ	1.31	0.74	0.60	0.69	0.57						
Relevance	Society representation	Lawmaking & oversight	Openness	Inclusiveness	Accessibility	Accountability	Effectiveness	Communication & Engagement	Business process collaboration		
Mean L (1–5)	3.59	3.97	4.31	3.94	4.28	4.16	3.91	4.31	3.59		
σ	1.13	0.86	0.78	0.91	0.68	0.72	0.78	0.90	0.87		
Digital Transformation											
Organizational Perspective	Process	People	Culture	Structure	Systems	Data					
Mean L (1–5)	3.69	3.66	3.25	3.31	3.88	3.94					
σ	0.86	0.83	0.80	0.93	0.98	0.88					
Organizational barriers	Lack of roadmap & plan	Lack of skills & knowhow	Personnel shortage	Lack of political support	Lack of managerial support	Organizational complexity	Lack of coordination	Resistance to change	Bureaucratic culture	Fear of innovation	Lack of budget resources
Mean L (1–5)	3.50	3.34	3.16	3.28	2.91	3.16	3.13	3.53	3.63	3.22	2.91
σ	1.02	1.23	1.25	1.08	0.86	1.11	0.91	0.95	1.01	1.01	0.93
Organizational drivers	Strong top-down leadership	Identify user needs bottom up	Internal status quo issues	Expected benefits (internal)	Expected benefits (external)	External society pressure	External legal obligations	Disruptive technology effects			
Mean L (1–5)	3.47	3.31	3.09	3.72	3.28	3.16	2.94	3.16			
σ	0.88	0.82	0.89	0.81	0.89	0.92	0.95	0.95			
Importance of IPU trends											
Trend	Embedded Digital technologies	MPs committed to digital technologies	MPs role diminished as ICT operational	Rise in XML adoption has leveled off	Use of social media messaging	Video stream overtakes traditional broadcasting	Barriers to ICT: training & skill deficits	Security and trust concerns among MPs & parliamentary administrators	Knowledge of work of parliaments: barrier to citizen's engagement	Parliament collaborates with PMOs [1]	Inter-parl. support in areas of ICT
Mean L(1–5)	3.34	3.63	2.91	3.06	3.50	3.94	3.81	3.78	3.84	3.28	3.69
σ	1.00	1.07	1.00	0.98	0.92	0.84	1.00	1.07	0.99	0.81	0.97

[1] Parliamentary Monitoring Organizations.

References

1. Pedersen, K. E-government transformations: Challenges and strategies. *Transform. Gov. People Process. Policy* **2018**, *12*, 84–109. [CrossRef]
2. Nograšek, J.; Vintar, M. Observing organisational transformation of the public sector in the e-government era. *Transform. Gov. People Process. Policy* **2015**, *9*, 52–84. [CrossRef]
3. Baptista, J.; Stein, M.-K.; Klein, S.; Watson-Manheim, M.B.; Lee, J. Digital work and organisational transformation: Emergent Digital/Human work configurations in modern organisations. *J. Strat. Inf. Syst.* **2020**, *29*, 101618. [CrossRef]
4. Benbya, H.; Nan, N.; Tanriverdi, H.; Yoo, Y. Complexity and Information Systems Research in the Emerging Digital World. *MIS Q.* **2020**, *44*, 1–17.
5. Maedche, A.; Legner, C.; Benlian, A.; Berger, B.; Gimpel, H.; Hess, T.; Hinz, O.; Morana, S.; Söllner, M. AI-Based Digital Assistants. *Bus. Inf. Syst. Eng.* **2019**, *61*, 535–544. [CrossRef]
6. Lyytinen, K.; Nickerson, J.V.; King, J.L. Metahuman systems = humans + machines that learn. *J. Inf. Technol.* **2020**. [CrossRef]
7. Dai, X.; Norton, P. The Internet and Parliamentary Democracy in Europe. *J. Legis. Stud.* **2007**, *13*, 342–353. [CrossRef]
8. Leston-Bandeira, C. The Impact of the Internet on Parliaments: A Legislative Studies Framework. *Parliam. Aff.* **2007**, *60*, 655–674. [CrossRef]
9. Nograšek, J.; Vintar, M. E-government and organisational transformation of government: Black box revisited? *Gov. Inf. Q.* **2014**, *31*, 108–118. [CrossRef]
10. Magnusson, J.; Khisro, J.; Melin, U. A Pathology of Public Sector IT Governance: How IT Governance Configuration Counteracts Ambidexterity. In *Electronic Government*; EGOV 2020; Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2020; Volume 12219, pp. 29–41. [CrossRef]
11. Mergel, I.; Edelmann, N.; Haug, N. Defining digital transformation: Results from expert interviews. *Gov. Inf. Q.* **2019**, *36*, 101385. [CrossRef]
12. Piccinini, E.; Hanelt, A.; Gregory, R.; Kolbe, L. Transforming Industrial Business: The Impact of Digital Transformation on Automotive Organizations. In Proceedings of the International Conference on Information Systems, Fort Worth, TX, USA, 13–16 December 2015.
13. Tangi, L.; Janssen, M.; Benedetti, M.; Noci, G. Barriers and Drivers of Digital Transformation in Public Organizations: Results from a Survey in the Netherlands. In *Electronic Government*; EGOV 2020; Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2020; Volume 12219, pp. 42–56. [CrossRef]
14. Skog, D.A.; Wimelius, H.; Sandberg, J. Digital Disruption. *Bus. Inf. Syst. Eng.* **2018**, *60*, 431–437. [CrossRef]
15. Romanelli, M. New Technologies for Parliaments Managing Knowledge for Sustaining Democracy. *Manag. Dyn. Knowl. Econ. J.* **2016**, *4*, 649–666.
16. Papaloi, A.; Gouscos, D. E-Parliaments and Novel Parliament-to-Citizen services. *JeDEM—eJournal eDemocracy Open Gov.* **2011**, *3*, 80–98. [CrossRef]
17. Fitsilis, F.; Koryzis, D.; Svolopoulos, V.; Spiliotopoulos, D. Implementing Digital Parliament Innovative Concepts for Citizens and Policy Makers. In *HCI in Business, Government and Organizations*; Interacting with Information Systems; HCIBGO 2017; Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2017; Volume 10293, pp. 154–170. [CrossRef]
18. De Barros, A.T.; Bernardes, C.B.; Rehbein, M. Brazilian Parliament and digital engagement. *J. Legis. Stud.* **2016**, *22*, 540–558. [CrossRef]
19. Davis, F.D. Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology. *MIS Q.* **1989**, *13*, 319–340. [CrossRef]
20. Kouroubali, A.; Katehakis, D.G. The new European interoperability framework as a facilitator of digital transformation for citizen empowerment. *J. Biomed. Inform.* **2019**, *94*, 103166. [CrossRef] [PubMed]
21. Koryzis, D.; Fitsilis, F.; Spiliotopoulos, D.; Theocharopoulos, T.; Margaris, D.; Vassilakis, C. Policy Making Analysis and Practitioner User Experience. In *HCI International 2020—Late Breaking Papers: User Experience Design and Case Studies*; HCII 2020; Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2020; Volume 12423, pp. 415–431. [CrossRef]
22. Fallon, F.; Allen, B.; Williamson, A. *Parliament 2020: Visioning International Comparison–Australia, Canada, Chile and the United Kingdom*; Hansard Society: London, UK, 2011.
23. Inter-Parliamentary Union. *World E-Parliament Report 2018*; Inter-Parliamentary Union: Geneva, Switzerland, 2018; ISBN 9789291427352.
24. Burke, B.; Litan, A.; Natis, Y.V. Gartner Hype Cycle for Emerging Technologies. 2020. Available online: https://www.gartner.com/document/3987951 (accessed on 19 December 2020).
25. Maestas, C. *Expert Surveys as a Measurement Tool*; Atkeson, L.R., Alvarez, R.M., Eds.; Oxford University Press: Oxford, UK, 2016; Volume 1.
26. Charalambous, G.; Lamprianou, I. Societal Responses to the Post-2008 Economic Crisis among South European and Irish Radical Left Parties: Continuity or Change and Why? *Gov. Oppos.* **2014**, *51*, 261–293. [CrossRef]

27. Palinkas, L.A.; Horwitz, S.M.; Green, C.A.; Wisdom, J.P.; Duan, N.; Hoagwood, K. Purposeful Sampling for Qualitative Data Collection and Analysis in Mixed Method Implementation Research. *Adm. Policy Ment. Heal. Ment. Heal. Serv. Res.* **2015**, *42*, 533–544. [CrossRef]
28. Hellweger, S.; Wang, X. What Is User Experience Really: Towards a UX Conceptual Framework. *arXiv* **2015**, arXiv:1503.01850.
29. Shackel, B. Usability—Context, framework, definition, design and evaluation. *Interact. Comput.* **2009**, *21*, 339–346. [CrossRef]
30. Fitsilis, F.; Koryzis, D.; Dalas, A.; Spiliotopoulos, D. Questionnaire on the Digital Parliament. 2021. Available online: https://figshare.com/articles/dataset/Questionnaire_on_the_digital_parliament/13604030/3 (accessed on 13 March 2021).
31. Murphy, J. *Parliaments and Crisis: Challenges and Innovations*; International Institute for Democracy and Electoral Assistance: Stockholm, Sweden, 2020; ISBN 9789176713082.
32. Ritz, A.; Brewer, G.A.; Neumann, O. Public Service Motivation: A Systematic Literature Review and Outlook. *Public Adm. Rev.* **2016**, *76*, 414–426. [CrossRef]
33. van Wart, M. *Leadership in Public Organizations*; Routledg: Abingdon-on-Thames, UK, 2014; ISBN 9781315702926.
34. Leventis, S.; Anastasiou, V.; Fitsilis, F. Application of enterprise integration patterns for the digital transformation of parliamentary control. In Proceedings of the 13th International Conference on Theory and Practice of Electronic Governance, New York, NY, USA, 23 September 2020.
35. Garofalakis, J.; Plessas, K.; Plessas, A.; Spiliopoulou, P. Modelling Legal Documents for Their Exploitation as Open Data. In *Business Information Systems*; BIS 2019; Lecture Notes in Business Information Processing; Springer: Cham, Switzerland, 2019; Volume 353, pp. 30–44. [CrossRef]
36. Papastylianou, A.; Stasis, A.; Rantos, K.; Kalogirou, V. Blended Learning and Open Courseware for Promoting Interoperability in Public Services. In *E-Democracy—Safeguarding Democracy and Human Rights in the Digital Age*; e-Democracy 2019; Communications in Computer and Information Science; Springer: Cham, Switzerland, 2020; Volume 1111, pp. 79–93. [CrossRef]
37. Leston-Bandeira, C.; Bender, D. How deeply are parliaments engaging on social media? *Inf. Polity* **2013**, *18*, 281–297. [CrossRef]
38. Stieglitz, S.; Brockmann, T.; Dang-Xuan, L. Usage of Social Media for Political Communication. In Proceedings of the 16th Pacific Asia Conference on Information Systems, Ho Chi Minh City, Vietnam, 11–15 July 2012; p. 22.
39. Demidova, E.; Barbieri, N.; Dietze, S.; Funk, A.; Holzmann, H.; Maynard, D.; Papailiou, N.; Peters, W.; Risse, T.; Spiliotopoulos, D. Analysing and Enriching Focused Semantic Web Archives for Parliament Applications. *Futur. Int.* **2014**, *6*, 433–456. [CrossRef]
40. Charalabidis, Y.; Loukis, E.; Alexopoulos, C.; Lachana, Z. The Three Generations of Electronic Government: From Service Provision to Open Data and to Policy Analytics. In *Electronic Government*; EGOV 2019; Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2019; Volume 11685, pp. 3–17. [CrossRef]
41. Flavián, C.; Ibáñez-Sánchez, S.; Orús, C. The impact of virtual, augmented and mixed reality technologies on the customer experience. *J. Bus. Res.* **2019**, *100*, 547–560. [CrossRef]
42. Alexopoulos, C.; Lachana, Z.; Androutsopoulou, A.; Diamantopoulou, V.; Charalabidis, Y.; Loutsaris, M.A. How Machine Learning is Changing e-Government. In Proceedings of the 12th International Conference on Theory and Practice of Electronic Governance, New York, NY, USA, 3 April 2019.
43. Margetts, H.; Dorobantu, C. Rethink government with AI. *Nature* **2019**, *568*, 163–165. [CrossRef]
44. El Saddik, A. Digital Twins: The Convergence of Multimedia Technologies. *IEEE MultiMedia* **2018**, *25*, 87–92. [CrossRef]
45. Strawn, G. Open Science and the Hype Cycle. *Data Intell.* **2021**, *3*, 1–7. [CrossRef]
46. Dedehayir, O.; Steinert, M. The hype cycle model: A review and future directions. *Technol. Forecast. Soc. Chang.* **2016**, *108*, 28–41. [CrossRef]
47. Agarwal, P.; Sastry, N.; Wood, E. Tweeting MPs: Digital Engagement between Citizens and Members of Parliament in the UK. In Proceedings of the 13th International AAAI Conference on Web and Social Media, Munich, Germany, 11–14 June 2019; Volume 13, pp. 26–37.
48. Costa, O.; Lefébure, P.; Rozenberg, O.; Schnatterer, T.; Kerrouche, E. Far Away, So Close: Parliament and Citizens in France. *J. Legis. Stud.* **2012**, *18*, 294–313. [CrossRef]
49. Matt, C.; Hess, T.; Benlian, A. Digital Transformation Strategies. *Bus. Inf. Syst. Eng.* **2015**, *57*, 339–343. [CrossRef]
50. Gimpel, H.; Hosseini, S.; Huber, R.X.R.; Probst, L.; Röglinger, M.; Faisst, U. Structuring Digital Transformation: A Framework of Action Fields and Its Application at ZEISS. *J. Inf. Technol. Theory Appl.* **2018**, *19*, 3.
51. Nwaiwu, F. Tomas Bata University in Zlín Review and Comparison of Conceptual Frameworks on Digital Business Transformation. *J. Compet.* **2018**, *10*, 86–100. [CrossRef]
52. De Campos, L.M.; Fernández-Luna, J.M.; Huete, J.F.; Redondo-Expósito, L. Positive unlabeled learning for building recommender systems in a parliamentary setting. *Inf. Sci.* **2018**, *433-434*, 221–232. [CrossRef]

big data and
cognitive computing

MDPI

Article

GeoLOD: A Spatial Linked Data Catalog and Recommender

Vasilis Kopsachilis *,† and Michail Vaitis †

Department of Geography, University of the Aegean, GR-811 00 Mytilene, Greece; vaitis@aegean.gr
* Correspondence: vkopsachilis@geo.aegean.gr
† Current address: Department of Geography, University Hill, GR-811 00 Mytilene, Greece.

Abstract: The increasing availability of linked data poses new challenges for the identification and retrieval of the most appropriate data sources that meet user needs. Recent dataset catalogs and recommenders provide advanced methods that facilitate linked data search, but none exploits the spatial characteristics of datasets. In this paper, we present GeoLOD, a web catalog of spatial datasets and classes and a recommender for spatial datasets and classes possibly relevant for link discovery processes. GeoLOD Catalog parses, maintains and generates metadata about datasets and classes provided by SPARQL endpoints that contain georeferenced point instances. It offers text and map-based search functionality and dataset descriptions in GeoVoID, a spatial dataset metadata template that extends VoID. GeoLOD Recommender pre-computes and maintains, for all identified spatial classes in the Web of Data (WoD), ranked lists of classes relevant for link discovery. In addition, the on-the-fly Recommender allows users to define an uncatalogued SPARQL endpoint, a GeoJSON or a Shapefile and get class recommendations in real time. Furthermore, generated recommendations can be automatically exported in SILK and LIMES configuration files in order to be used for a link discovery task. In the results, we provide statistics about the status and potential connectivity of spatial datasets in the WoD, we assess the applicability of the recommender, and we present the outcome of a system usability study. GeoLOD is the first catalog that targets both linked data experts and geographic information systems professionals, exploits geographical characteristics of datasets and provides an exhaustive list of WoD spatial datasets and classes along with class recommendations for link discovery.

Keywords: linked data; spatial datasets; data catalog; dataset recommender

Citation: Kopsachilis, V.; Vaitis, M. GeoLOD: Spatial Linked Data Catalog and Recommender. *Big Data Cogn. Comput.* **2021**, *5*, 17. https://doi.org/10.3390/bdcc5020017

Academic Editor: Min Chen

Received: 31 January 2021
Accepted: 15 April 2021
Published: 19 April 2021

1. Introduction

Linked data principles [1] lay the technological background for data publishing on the web so that they can be transparently and uniformly accessed by humans and software. Link establishment among related data increases data sharing, interoperability, and reuse; aids dataset enrichment; and unleashes powerful retrieval capabilities already exploited by question answering [2–5] and query federation [6–11] systems. The idea of a web of open and interlinked data has been embraced by scientists and organizations, and steps have been taken towards this direction during the last decade or so. At the early stages of linked data development, providers such as DBpedia [12], MusicBrainz [13] and GeoNames [14] converted their data to RDF and made them accessible through dumps, SPARQL endpoints or embedded them in HTML documents using RDFa [15]. Since then, many tools have been developed, such as search engines [16,17], data catalogs [18,19], link discovery frameworks [20,21], and dataset recommenders [7,22–24], forming the linked data tools ecosystem and facilitating users to consume linked data and lowering the barriers for its adoption by non-expert users. Today, the Linked Open Data (LOD) cloud diagram includes more than 1200 datasets, and DataHub maintains metadata for more than 700 datasets. References [25,26] note that linked data size is expanding and the number of the LOD cloud diagram datasets increased from 203 to 1269 during the period 2010–2020. LODLaundromat [27] reports 38 billion indexed triples in 2018.

The increasing availability of linked data provides more options to users, but at the same time, increases the difficulty in identifying the appropriate data sources that meet their needs. Concerning linked data search, user needs vary and some common scenarios include searching for (a) topic-specific datasets (e.g., about conferences, music, or geography) [28]; (b) datasets that contain a given entity [29,30]; and (c) similar datasets to a given dataset [23,31]. These scenarios are being covered by available tools and applications; however, to the best of our knowledge, there is not a tool that addresses user needs related to geographical aspects of datasets during linked data search and exploration. In this work, we identify and address four possible scenarios:

1. A user searches for datasets that cover a specific geographical area (e.g., a country);
2. A linked data publisher searches for datasets containing georeferenced information in order to georeference their data;
3. A linked data publisher searches for datasets that contain related instances to their own datasets in order to establish links between instances; and
4. A geographical information systems (GIS) professional wants to enrich their spatial data with linked data.

These scenarios are covered in GeoLOD, a web catalog of spatial Web of Data (WoD) datasets and classes and a recommender for spatial datasets and classes that may contain related instances. The terms spatial datasets and spatial classes denote datasets and classes, respectively, that contain georeferenced instances, that is, instances whose locations are expressed with longitude and latitude coordinates. GeoLOD parses LOD cloud and DataHub catalogs, identifies spatial datasets and their spatial classes and extracts their metadata. It generates additional metadata that capture spatial aspects of datasets and classes, such as their bounding box and number of spatial entities and associated spatial vocabularies, and exposes them in GeoVoID, a vocabulary that extends the Vocabulary of Interlinked Datasets (VoID) [32], to describe spatial aspects of datasets. GeoLOD Catalog allows access to the lists of linked data spatial datasets and classes (along with their metadata) through a user interface and provides text and map-based search functionality, thus addressing scenarios 1 and 2.

GeoLOD Recommender generates ranked lists of spatial datasets and classes that may contain related instances with a given dataset or class, so as to be further examined in link discovery processes for the establishment of `owl:sameAs` links or other links that denote close semantic relation among their instances. The recommendation method is based on the work presented in [33] that builds a recommendation algorithm on the hypothesis that "pairs of classes whose instances present similar spatial distribution are more related than pairs of classes whose instances present dissimilar spatial distribution, in the sense that the former are more likely to contain semantically related instances" (p. 152), and thus are better candidates to be used as input in a link discovery process. GeoLOD applies the recommendation algorithm to generate recommendations for each class in the Catalog in the background. It allows the exploration for related classes and datasets through the user interface and the export of automatically generated SILK and LIMES configuration files for a selected pair of recommended classes that can be directly used for link discovery processes. Additionally, it allows on-the-fly recommendations for classes provided through a user-defined SPARQL endpoint, not listed in the Catalog, and for GeoJSON and Shapefile datasets, which are typical geographic information systems (GIS) file formats, thus addressing scenarios 3 and 4.

In addition to the user interface, GeoLOD provides a REST API to serve its content in well-known templates and formats, enabling software-based consumption. Specifically, it provides services that expose GeoLOD metadata and content description in the Data Catalog Vocabulary (DCAT) [34] format, an RDF vocabulary designed to facilitate interoperability among data catalogs, and dataset descriptions in GeoVoID that can aid source selection in query federation systems. It also provides services that export (a) SILK and LIMES configuration files for a selected pair of classes and (b) class recommendation lists for datasets and classes in order to be consumed as input in batch link discovery processes. The main novelties of GeoLOD are:

- It is the first catalog of linked data spatial datasets and classes provided through SPARQL endpoints, offering services for describing spatial aspects of their content and map-based search;
- It introduces GeoVoID, an automatically generated dataset description vocabulary that extends VoID, to express spatial metadata and statistics of datasets;
- It provides a comprehensive list of recommended pairs of datasets and classes that may contain related instances, along with automatically generated SILK and LIMES configuration files and machine-readable recommendation lists so as to be used as input in (batch) link discovery processes; and
- It allows on-the-fly recommendations for user-defined SPARQL endpoints and spatial datasets in GeoJSON and Shapefile format.

The rest of the paper is organized as follows. In Section 2, we present applications related to linked data search and dataset recommendation. In Section 3, we present the design and methods of the GeoLOD application, and in Section 4, we present its implementation and the usage of the user interface and the REST API. In Section 5, we present statistics that summarize the linked data status regarding the geospatial domain, we assess the applicability of GeoLOD recommender in relation with the LIMES framework, and we evaluate GeoLOD usability by different user categories, namely linked data and GIS experts. We conclude the paper in Section 6 by discussing the results and by providing pointers for the improvement of the application.

2. Related Work

In this section, we present the work related to GeoLOD, classified into three categories: (a) vocabularies and tools for dataset description, (b) dataset catalogs, and (c) dataset recommenders for link discovery. We focus on prototypes and available systems for the linked data domain.

2.1. Dataset Description

VoID [32] (Vocabulary of Interlinked Datasets) is a well-known vocabulary for describing dataset content by expressing general information (such as, title, keywords, distribution URL, and provenance metadata), statistics (such as number of triples, classes, and properties), and connectivity details to other datasets. It aims to facilitate users and software agents in their dataset exploration [35], and many tools have been developed to generate automated VoID-based or similar dataset descriptions and statistics [36,37]. For example, RDFStats [38] provides an API that generates statistical items for SPARQL endpoints and RDF documents, including instance counts (per class) and histograms (per class, property and value type), originally developed to aid query federation systems. ExpLOD [39] summarizes RDF datasets usage and interlinking by computing representative dataset graphs and statistics, such as number of class instances and predicates used to describe an instance. LODStats [40] defines 32 statistical criteria, extending those defined in VoID, in a scalable and high-performance framework. Aether [41] is a statistics generator and visualization web application that focuses on comparing datasets between versions and on error detection. Loupe [42] and ABStat [43] produce ontology-driven dataset summaries that highlight their structure. ProLOD++ [44] augments dataset analytics with data mining functionality for identifying dependencies between dataset entities such as synonymously used predicates. In addition to dataset statistics, several tools, including LODex [45], LOD-Vader [46], LODAtlas [47], and LODSynthesis [31], provide high-level dataset summaries and visualizations. Concerning the description of geographical elements of the datasets, VoID supports the expression of their geographical coverage (e.g., bounding box) using the Dublin Core [48] spatial coverage predicate, and LODStats allows the (indirect) computation of geographical coverage by combining the minimum and maximum statistical criteria of longitude and latitude property values. Nevertheless, none of the above-described vocabularies and tools capture the geographical aspects of datasets covered in this work, such as the number of georeferenced instances in datasets and classes.

2.2. Dataset Catalogs

Dataset catalogs provide single entry points for available linked-data datasets, and the most prominent examples are arguably the Linked Open Data (LOD) cloud and the DataHub. The LOD cloud visualizes datasets by topic, portrays their connectivity, and exports the list of its datasets in JSON format along with their basic provenance and descriptive dataset-level metadata, such as title, description, domain, point of contact, and distribution info (e.g., access URL and SPARQL enpoint URL). DataHub provides a user interface and a CKAN API (an API for querying data catalogs) for searching and filtering (not exclusively RDF) datasets and viewing their metadata. Both catalogs are populated through user-submitted datasets and metadata. LODAtlas [47] is a data catalog that provides keyword search and faceted navigation for RDF datasets parsed from several other catalogs including DataHub, Europeana, and Data.gov. It maintains dataset metadata, statistics about the number of their triples and their in- and out-going links. Moreover, it allows concurrent and in-depth exploration and comparison of multiple datasets' characteristics and provides an overview of their connectivity based on visual summaries. LODLaundromat [49] aims to improve linked data quality by republishing data in a "cleaner" state after correcting syntax errors, filtering duplicates, replacing blank nodes, etc. As part of the cleaning process, it offers description and search services for 650,000 cleaned RDF datasets (mostly data dumps). SPARQLES [50] monitors more than 500 SPARQL endpoints, collected from DataHub, regarding their availability, performance, interoperability, and discoverability, and provides a user interface for humans and an API for software agents for consuming its content. SPORTAL [51] is a catalog of SPARQL endpoints that allows SPARQL and keyword-based search. Endpoints are profiled by extended VoID descriptions, computed by directly querying their content. IDOL [52] provides metadata and analytics about an exhaustive list of RDF datasets in various formats (e.g., zip files and SPARQL endpoints), located by parsing eight data catalogs (including LOD cloud, LODLaundromaut, and the Registry of Research Data Repositories [53]). However, the list of datasets and their analytics are available only through a dump file. Contrary to the above generic data catalogs, LSLOD [54] and YummyData [55] are domain-specific data catalogs. The LSLOD Catalogue contains 52 life-sciences-related SPARQL endpoints for serving ontology alignment purposes between different datasets in the life science domain. Even though some catalogs allow (indirectly) the search for spatial datasets (e.g., in LODAtlas, users can retrieve spatial datasets by selecting a spatial vocabulary in the faceted search component), GeoLOD, to the best of our knowledge, is the first geographical domain data catalog that provides summaries for spatial aspects of datasets. Moreover, GeoLOD Catalog implements some novel features like the map-based dataset and class search and the on-the-fly projection of class spatial instances on an interactive map.

2.3. Dataset Recommenders for Link Discovery

Link Discovery refers to the problem of identifying and interlinking pairs of instances between two given triplesets for which a relation holds [56]. Two well-known link-discovery frameworks are SILK [20] and LIMES [21], which execute a link discovery process by allowing the set up of a workflow in configuration files or in user interfaces. The general workflow of a link discovery process consists of (a) providing as input two triplesets (e.g., two datasets or two classes), usually referred to as source and target, respectively; (b) defining the type of relation between their instances that will be discovered and established (e.g., `owl:sameAs`, which means the two instances refer to same real-world object); (c) defining the matching rule, that consists of one or more similarity metrics and the instance properties that will be evaluated (e.g., string equality of instance labels); and finally (d) executing the workflow to generate the recommended links between the instances of the two triplesets. A common obstacle for initiating a link discovery process is that sometimes there is no prior knowledge of which two triplesets can be used as input for the link discovery process, or a linked data publisher may not be aware of target triplesets that are likely to contain related instances with their (source) tripleset. This is the focus of the Dataset Recommendation for Link Discovery domain, which refers to the automated process of recommending triplesets (e.g., datasets or classes) that may contain related instances to a given tripleset in order to be used as input in a link-discovery process.

Although several methodologies have been proposed to address the problem of Dataset Recommendation for Link Discovery [22,28,57–62], only few are implemented in tools and web applications [23,31,63,64]. One of them, the FluidOps portal [64], offers a data source exploration service, involving users in the source selection process, where a user begins to explore by providing some input (e.g., a keyword) and then refining the results through faceted search. It employs a data source contextualization method for discovering sources containing "somehow" related entities, and thus can serve link-discovery and distributed query processing tasks. TRT [63] recommends relevant triplesets for link discovery by applying link prediction metrics on a graph that maintains dataset connectivity information extracted from DataHub metadata. TRTML [23] augments the recommendation process with supervised learning algorithms. The input to the TRT/TRTML tool is the VoID description of the tripleset that the user wants to get recommendations, and the output is a ranked list of relevant triplesets for link discovery. LODSynthesis [31] is a suite of services for linked data search that includes object co-reference, fact checking, dataset discovery based on connectivity analysis, and connectivity analytics and visualizations. It indexes the entire content of hundreds of datasets in the LOD cloud and recommends relevant datasets by taking into consideration the closure of equivalence relationships based on existing instance (`owl:sameAs`) and class (`owl:EquivalentClass`) equivalence links. As an example, users can request for the K datasets that are most connected with the Hellenic Fire Brigade dataset. A related but slightly different tool is Linklion [30], a semantic web link repository, that is, a catalog of identified links between data sources populated from user-employed link discovery processes, which contains 12.6 million links of 10 different relation types (e.g., `owl:sameAs`, `dbo:spokenIn`) for 449 datasets. The main difference between our work and all the above is that their recommendation processes are based on information about existing links between datasets, while ours is based on the similarity of the spatial distribution of datasets and classes instances. GeoLOD novelties also include on-the-fly class recommendation for spatial datasets in GIS formats and the export of the recommended pair of classes to SILK and LIMES configuration files for direct use in a link-discovery process.

3. Design and Methods

GeoLOD consists of two distinct but complementary modules: (a) the Catalog of spatial datasets and classes, and (b) the Recommender of candidate datasets and classes for link discovery. In the following sections, we present the design of and the methods used in each module.

3.1. The Catalog

The goal of the Catalog is to provide lists of linked data spatial datasets and classes and methods for their textual and spatially-based retrieval. Each catalog item (a spatial dataset or a class) should be described by its metadata, with an emphasis on describing their spatial characteristics. Users and agents should be able to browse and search the catalog and select an item to view its full description. The main design decisions include (a) the definitions of the terms spatial dataset and spatial class, (b) the identification of the methods for collecting information about available spatial datasets and classes, and (c) the metadata set for describing catalog items.

3.1.1. Definitions

An RDF triple is a statement about two resources that follows the `subject predicate object` structure, where `subject` and `object` represent two resources and `predicate` their relation. A set of triples (`S`) is denoted as $S = I \times R \times (I \cup L)$, where `I`, `L`, and `R` represent instances, literals, and relations, respectively, so that `subject` corresponds to an instance, `predicate` to a relation, and `object` to an instance or a literal. With the term spatial dataset, we refer to "a set of RDF triples published, maintained or aggregated provided by a single provider" [32] containing spatial instances, that is, a subject explicitly georeferenced with predicates defined in a spatial vocabulary. A spatial vocabulary defines predicates that allow the representation of an instance location in the form of longitude/latitude coordinates in a well-known Coordinate Reference System (CRS), such as WGS84 (e.g., `Athens hasLongitude ‘‘23.58’’`). A spatial class is a subset of a spatial dataset containing spatial instances declared to be instances of a dataset class using the `rdf:type` predicate (e.g., `Athens rdf:type City`). In this work, we search and catalog spatial datasets and their spatial classes, whose instances' locations are expressed as single points, that is, by a longitude and a latitude value, using the W3C Basic Geo [65], GeoVocab [66], GeoSPARQL [67], GeoNames [68], or GeoRSS [69], which are common spatial vocabularies listed in Linked Open Vocabulary (LOV) [70] and LOV4IoT [71]. Furthermore, we search and catalog only those datasets provided by SPARQL endpoints and not by other means, such as RDF dump files. A SPARQL endpoint is an interface that is accessible through a URL and allows access to the triples of a dataset using SPARQL, which is the standard language for querying linked data. Therefore, the terms datasets and SPARQL endpoints are used in the remainder of the paper interchangeably.

3.1.2. Data Collection

The initial pool of information about available linked data datasets is formed by parsing the content of other well-known dataset catalogs, namely LOD cloud and DataHub, which provide means for automated consumption of their contents. Specifically, LOD cloud exposes a list of datasets and their metadata at https://lod-cloud.net/lod-data.json (accessed on 16 April 2021) in JSON (an open standard and lightweight data-interchange format), and DataHub allows access to its dataset list using the CKAN API [72] (an API for querying data catalogs). GeoLOD Catalog parses the LOD cloud and DataHub to locate datasets provided through SPARQL endpoints and extract basic metadata, such as their title and endpoint URL. Then, it sends ASK queries to the located SPARQL endpoints to identify which of them uses any of the spatial vocabularies defined in Section 3.1.1. An ASK query is a SPARQL variation that is used to return a true or false answer to the issued query. For example, the ASK query below returns true if the endpoint contains triples that use the `http://www.w3.org/2003/01/geo/wgs84_pos#long` and `http://www.w3.org/2003/01/geo/wgs84_pos#lat` predicates (hereafter, for brevity `geo:long` and `geo:lat`, respectively) of the W3C Basic Geo vocabulary to express the coordinates of an instance (represented by the variable `?subject`).

```
ASK { ?subject <http://www.w3.org/2003/01/geo/wgs84_pos#long> ?x
?subject <http://www.w3.org/2003/01/geo/wgs84_pos#lat> ?y
}
```

After the available spatial datasets have been identified, we retrieve for each dataset its spatial classes by sending SELECT queries to its SPARQL endpoint. A SELECT query is another variation of SPARQL that is used to extract the raw values that answer to the given query. Specifically, we send five SELECT queries (Table 1), one for each vocabulary, to retrieve dataset classes by vocabulary. For example, the W3C Basic Geo SELECT query returns a list of the classes (variable `?class`) that contain instances (variable `?s`) using the `geo:long` and `geo:lat` predicates for expressing their location. We note that if a class uses more than one spatial vocabulary (for example, an instance is georeferenced using W3C Basic Geo and GeoRSS vocabularies), we retrieve the class once in order to avoid duplicates. Similar SELECT SPARQL queries are sent to calculate the bounding box, the number of spatial instances and other metadata of the spatial classes and datasets, which are presented in the following section.

Table 1. SELECT SPARQL queries for retrieving dataset spatial classes.

Spatial Vocabulary	SELECT Query
GeoVocab	SELECT DISTINCT ?class { ?geom <http://www.w3.org/2003/01/geo/wgs84_pos#long> ?x. ?geom <http://www.w3.org/2003/01/geo/wgs84_pos#lat> ?y. ?s <http://geovocab.org/geometry#geometry> ?geom. ?s <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> ?class}

GeoSPARQL	SELECT DISTINCT ?class {
	?s <http://www.opengis.net/ont/geosparql#hasGeometry> ?geom.
	?geom <http://www.opengis.net/ont/geosparql#asWKT> ?wkt.
	?s <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> ?class}

GeoNames	SELECT DISTINCT ?class {
	?s <http://www.w3.org/2003/01/geo/wgs84_pos#long> ?x.
	?s <http://www.w3.org/2003/01/geo/wgs84_pos#lat>?y.
	?s <http://www.geonames.org/ontology#featureClass> ?class.}

W3C Basic Geo	SELECT DISTINCT ?class {
	?s <http://www.w3.org/2003/01/geo/wgs84_pos#long> ?x .
	?s <http://www.w3.org/2003/01/geo/wgs84_pos#lat> ?y.
	?s <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> ?class.}

GeoRSS	SELECT DISTINCT ?class {
	?s <http://www.georss.org/georss/point> ?point.
	?s <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> ?class}

3.1.3. Item Metadata and GeoVoID

GeoLOD Catalog contains two main categories of items: spatial datasets and spatial classes. Spatial datasets are described by some generic metadata, namely their title, description, SPARQL endpoint URL, and VoID URL (if available), extracted from LOD cloud and DataHub metadata. Moreover, for each dataset, we compute spatial metadata, namely its bounding box, (that is, the minimum bounding rectangle (MBR) that contains all its instance locations), the number of its spatial classes and spatial instances, and the spatial vocabularies found, extracted by sending the appropriate SELECT queries (as described in Section 3.1.2). Spatial classes are described by some generic metadata, namely their URI (Uniform Resource Identifier), label, description, and the dataset that they belong to. For each class, we compute spatial metadata, namely its MBR, the number of its spatial instances, and the spatial vocabulary that it uses. Figure 1 summarizes the metadata set for GeoLOD datasets and classes and their association.

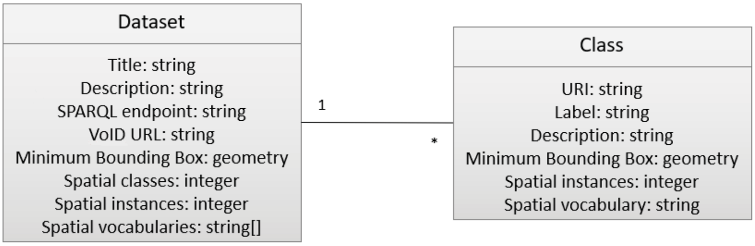

Dataset
Title: string
Description: string
SPARQL endpoint: string
VoID URL: string
Minimum Bounding Box: geometry
Spatial classes: integer
Spatial instances: integer
Spatial vocabularies: string[]
1
*
Class
URI: string
Label: string
Description: string
Minimum Bounding Box: geometry
Spatial instances: integer
Spatial vocabulary: string

Figure 1. GeoLOD Catalog item metadata. A class may belong to 1 dataset and a dataset can contain many (*) classes.

To describe spatial datasets in machine-readable format we designed and introduce GeoVoID, an RDF dataset description vocabulary that extends VoID [32] to express spatial metadata at dataset level. In VoID, a `void:Dataset` class represents the instance of a dataset, which is described by properties, such as `void:entities` (denoting the total number of its entities), `void:classes` (denoting the total number of its classes) and `void:triples` (denoting the total number of its triples). `void:classPartition` is a subset of a `void:Dataset` that contains the description of a certain `rdfs:Class`, which is declared with the property `void:class`. In GeoVoID, each `void:Dataset` class is used to describe a spatial dataset and contains a mandatory `dctetms:spatial` predicate, which denotes the dataset MBR in Well Known Text (WKT) format, which is a markup language for representing vector geometry objects. The newly defined predicates `geovoid:vocabulary`, `geovoid:classes`, and `geovoid:entities` denote the dataset spatial vocabularies, number of spatial classes, and number of spatial instances, respectively, (we remind the reader that VoID corresponding predicates are not restricted to spatial vocabularies, classes, and instances). The `void:classPartition` predicate contains the list of spatial classes of the dataset, where each spatial class is represented by the `void:class` class. Each `void:class` can also contain the `dctetms:spatial`, `geovoid:vocabulary` and `geovoid:entities` predicates to denote the corresponding spatial metadata for a class. The GeoVoID schema is available at http://snf-661343.vm.okeanos.grnet.gr/schemas/geovoid, (accessed on 16 April 2021) and its term definitions are in accordance with the definitions used in this paper; that is, a spatial entity is a georeferenced instance, a spatial class is a class containing one or more spatial instances, and a spatial vocabulary is a vocabulary that can be used for instance georeferencing.

3.2. The Recommender

The goal of the GeoLOD Recommender is to provide to each spatial class in the GeoLOD Catalog a ranked list of other spatial classes that may contain related instances, that is, instances that refer to the same real world object or to semantically close objects (e.g., a university and its campus). The recommended pairs of classes can be used as input in a link-discovery process, using tools such as SILK and LIMES, for the establishment of `owl:sameAs` links or other links (e.g., `rdf:seeAlso`) that denote a close semantic relation between instances. Recommender generates recommendation lists for all spatial classes in the background and provides them through the web interface at both class and dataset-level. In addition, it allows the on-the-fly recommendation for datasets that are not listed in the catalog and for non-RDF spatial datasets in well-known spatial data representation formats, such as Shapefile and GeoJSON.

The recommendation process implements the methodology presented in [33], which generates a ranked list of relevant classes for a link discovery process to a given class, based on the similarity of the spatial distribution of their instances. Below, we briefly present the recommendation process, which is analyzed in detail in [33]. Initially, the algorithm builds spatial summaries for each class that capture (a) its spatial extent, by calculating its ConvexHull (the minimum polygon that encloses all instance locations of the class), and (b) the spatial distribution of its instances, by overlaying them on a global pre-computed QuadTree and generating a set of QuadTree cells IDs, that consists of the QuadTree cells IDs that overlap with the instances of the class. QuadTree is a spatial index that segments the world into not-equally-sized cells (each having an ID), where small cells cover areas that present high concentration of linked data instances (such as cities) and large cells cover areas that present low concentration of linked data instances (such as oceans). The algorithm exploits above-described class summaries and computes the similarity of an input (source) class (the class for which someone wants to get recommendations) with each of the other summarized (target) classes. In order to reduce the number of similarity computations, the algorithm filters out target classes that do not spatially overlap with the source class (i.e., their ConvexHulls are disjointed), and their spatial distribution summaries do not have a minimum number of common QuadTree cell IDs (which means that the two classes share few instances in close proximity). Finally, the algorithm computes a similarity score for the source class and each of the remaining (not filtered out) target classes using one of the similarity metrics proposed in [33]: Number of Common Cells (CC), Jaccard Similarity (JS), Overlap Coefficient (OC), Poisson Distribution Probability (PD), Pointwise Mutual Information (PMI), and Phi Coefficient (PHI). The output of the algorithm is a ranked list of recommended classes to the source class for a link-discovery process. The ranking is determined by the selected metric score so that the higher the similarity between the source and a target class summary sets, the more likely for this pair of classes to contain related instances.

GeoLOD creates summaries and recommendations for all classes in the Catalog by executing the recommendation algorithm described above with the following modifications. Instead of determining the ranking based on one metric, it combines the three most effective metrics, which, according to the evaluation performed in [33], are the Poisson Distribution Probability (PD), the Pointwise Mutual Information (PMI), and the Phi Coefficient (PHI), as follows: the pairs of classes (the source and each of the target classes) are ranked three times based on the similarity score for each metric. Then, the three ranking positions for each pair are summed to compute its combined ranking. For example, if a pair of classes is ranked 1st for the PD, 6th for the PMI, and 3rd for the PHI metric, its combined ranking is 10. Finally, the combined ranking of all pairs is sorted in ascending order to generate the final ranked list of recommended classes.

To further reduce the size of the final lists of recommended classes to a source class, GeoLOD applies an additional filtering condition to exclude pairs of classes that achieve a low similarity score at least for one of the three metrics. The thresholds defined in the following condition were set empirically and are assessed in Section 5.3:

```
PD > 0.90 and PMI > 1 and PHI > 0.02
```

4. The GeoLOD Application

4.1. Implementation

GeoLOD web interface is available at http://geolod.net/ (accessed on 16 April 2021). The frontend application was developed in *React* [73] and the backend API in *Node.js* [74]. The queries to the SPARQL endpoints were sent with the *Fetch SPARQL endpoint* node.js module [75]. The thumbnails depicting the bounding box of datasets and classes were generated with the *Static Image Mapbox API* [76], and the interactive maps were built on *Leaflet* [77] and *OpenLayers* [78]. The database behind the application is the `PostgreSQL` with the *PostGIS* [79] extension for spatial data management. GeoLOD is hosted in a *Ubuntu 18 LTS 4GB* Virtual Machine, provided by *okeanos*, a GRNET cloud Infrastructure as a Service (IaaS) for Greek academic institutes.

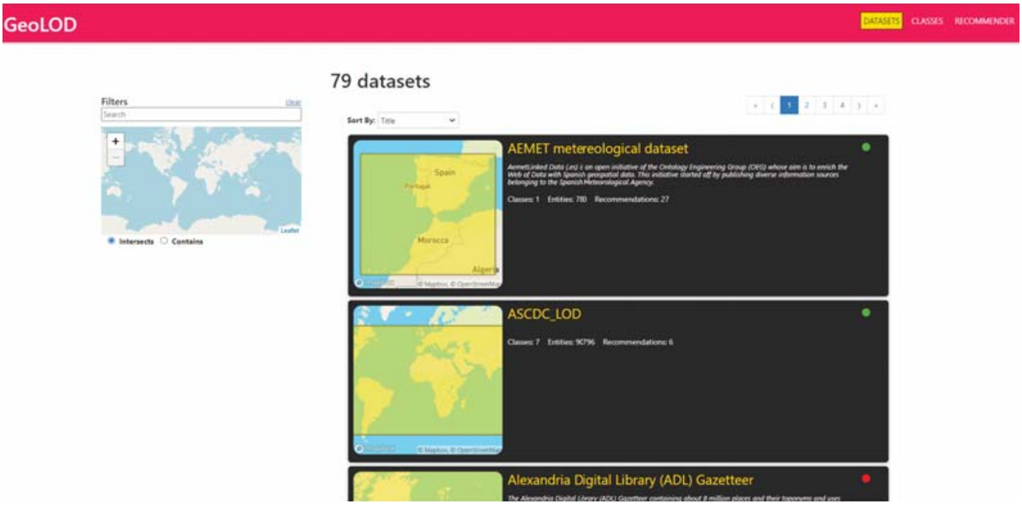
GeoLOD content, that is, the list of spatial datasets and classes with their metadata and the recommendation lists for all classes, is updated automatically every two months, as a background process. For each update, newly identified spatial datasets and classes are imported into the Catalog (according to the methods described in Section 3), and existing datasets and classes are checked for content changes and updated accordingly; for instance, if the number of a class spatial instances has changed, we update its metadata and recalculate its minimum bounding rectangle (MBR).

4.2. Use Cases

In the GeoLOD interface (Figure 2), users can browse the complete list of identified spatial datasets and classes or filter them using text and map-based criteria. Upon entering a keyword in the *Filters* dialog box, GeoLOD searches in datasets and classes titles and descriptions, and upon selecting an area in the interactive map, GeoLOD returns datasets and classes whose minimum bounding rectangles intersect or are contained in the selected area, thus allowing users to browse datasets and classes that contain instances in specific geographical areas, such as continents, countries or other user-defined areas. Additionally, users can sort the datasets and classes lists in multiple ways, including sorting by title, number of instances, and number of recommendations. Upon selecting an item (a dataset or a class), users can view its full description and perform some actions.

On a dataset description page (Figure 3), users can view its title, description, SPARQL endpoint URL, its bounding box on a thumbnail, the spatial vocabularies it uses, the number of spatial entities and classes it contains, the number of recommendations (computed as the sum of recommendations for all dataset classes) and navigate in the list of dataset spatial classes. An icon indicates whether the SPARQL endpoint is currently available (green) or unavailable (red). In addition, users can download its VoID file (if available) and export its GeoVoID description (see Section 3.1.3) and the dataset recommendations list in JSON. The latter can be used for batch link discovery processes and consists of all recommendations for dataset classes. A sample of the JSON file is depicted below: `Recommendations` is the root element, which contains an array of recommendations. Each array object (described inside `{` and `}` characters) refers to a recommendation, that is, a pair of classes, and contains the source and the target class SPARQL endpoint (properties `sourceEndpoint` and `targetEndpoint`) and URI (properties `sourceClass` and `targetClass`), respectively.

On a class description page, users can view its label, description, URI, the dataset it belongs to, its bounding box on a thumbnail, the spatial vocabulary it uses, the number of its spatial entities and the list of recommended classes and export the list of recommended classes in JSON. Furthermore, they can download live copies of class instances (extracted on the fly from the SPARQL endpoint) in RDF, JSON, and GeoJSON or browse class spatial instances on an interactive map (Figure 4). We note that the GeoJSON downloads are transformed in order to be readily consumable by a geographic information system (GIS) software, such as QGIS.



79 datasets

AEMET metereological dataset

AemetLinked Data (.es) is an open initiative of the Ontology Engineering Group (OEG) whose aim is to enrich the Web of Data with Spanish geospatial data. This initiative started off by publishing diverse information sources belonging to the Spanish Meteorological Agency.

Classes: 1 Entities: 780 Recommendations: 27

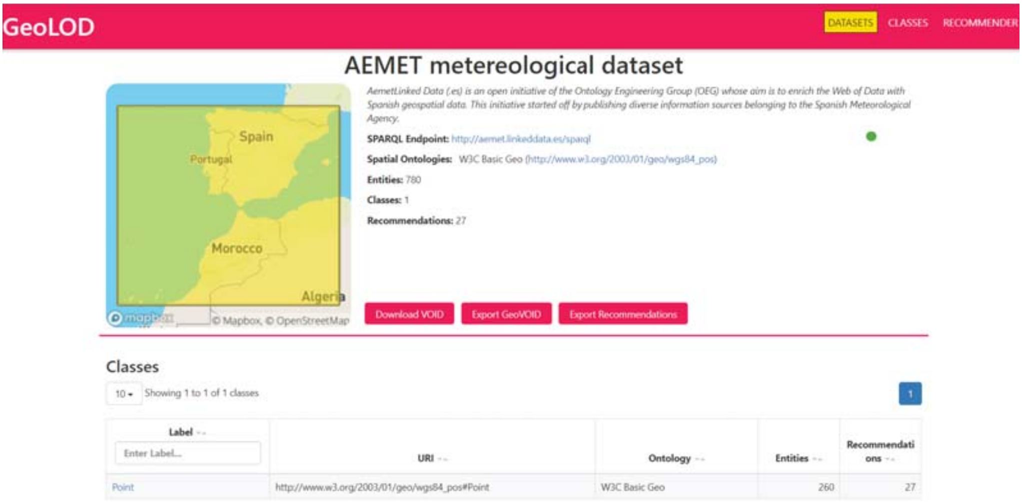
ASCDC_LOD

Classes: 7 Entities: 90796 Recommendations: 6

Alexandria Digital Library (ADL) Gazetteer

The Alexandria Digital Library (ADL) Gazetteer containing about 8 million places and their toponyms and uses

Figure 2. GeoLOD home page with the list of linked data spatial datasets.



AEMET metereological dataset

AemetLinked Data (.es) is an open initiative of the Ontology Engineering Group (OEG) whose aim is to enrich the Web of Data with Spanish geospatial data. This initiative started off by publishing diverse information sources belonging to the Spanish Meteorological Agency.

SPARQL Endpoint: http://aemet.linkeddata.es/sparql

Spatial Ontologies: W3C Basic Geo (http://www.w3.org/2003/01/geo/wgs84_pos)

Entities: 780

Classes: 1

Recommendations: 27

Classes

10 ▾ Showing 1 to 1 of 1 classes

Label Enter Label...	URI	Ontology	Entities	Recommendations
Point	http://www.w3.org/2003/01/geo/wgs84_pos#Point	W3C Basic Geo	260	27

Figure 3. The *AEMET* dataset description page.

```
{‘‘Recommendations’’:[{
‘‘sourceEndpoint’’:‘‘http://aemet.linkeddata.es/sparql’’,
‘‘sourceClass’’:‘‘http://www.w3.org/2003/01/geo/wgs84_pos#Point’’,
‘‘targetEndpoint’’:‘‘http://www.linklion.org:8890/sparql’’,
‘‘targetClass’’:‘‘http://linkedgeodata.org/ontology/AerowayThing’’
},{
‘‘sourceEndpoint’’:‘‘http://aemet.linkeddata.es/sparql’’,
‘‘sourceClass’’:‘‘http://www.w3.org/2003/01/geo/wgs84_pos#Point’’,
‘‘targetEndpoint’’:‘‘http://www.linklion.org:8890/sparql’’,
‘‘targetClass’’:‘‘http://linkedgeodata.org/ontology/Viewpoint’’
},{
...
}]}
```

A snapshot of the recommendation list for a given class (specifically, for the *Point* class of the *AEMET* dataset that contains information about meteorological stations) is depicted in Figure 5. Users can navigate through the list, view details for a recommended class, such as the number of estimated related instances and the ranking order, and export SILK and LIMES configuration files for the pair of classes for direct use in a link discovery process. The configuration files are automatically generated using as input the source (in this example *Point*) and the selected target class SPARQL endpoint URLs and URIs and configured to perform a basic instance matching that (a) "cleans" instance labels, by converting them in lower case and removing special characters, and checks for their *Levenshtein Distance*, which is a typical string similarity metric, and (b) checks the distance of instance locations using the *Euclidean Distance* metric.

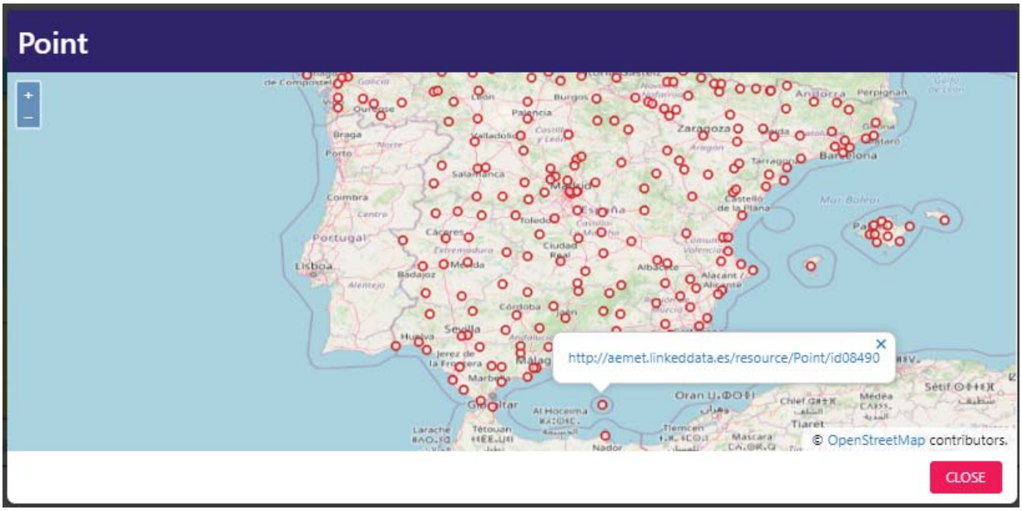
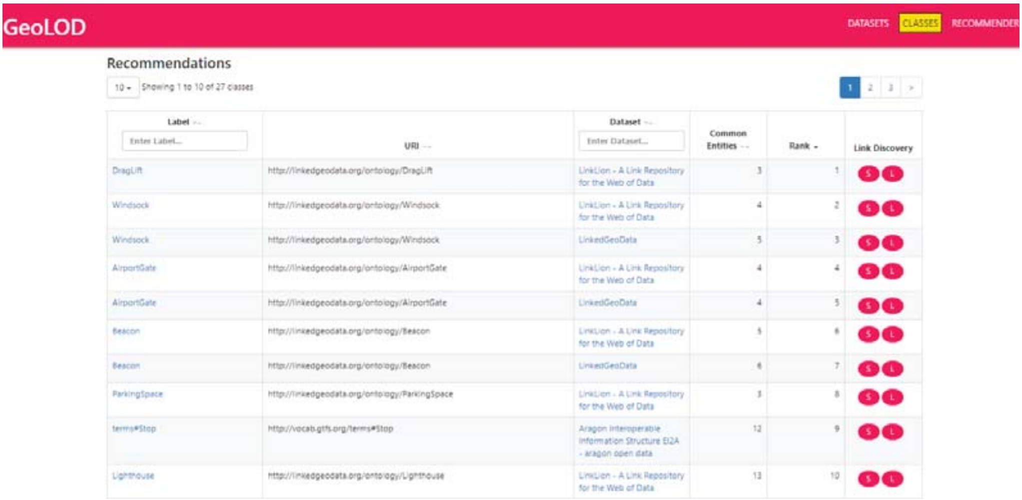

Point
+
−
http://aemet.linkeddata.es/resource/Point/id08490
© OpenStreetMap contributors.
CLOSE

Figure 4. *Point* class instances of the *AEMET* dataset on map. The user can click on an instance to get more info in a pop up.



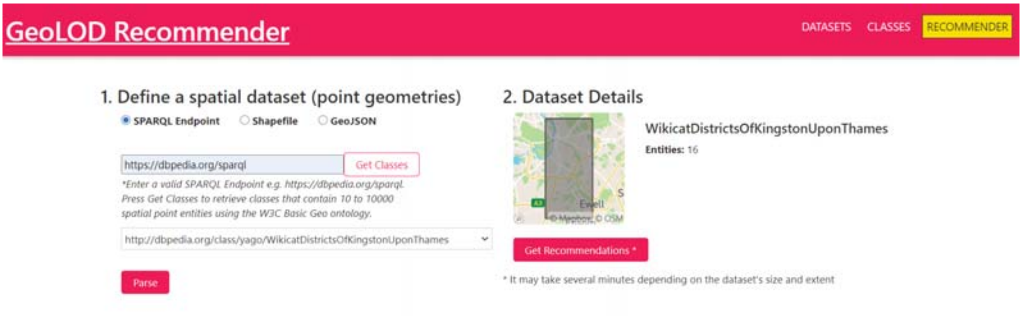
Recommendations

10 ▾ Showing 1 to 10 of 27 classes

Label Enter Label...	URI	Dataset Enter Dataset...	Common Entities	Rank	Link Discovery
DragLift	http://linkedgeodata.org/ontology/DragLift	LinkLion - A Link Repository for the Web of Data	3	1	S L
Windsock	http://linkedgeodata.org/ontology/Windsock	LinkLion - A Link Repository for the Web of Data	4	2	S L
Windsock	http://linkedgeodata.org/ontology/Windsock	LinkedGeoData	5	3	S L
AirportGate	http://linkedgeodata.org/ontology/AirportGate	LinkLion - A Link Repository for the Web of Data	4	4	S L
AirportGate	http://linkedgeodata.org/ontology/AirportGate	LinkedGeoData	4	5	S L
Beacon	http://linkedgeodata.org/ontology/Beacon	LinkLion - A Link Repository for the Web of Data	5	6	S L
Beacon	http://linkedgeodata.org/ontology/Beacon	LinkedGeoData	6	7	S L
ParkingSpace	http://linkedgeodata.org/ontology/ParkingSpace	LinkLion - A Link Repository for the Web of Data	3	8	S L
terms#Stop	http://vocab.gtfs.org/terms#Stop	Aragon Interoperable Information Structure EI2A - aragon open data	12	9	S L
Lighthouse	http://linkedgeodata.org/ontology/Lighthouse	LinkLion - A Link Repository for the Web of Data	13	10	S L

Figure 5. The ranked class recommendations list for the *Point* class of the *AEMET* dataset.

Figure 6 shows the on-the-fly recommender user interface for generating recommendations for datasets that are not listed in the GeoLOD Catalog. Initially, users select the type of the input dataset that can be a SPARQL endpoint, a GeoJSON, or a Shapefile (step 1). For the first case, they enter the URL of the endpoint and select a class from the automatically populated list; for the other cases, they upload the corresponding files. GeoLOD parses the input dataset, builds in real time the required summaries and metadata and generates a preview (step 2). Finally, users click the *Get Recommendations* button and GeoLOD searches in the Catalog to return the list of recommended classes for link discovery.



GeoLOD Recommender

1. Define a spatial dataset (point geometries)

● SPARQL Endpoint ○ Shapefile ○ GeoJSON

https://dbpedia.org/sparql

*Enter a valid SPARQL Endpoint e.g. https://dbpedia.org/sparql. Press Get Classes to retrieve classes that contain 10 to 10000 spatial point entities using the W3C Basic Geo ontology.

http://dbpedia.org/class/yago/WikicatDistrictsOfKingstonUponThames

2. Dataset Details

WikicatDistrictsOfKingstonUponThames

Entities: 16

* It may take several minutes depending on the dataset's size and extent

Figure 6. The on-the-fly recommender interface.

4.3. REST API

GeoLOD provides a REST API that can be used by software agents. Table 2 lists the names, the request URI (the left part of the URI is http://snf-661343.vm.okeanos.grnet.gr accessed on 16 April 2021), and the descriptions of the main services.

Table 2. GeoLOD REST services.

Service Name	**Request URI**	**Description**
GeoLOD Description	/api/download/dcat	Returns a DCAT-compliant turtle file that contains general information about GeoLOD and the list of the datasets in the Catalog
Dataset List	/api/datasets	Returns, in JSON, the list of datasets with their metadata (including internal dataset IDs) in the GeoLOD Catalog
Dataset Description	/api/datasets/<ID>	Returns, in JSON, the specified dataset metadata with the list of its classes. The dataset ID is a variable corresponding to the internal dataset ID. (e.g., http://snf-661343.vm.okeanos.grnet.gr/api/datasets/915 accessed on 16 April 2021, returns the metadata for the *AEMET* dataset)
Class List	/api/classes	Returns, in JSON, the list of classes with their metadata (including internal classes IDs) in the GeoLOD Catalog.
Class Description	/api/classes/<ID>	Returns, in JSON, the specified class metadata with the list of its recommended classes. The class ID is a variable corresponding to the internal class ID. (e.g., http://snf-661343.vm.okeanos.grnet.gr/api/classes/139090 accessed on 16 April 2021, returns the metadata for the *CaveEntrance* class of *Linklion* dataset).
Dataset GeoVoID	/api/download/geovoid/<ID>	Returns, in turtle format, the GeoVoID description of the specified dataset.
Dataset Recommendations	/api/download/ datasetrecommendations/<ID>	Returns, in JSON, the list of recommendations for all specified dataset classes.
Class Recommenations	api/download/ classesrecommendations/<ID>	Returns, in JSON, the list of recommendations for the specified class.

5. Results

In this Section, we present statistics that provide insights into the characteristics of spatial datasets in the Web of Data (Section 5.1) and the potential interlinkings between spatial datasets and classes based on GeoLOD recommendations (Section 5.2). In Section 5.3, we assess the applicability of the Recommender by examining the relation between GeoLOD class recommendations and LIMES instance recommendation for different algorithm variations. Finally, we present the findings of the system usability study that we performed to evaluate GeoLOD application (Section 5.4).

5.1. Catalog Statistics

In November 2020, LOD cloud and DataHub contained 478 and 723 datasets provided through SPARQL endpoints, respectively. Many datasets are listed in both catalogs, and some are provided through the same endpoint. GeoLOD identified 629 unique SPARQL endpoints from both catalogs. After sending simple SPARQL ASK queries to each (see Section 3.1.2), 477 returned an error response, such as URL unavailable or timeout, indicating that approximately only 24% of the total SPARQL endpoints found in LOD cloud and DataHub are active. Of the remaining 152 active endpoints, 60 responded true; that is, they contain a spatial vocabulary, which means that approximately 39% of the active endpoints contain georeferenced information.

In the following pages, we analyze the content of the identified spatial datasets, and we present statistics that reveal the availability and distribution of the spatial information in the Web of Data. Initially, we sent SPARQL SELECT queries to the 60 SPARQL endpoints in order to retrieve their spatial classes and collect statistics, namely, the number of its total classes, spatial classes, total instances, and spatial instances. During the investigation, we found endpoints that could not respond to the issued SELECT queries and endpoints that are duplicates or mirror other endpoints, and we excluded them from subsequent analysis. We also removed classes that contain very few instances (less than 5), because these classes cannot be used for generating recommendations, or too many instances (more than 100,000) in order to avoid high computational costs. Finally, we excluded the DBpedia dataset from our analysis, which contains 22,742 spatial classes (approximately seven times more than the sum of spatial classes of the other datasets) and more than 1 million spatial instances.

Due to the above restrictions, we finally analyzed 40 SPARQL endpoints, presented in Table 3. The total number of identified spatial classes is 3418, that is, approximately 5% of the total classes (66,571) provided by the 40 identified spatial datasets. Accordingly, we identified approximately 77 million georeferenced instances, that is, approximately 18% of the total instances (424 million) provided by the same datasets. Table 3 reveals that the biggest providers of spatial information are the *LinkedGeoData* and *Linklion* datasets, containing 952 and 902 spatial classes and more than 48 and 20 million spatial instances, respectively.

Next, we present information about the spatial characteristics of linked data datasets and classes. Table 4 presents the statistical distribution of datasets and classes by the size of their spatial extents (i.e., their mininum bounding rectangles), classified into five categories, each representing an area roughly equal to a common geographical notion, ranging from small areas, covering medium sized cities, to large areas, covering the whole world. Most datasets and classes are “global” or cover areas approximately equal to continents (about 78% of datasets and 87% of classes), which shows that most linked data providers publish large area datasets and that few providers published local datasets. Furthermore, by inspecting classes content on the GeoLOD interactive map, we noticed that in many cases, the population completeness, that is, the percentage of all real-world objects of a particular type that are represented in a class [80], regarding spatial instances at local level is small. The implication of these findings is that local mapping organizations have not yet adopted linked data technologies. Figure 7 shows the spatial extents of all spatial datasets and their density all over the world and indicates that most non-global-scale datasets are located in and around Europe. A closer examination of Figure 7 reveals potential georeferencing errors for some datasets. For example, there is a dataset that extends in a small area around zero longitude and latitude in the Gulf of Guinea at the Atlantic Ocean and another whose MBR is a thin line that starts in the Pacific Ocean, east of South America, and ends in Australia.

Table 3. Number of total and spatial classes, total and spatial instances for 40 SPARQL endpoints. N/A denotes that the number could not be retrieved because of errors returned from the endpoint.

SN	DATASET	CLASSES		INSTANCES	
		TOTAL	SPATIAL	TOTAL	SPATIAL
1	AEMET metereological dataset	35	1	N/A	260
2	AragoDBPedia - aragon open data	164	1	N/A	357,678
3	Datos.bcn.cl	500	6	5,303,750	830
4	DBpedia in Basque	223	20	1,168,342	51,547
5	DBpedia in Dutch	666	142	6,718,584	252,310
6	DBpedia in French	442	188	6,015,375	225,030
7	DBpedia in German	557	123	6,682,441	N/A
8	DBpedia in Greek	14,439	245	2,852,513	12,609
9	DBpedia in Japanese	727	100	4,254,851	36,827
10	DBpedia in Spanish	748	126	5,249,003	36,800
11	Dutch Ships and Sailors	92	11	N/A	42,810
12	El Viajero's tourism dataset	67	1	1,019,390	643
13	Environment Agency Bathing Water Quality	93	7	801,310	1216
14	European Nature Information System	629	13	N/A	1,129,574
15	European Pollutant Release and Transfer Register	375	10	78,719,353	3,325,006
16	EuroVoc	413	1	91,726,256	672
17	Geological Survey of Austria (GBA)—Thesaurus	23	1	3004	130
18	Indicators Academic Process 2017	79	7	516,097	159
19	Isidore	62	4	20,662,124	4101
20	ISPRA—The administrative divisions of Italy	99	4	449,218	23,211
21	Linked Logainm	114	38	214,423	108,065
22	LinkedGeoData	1908	952	N/A	48,249,489
23	LinkLion	1137	902	138,806,633	20,006,546
24	Lotico	23	7	N/A	N/A
25	MONDIS	662	3	12,855	10
26	MORElab	223	20	1,168,342	51,547
27	Open Data Communities—Lower layer Super Output Areas	334	14	7,912,454	2,694,723
28	OpenMobileNetwork	156	1	934,551	357,298
29	OxPoints (University of Oxford)	106	5	114,813	1457
30	Serendipity	607	109	N/A	61,845
31	Shoah victims? names	200	35	1,956,021	13,974
32	Social Semantic Web Thesaurus	521	16	14,214	54
33	Spanish Linguistic Datasets	57	1	2,977,659	764
34	Suface Forestire Mondiale 1990–2016	4188	2	187,608	100
35	TAXREF-LD: Linked Data French Taxonomic Register	1921	10	8,255,730	996
36	Test Site, LOD Lab 317	98	4	5,669,728	115,225
37	URIBurner	33,656	262	22,175,094	456,360
38	Verrijkt Koninkrijk	52	12	329,621	42,831
39	WarSampo	90	10	1,797,432	33,685
40	World War 1 as Linked Open Data	85	4	14,644	883
	SUM	66,571	3418	424,674,433	77,697,265

Table 4. Datasets and classes classified by the size of their spatial extent.

Spatial Extent (Km^2)	Datasets	Classes
City-level (<1 K)	1	54
Region level (1 K–100 K)	3	218
Country level (100 K–1000 K)	5	183
Continent level (1000 K–50,000 K)	17	1316
World level (>50,000 K)	14	1647

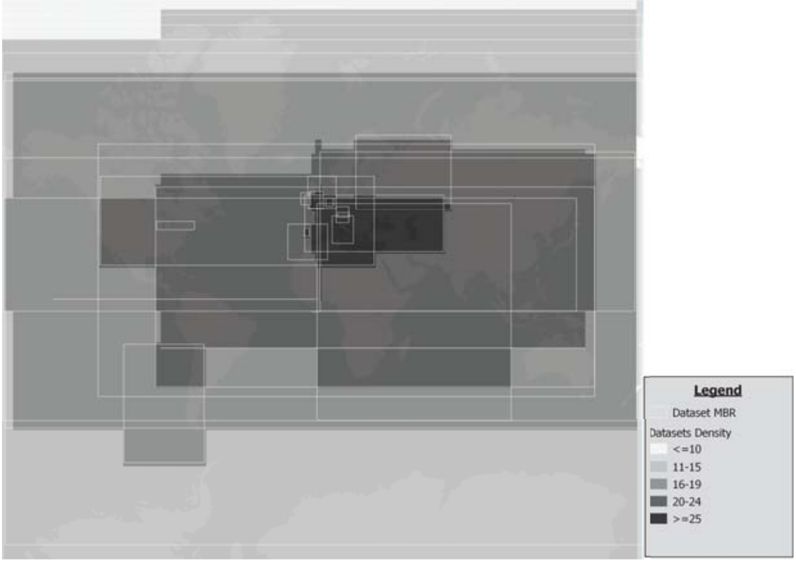


Figure 7. Spatial datasets minimum bounding rectangles and density.

We close this section by presenting two more findings. Regarding the use of spatial vocabularies, the most used spatial vocabulary is the W3C Basic Geo, which is used in all datasets (40) that were examined and in 3345 classes. Ten datasets also use the Geonames and one dataset the GeoVocab vocabularies in 36 and 37 spatial classes, respectively. GeoRSS is used with W3C Basic Geo in 15 datasets, and no dataset was found that uses the GeoSPARQL vocabulary. Concerning the availability of VoID files, of the 629 identified datasets in LOD cloud and DataHub provided through SPARQL endpoints, only 236 were found to publish a VoID description, and, of the 40 datasets listed in Table 3, the respective number is 11, which shows that providers usually do not provide VoID description of their datasets. Furthermore, in none of the provided VoID descriptions did we find information for describing the spatial aspects that we present in this paper, such as dataset bounding boxes.

5.2. Recommender Statistics

In this section, we analyze the outcome of the GeoLOD Recommender that provides insights into the potential interlinking of linked data spatial datasets and classes. In particular, we executed the recommendation algorithm for 3418 spatial classes provided by the 40 spatial datasets (Table 3) using the ranking mechanism and filtering condition presented in Section 3.2.

Table 5 presents the results of the recommendation algorithm summarized by dataset. For each dataset, it shows (a) the number of its spatial classes as listed in Table 3 (column DC), (b) the number of dataset classes for which there are recommendations (column DCR), (c) the number of recommendations to other dataset classes (column OCR), and (d) the number of recommendations to other datasets (column ODR). It is worth noting that the numbers in Table 5 refer to GeoLOD recommendations (with the specific algorithm parameters) and not to the correctly recommended classes and datasets. Furthermore, the presented statistics include only recommendations for other dataset classes and not for classes provided by the same dataset as the source class. Finally, we note that columns DCR, OCR, and ODR can be read in two ways; the number of dataset classes for which there are recommendations (column DCR) denotes the number of dataset classes for which there are recommendations to classes of other datasets (outbound recommendations) but also the number of dataset classes for which there are recommendations from classes of other datasets (inbound recommendations).

Table 5. GeoLOD Recommendations statistics (DC = Number of dataset classes, DCR = Number of dataset classes for which there are recommendations, OCR = Number of recommendations to other dataset classes, ODR = Number of recommendations to other datasets).

SN	DATASET	DC	DCR	OCR	ODR
1	AEMET metereological dataset	1	1	26	5
2	AragoDBPedia—aragon open data	1	1	31	7
3	Datos.bcn.cl	6	6	777	7
4	DBpedia in Basque	20	20	1380	19
5	DBpedia in Dutch	142	102	1620	19
6	DBpedia in French	188	144	4569	26
7	DBpedia in German	123	98	1881	22
8	DBpedia in Greek	245	174	1864	25
9	DBpedia in Japanese	100	92	2407	15
10	DBpedia in Spanish	126	116	1857	11
11	Dutch Ships and Sailors	11	11	349	15
12	El Viajero’s tourism dataset	1	1	135	11
13	Environment Agency Bathing Water Quality	7	7	125	4
14	European Nature Information System	13	13	628	23
15	European Pollutant Release and Transfer Register	10	10	891	26
16	EuroVoc	1	1	89	10
17	Geological Survey of Austria (GBA)—Thesaurus	1	1	38	4
18	Indicators Academic Process 2017	7	1	2	2
19	Isidore	4	4	146	13
20	ISPRA—The administrative divisions of Italy	4	4	200	10
21	Linked Logainm	38	31	573	13
22	LinkedGeoData	952	900	28,603	34
23	LinkLion	902	885	28,610	37
24	Lotico	7	7	455	25
25	MONDIS	3	3	18	2
26	MORElab	20	20	1380	19
27	Open Data Communities—Lower layer Super Output Areas	14	14	362	7
28	OpenMobileNetwork	1	1	117	8
29	OxPoints (University of Oxford)	5	5	48	5
30	Serendipity	109	107	2889	19
31	Shoah victims? names	35	20	643	13
32	Social Semantic Web Thesaurus	16	10	59	5
33	Spanish Linguistic Datasets	1	1	10	3
34	Suface Forestire Mondiale 1990–2016	2	0	0	0
35	TAXREF-LD: Linked Data French Taxonomic Register	10	4	30	3
36	Test Site, LOD Lab 317	4	4	58	4
37	URIBurner	262	186	3052	22
38	Verrijkt Koninkrijk	12	11	373	15
39	WarSampo	10	10	621	10
40	World War 1 as Linked Open Data	4	3	82	9
	SUM	3418	3029	86,998	530
	AVERAGE	85.45	75.73	2175.00	13.25

GeoLOD recommends one or more relevant classes for link discovery for 3029 classes, that is, for approximately 89% of all classes. This means that GeoLOD does not find recommendations for only 389 (out of 3418) classes. The 3029 classes belong to 39 different datasets, which means that for all datasets (except *Suface Forestière Mondiale 1990–2016*) GeoLOD produces recommendations. The total number of class recommendations is 86,998 (we note that class recommendations including same dataset classes is 164,782), and thus, the average classes recommendations per class is 25.45, which means that each class gets recommendations for (or from) approximately 0.75% of the total linked data classes (25.45 of 3418). At dataset level, each dataset has on average 2175 recommendations to classes of other datasets and 13.25 recommendations to other datasets, which means that each dataset gets recommendations to (or from) 13.25 other datasets, that is, approximately 33% of the total identified spatial datasets. Table 5 shows that *LinkedGeoData* and *Linklion* are hub datasets, regarding the number of recommendations they have to (or from) other datasets, having recommendations to 34 and 37 other datasets, respectively.

Regarding the execution time of the recommendation algorithm, it requires approximately one day to build summaries and 44 days to generate the recommendation lists for the 3418 classes. Thus, it requires on average 18 min to generate the recommendation list for each class, although the execution time depends on the source class size and spatial extent, ranging from a few seconds to several minutes. We note that this is also the average execution time of the GeoLOD on-the-fly recommender, which builds summaries and generates reccomendations in real time.

5.3. Recommender Applicability Assessment

In [33], we evaluated the effectiveness of the recommendation methodology that is implemented in GeoLOD, and we showed that the three most effective metrics are PD (Poisson Distribution Probability), PMI (Pointwise Mutual Information), and PHI (Phi Coefficient) and that the most effective, PD, generates ranked lists of recommended classes with 62% mean average precision, approximately 35% higher than simple baselines. In this work, we assess the benefits of employing GeoLOD Recommender as a preparatory step in link-discovery processes regarding its applicability and gains in time and we examine the effect of the ranking mechanism and the filtering condition that we presented in Section 3.2. For this reason, we execute three recommendation algorithm variations and estimate the percentage of GeoLOD recommended pairs of classes for which the LIMES link discovery framework finds possible instance links. We recall that LIMES recommends possible links between instances of two instance sets (in this case, classes), whereas GeoLOD recommends possible pairs of classes for which instance links can be recommended. Therefore, the higher the number of GeoLOD recommended pairs of classes for which LIMES recommends instance links, the higher the quality and usefulness of GeoLOD recommendations.

We execute the first (default) recommender algorithm variation as follows. We initially selected, from the list of recommendations presented in Section 5.2, a random sample of 5000 (out of the total 86,998) recommendations, that is, pairs of classes. To simplify the configuration of LIMES, we restricted on classes using the W3C Basic Geo spatial vocabulary. We then imported the sample set of recommendations as a batch process to LIMES, each configured with the corresponding source and target endpoint URL and class URI and with the following matching rule:

```
AND(levenshtein(a.rdfs:label,b.rdfs:label)|0.8, euclidean(a.slat|slong,b.tlat|tlong)|0.8)
```

that recommends a link between two instances when the *(Normalized) Levenshtein Distance* of the instances labels is greater than 0.8 and the LIMES euclidean metric of the instances location is greater than 0.8, which corresponds to a euclidean distance of 0.25 degrees, equal to 25 km at the equator in the WGS84 Coordinate Reference System. We should note that the labels' distance is measured after "cleaning" them, that is, converting them into lower case and removing special characters using the LIMES *regularAlphabet* function.

We examined two more aspects of the GeoLOD recommendations, namely, (a) the quality of Top-1 GeoLOD recommendations by importing in LIMES only the top ranked recommendations for each class and (b) the effect of the final filtering condition of the recommendation algorithm by importing in LIMES only those recommendations that satisfy the following (more strict compared to the default) condition:

```
PD>0.95 and PMI>3 and PHI>0.2
```

As baseline, we input in LIMES 5000 pairs of classes randomly selected from the GeoLOD Catalog. Since these pairs are not necessarily GeoLOD recommendations, we compare the applicability of the GeoLOD recommendations against random pairs of classes. Table 6 summarizes the experimental results for the three GeoLOD recommendation algorithm configurations and the baseline. For each, it shows the number of pairs of classes that were used as input in LIMES (column 2, LIMES executions), the number of pairs of classes for which LIMES found one or more possible instance links (that we call them hits) and its percentage to the number of LIMES executions (columns 3 and 4), and the average number of LIMES instance links recommendations per hit (column 5).

Table 6. GeoLOD recommender evaluation using LIMES.

	(2) LIMES Executions	**(3) Hits**	**(4) Hits (%)**	**(5) Average Instance Links per Hit**
(Default) GeoLOD recommendations	5000	2799	55.98%	4003
Top-1 GeoLOD recommendations	2799	1947	69.56%	9339
Strict GeoLOD recommendations	3858	2650	68.68%	13,119
Random Pairs of Classes (Baseline)	5000	344	6.88%	303

The percentage of pairs of classes for which LIMES recommends instance links for the GeoLOD class recommendations (column 4), regardless of configuration, outperforms the respective percentage of the randomly generated pairs of classes (baseline). Particularly, 55.98% of the default, 69.56% of the Top-1, and 68.68% of the strict GeoLOD recommendations contain link recommendations according to LIMES basic link specification. Strict GeoLOD recommendations present a higher percentage of hits compared to the default GeoLOD recommendations, but the recommendation list is significantly reduced (3858 recommendations compared to 86,998), which means that default GeoLOD recommendations include more false positives but, at the same time, more true positives compared to the strict GeoLOD recommendations. In the GeoLOD frontend, we use the default recommendation algorithm condition (`PD > 0.90 and PMI > 1 and PHI > 0.02`) because the recommendations are ranked and users can decide how far they want to go in the recommendation lists to find all the recommended pairs of classes for which LIMES recommends instance links. However, with minor modifications to the GeoLOD fronted, users could select between a strict or loose filtering condition.

We should note that if, for a pair of classes, LIMES recommends one or more instances' links, this does not necessarily mean that this pair of classes indeed contain related instances. Conducting rigorous experiments to evaluate the quality of LIMES recommendations, that is, whether instance link recommendations truly correspond to related instances, is out of the scope of this paper. Nevertheless, we can assume that if a pair of classes contains many LIMES instance link recommendations, it is more possible to truly contain related instances than a pair of classes with few LIMES instance link recommendations. Based on the above assumption, we compare the GeoLOD recommendation algorithm variations by examining the average number of instances links recommendations per relevant pair of classes. Table 6 shows that for random pairs of spatial classes the average number of LIMES instances links recommendations per pair (column 5) is 303, while for GeoLOD recommended pairs, the respective number is much higher for all GeoLOD recommendations configurations. Specifically, the highest average is achieved by the strict variation, presenting 13,119 instance links recommendations per pair of recommended classes. Therefore, we can conclude that GeoLOD (especially, the strict variation) is more likely to recommend pairs of classes that truly contain related instances than the random baseline.

Finally, we discuss the search space reduction of the GeoLOD Recommender and the time saved when it is used as a preparatory step of a link-discovery process. The number of pairwise class comparisons needed for finding all possible instance links for all identified spatial classes is 3418 × 3418 = 11,682,724. GeoLOD generates approximately 165,360 recommendations (including classes from same datasets), and thus reduces the search space approximately 70 times. In our experiments, LIMES required approximately one hour to compare 1000 pairs of classes for instance link recommendations, and thus, to compare all possible pairs of classes in GeoLOD Catalog, LIMES requires 486 days (with 6.88% probability of finding a pair of classes with links), while comparing the GeoLOD recommended pairs requires 7 days (with 55.98% probability of finding a pair of classes with links). For a single class, the execution time for instance link discovery is approximately 3.5 h (for examining 3418 pairs of classes), while, using the on-the-fly GeoLOD Recommender, it is 18 min (the average time GeoLOD requires to generate recommendations for a single class) plus, on average, 3 min (for the 50 recommended pairs of classes, the average GeoLOD recommendations per class including same dataset recommendations, comparisons in LIMES), that is, approximately 21 min.

5.4. Usability Study

As already stated, GeoLOD user interface mainly targets linked data experts and GIS professionals in order to facilitate them during their linked data exploration and link-discovery processes. For this reason, we conducted a system usability study to assess how each category of users perceives GeoLOD and to identify strong and weak features in order to improve the application. The study is based on the System Usability Scale (SUS) [81], which consists of 10 questions to be rated on a five-point scale ranging from strongly disagree to strongly agree, among which five are positive statements and the remaining are negative. An adjective rating was added as an eleventh question to collect user ratings of the perceived usability according to a seven-point scale with different wordings [82]. The participants were selected to be experts in either linked data, GIS, or both domains. Initially, invitations were sent to academia and business people with known experience in these domains, and those who responded positively participated on a voluntary basis. The study was completed in two web sessions, held on different days, allowing the participants to choose based on their availability. At the beginning of each session, we explained the purpose of the study and briefly introduced the GeoLOD application. Then, participants had some time to get familiar with the application and to execute some indicative tasks, such as:

- Search for datasets that contain data in a specific geographic area (e.g., Spain);
- View the description and the list of classes of a dataset of their choice;
- View the description and the list of recommendations of a class of their choice;
- View the instances of their selected class on the map;
- Get recommendations for a uncatalogued endpoint, for example `https://dbpedia.org/sparql` (only for linked data experts);
- Get recommendations for a shapefile that they own (only for GIS experts).

Finally, participants completed the online SUS with an adjective rating questionnaire. Each session concluded with a short discussion, where participants expressed their general comments and proposals for the improvement of the application.

In the study, in total, 41 users participated; 11 users perceived themselves as linked data experts and 30 as GIS experts. Of the 41 users, only four declared that they are experts on both domains. Table 7 summarizes the results of the usability study. The first two rows show the results for each category of users and the last row contains the total results. The mean SUS score indicates the overall level of usability, where the minimum possible score is 0 and the maximum possible is 100. The mean SUS score for all participants is 68.48, and the respective score for linked data experts is higher (81.36) than for GIS experts (63.75). The adjective rating corresponds to the results of the 11th seven-scale question "Overall, I would rate the user-friendliness of this product as:", where 1 means `Worst Imaginable` and 7 `Best Imaginable`. The mean adjective rating for all participants is 5.17, and the respective rating for linked data experts is also higher (5.64) than of GIS experts (5.00).

Table 8 and Figure 8 present with more analysis the results of the SUS and the adjective rating questionnaire per question and user category.

Table 7. Standard Usability Scale (SUS) with adjective rating questionnaire results.

Focus Group	Participants	Min SUS	Max SUS	Mean SUS	Mean Adj. Rating (1–7)
Linked Data experts	11	55	100	81.36	5.64
GIS experts	30	40	97.5	63.75	5.00
All	41	40	100	68.48	5.17

Table 8. Standard Usability Scale (SUS) questionnaire results per question in the scale 1 (Strongly disagree) to 5 (Strongly agree).

Question	Linked Data Experts	GIS Experts	All
1. I think that I would like to use this system frequently.	3.82	3.03	3.24
2. I found the system unnecessarily complex.	1.64	2.07	1.95
3. I thought the system was easy to use.	4.27	3.33	3.59
4. I think that I would need the support of a technical person to be able to use this system.	2.09	2.30	2.24
5. I found the various functions in the system were well integrated.	4.18	3.77	3.88
6. I thought there was too much inconsistency in this system.	1.45	2.10	1.93
7. I imagine that most people would learn to use this system very quickly.	4.55	3.33	3.66
8. I found the system very awkward to use.	1.82	1.83	1.83
9. I felt very confident using the system.	4.18	3.30	3.54
10. I needed to learn a lot of things before I could get going with this system.	1.45	2.97	2.56

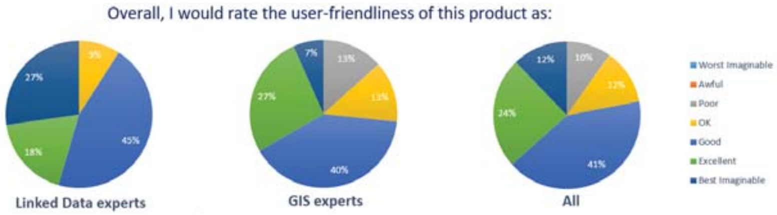


Figure 8. Adjective ratings per user category: Linked data experts (**left**), GIS experts (**center**), all (**right**).

The results of the study indicate that the opinion of the users regarding GeoLOD usability and friendliness is good and almost excellent among linked data experts. Furthermore, the responses to the first question of the SUS questionnaire shows that they believe that the application is useful. During the discussion, it emerged that users, especially those who were not linked data experts, would like more guidance (e.g., by including tooltips or explanatory text in the user interface), since they are not familiar with some terms, such as VoID and SPARQL endpoint. Some other proposals included the improvement of the on-the-fly recommender response times, responsiveness for mobile devices, and inclusion of datasets that contain polygon geometries.

6. Discussion and Conclusions

In this paper, we presented GeoLOD, a web catalog of spatial linked data datasets and classes and a recommender for datasets and classes that may contain related spatial instances. GeoLOD addresses user needs for linked data search, taking into account the spatial characteristics of datasets, and is the first exhaustive catalog and recommender exclusively for spatial datasets and classes. It provides a user-friendly interface and an API for automated content consumption. It currently contains metadata for 79 spatial datasets and 5130 spatial classes, identified by parsing the LOD cloud and DataHub catalogs. It also provides more than 166,000 recommendations for pairs of classes that may contain the same or closely related instances and an on-the-fly recommender for user-submitted SPARQL endpoints and spatial datasets in GeoJSON and Shapefile formats. The catalog and the recommendations lists are updated in the background every two months.

GeoLOD is compliant with the linked data standards for describing catalogs and datasets, providing its content in DCAT and datasets descriptions in GeoVoID. GeoVoID was introduced in this paper and extends VoID to describe spatial characteristics of datasets. In the results section, we have presented statistics about the availability of SPARQL endpoints and VoID descriptions that confirm other recent studies [25,26,51,83]; few datasets are accompanied by their VoID descriptions, and furthermore, there is no description of their spatial characteristics, such as their bounding box or the number of their georeferenced instances. GeoLOD fills this gap by automatically generating GeoVoID descriptions for each dataset in the Catalog. Our analysis reveals that most spatial datasets and classes are published by global data providers (such as DBpedia, LinkedGeoData, and Linklion) and cover the whole or large areas of the world. The study of linked data spatial characteristics reveals georeferencing errors or generalizations, including misplaced instances, the “null island” effect (instances located at zero longitude and latitude), the representation of large-area objects (e.g., countries) with points and low population completeness [80] regarding georeferenced instances (e.g., a class about airports contains a random subset of the existing airports). A study of systematic errors and their causes in geographic linked data [84] reveals that about 10% of all spatial data on the linked data cloud are erroneous to some degree. These errors could be minimized if local mapping organizations or agencies participated more actively in the linked data domain since they usually possess complete and high-quality spatial datasets. Some reasons that may prevent their engagement with linked data could be the absence or immaturity of linked data publishing tools and the subsequent high barriers for publishing spatial linked data. One of GeoLOD’s goals is to provide an easy-to-use tool that could help users, who are not linked data experts, to get familiar with the linked data landscape and thus to lower the barrier for data publishing. As the usability study indicates, users from the geospatial domain are positive about adopting GeoLOD; however they would like a more user-friendly interface regarding the explanation of terms unknown to them.

GeoLOD includes three innovative features regarding dataset interlinking: (a) a complete list of recommendations for pairs of classes that may include related instances, (b) an on-the-fly recommender for uncatalogued SPARQL endpoints and non-RDF spatial datasets, and (c) automatic generation of SILK and LIMES configuration files. These features help users to discover links between related instances, thus fulfilling the fourth linked data principle, which suggests the establishment of links between related instances so that users can discover related things. In the results, we showed the benefits of employing GeoLOD Recommender as a preparatory step for link-discovery processes. It recommends pairs of classes with 55.98% probability to contain link recommendations between class instances, using a basic link specification in LIMES, while the corresponding probability for random pairs of linked data classes is 6.88%. Furthermore, it reduces the search space for looking in the Web of Data for candidate classes that can be used as input in link discovery processes 70 times.

We conclude the paper by pointing the future work on GeoLOD. Firstly, the user interface can be improved in terms of providing more help to the users. The catalog can be populated with more content, including spatial datasets that are provided through RDF dumps, listed in other data catalogs (such as LOD Laundromat), using other well-known spatial vocabularies and expressing instance location with line or polygon geometries in various coordinate reference systems. The on-the-fly recommender can be extended to support SPARQL endpoints that use additional spatial vocabularies (other than W3C Basic Geo) and additional spatial data formats, such as the Web Feature Service (WFS) [85]. We plan to take action and conduct experiments to fine-tune the recommendation algorithm's filtering and thresholds criteria and further reduce its overall execution time. Other ideas include the involvement of GeoLOD users so as to provide feedback about "good" or "bad" recommendations and the exploitation of SILK/LIMES web services for instant instance links recommendations.

Author Contributions: Conceptualization, V.K.; methodology, V.K.; software, V.K.; validation, V.K. and M.V.; formal analysis, V.K. and M.V.; investigation, V.K.; resources, V.K.; data curation, V.K.; writing—original draft preparation, V.K.; writing—review and editing, V.K. and M.V.; visualization, V.K.; supervision, M.V.; project administration, M.V.; funding acquisition, V.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research is being supported by the funding program "YPATIA" of the University of the Aegean.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data supporting reported results can be found at GeoLOD website at http://geolod.net/ accessed on 16 April 2021.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Berners-Lee, T. Linked Data—Design Issues. Available online: https://www.w3.org/DesignIssues/LinkedData.html (accessed on 24 December 2020).
2. Unger, C.; Freitas, A.; Cimiano, P. An introduction to question answering over linked data. In *Reasoning Web International Summer School*; Springer: Berlin/Heidelberg, Germany, 2014; pp. 100–140.
3. Lopez, V.; Unger, C.; Cimiano, P.; Motta, E. Evaluating question answering over linked data. *J. Web Semant.* **2013**, *21*, 3–13. [CrossRef]
4. Höffner, K.; Walter, S.; Marx, E.; Usbeck, R.; Lehmann, J.; Ngonga Ngomo, A.C. Survey on challenges of Question Answering in the Semantic Web. *Semant. Web* **2017**, *8*, 895–920. [CrossRef]
5. Dimitrakis, E.; Sgontzos, K.; Tzitzikas, Y. A survey on question answering systems over linked data and documents. *J. Intell. Inf. Syst.* **2020**, *55*, 233–259. [CrossRef]
6. Saleem, M.; Khan, Y.; Hasnain, A.; Ermilov, I.; Ngonga Ngomo, A.C. A fine-grained evaluation of SPARQL endpoint federation systems. *Semant. Web* **2014**. [CrossRef]
7. Vidal, M.E.; Castillo, S.; Acosta, M.; Montoya, G.; Palma, G. On the Selection of SPARQL Endpoints to Efficiently Execute Federated SPARQL Queries. In *Transactions on Large-Scale Data- and Knowledge-Centered Systems XXV*; Hameurlain, A., Kung, J., Wagner, R., Eds.; Springer: Berlin/Heidelberg, Germany, 2016; pp. 109–149.
8. Harth, A.; Hose, K.; Karnstedt, M.; Polleres, A.; Sattler, K.U.; Umbrich, J. Data Summaries for On-demand Queries over Linked Data. In *Proceedings of the 19th International Conference on World Wide Web, WWW '10, Raleigh, NC, USA, 26–30 April 2010*; ACM: New York, NY, USA, 2010; pp. 411–420. [CrossRef]
9. Quilitz, B.; Leser, U. Querying Distributed RDF Data Sources with SPARQL. In *The Semantic Web: Research and Applications*; Bechhofer, S., Hauswirth, M., Hoffmann, J., Koubarakis, M., Eds.; Springer: Berlin/Heidelberg, Germany, 2008; pp. 524–538.
10. Schwarte, A.; Haase, P.; Hose, K.; Schenkel, R.; Schmidt, M. FedX: Optimization Techniques for Federated Query Processing on Linked Data. In Proceedings of the International Semantic Web Conference, Bonn, Germany, 23–27 October 2011.
11. Görlitz, O.; Staab, S. SPLENDID: SPARQL Endpoint Federation Exploiting VOID Descriptions. In Proceedings of the Second International Conference on Consuming Linked Data—Volume 782, COLD'11, Heraklion, Greece, 23 October 2011; pp. 13–24.
12. DBpedia. Available online: https://wiki.dbpedia.org/ (accessed on 20 March 2021).

13. MusicBrainz. Available online: https://musicbrainz.org/ (accessed on 20 March 2021).
14. GeoNames. Available online: https://www.geonames.org (accessed on 20 March 2021).
15. Adida, B.; Birbeck, M.; McCaron, S.; Herman, I. RDFa Core 1.1—Third Edition. 2014. Available online: https://www.w3.org/TR/rdfa-core/ (accessed on 24 December 2020).
16. Oren, E.; Delbru, R.; Catasta, M.; Cyganiak, R.; Stenzhorn, H.; Tummarello, G. Sindice.com: A document-oriented lookup index for open linked data. *IJMSO* **2008**, *3*, 37–52. [CrossRef]
17. Harth, A.; Hogan, A.; Umbrich, J.; Kinsella, S.; Polleres, A.; Decker, S. Searching and Browsing Linked Data with SWSE. In *Semantic Search over the Web*; Virgilio, R.D., Guerra, F., Velegrakis, Y., Eds.; Data-Centric Systems and Applications; Springer: Berlin/Heidelberg, Germany, 2012; pp. 361–414.
18. The Linked Open Data Cloud. Available online: https://lod-cloud.net/ (accessed on 20 March 2021).
19. DataHub. Available online: https://old.datahub.io (accessed on 20 March 2021).
20. Volz, J.; Bizer, C.; Gaedke, M.; Kobilarov, G. Silk—A Link Discovery Framework for the Web of Data. In Proceedings of the LDOW, Madrid, Spain, 20 April 2009; Bizer, C., Heath, T., Berners-Lee, T., Idehen, K., Eds.; Volume 538.
21. Ngonga Ngomo, A.C.; Sherif, M.A.; Georgala, K.; Hassan, M.; Dreßler, K.; Lyko, K.; Obraczka, D.; Soru, T. LIMES–A Framework for Link Discovery on the Semantic Web. *Künstl. Intell.* **2018**. [CrossRef]
22. Nikolov, A.; d'Aquin, M. Identifying relevant sources for data linking using a semantic web index. In Proceedings of the WWW2011 Workshop: Linked Data on the Web (LDOW 2011) at 20th International World Wide Web Conference (WWW 2011), Hyderabad, India, 29 March 2011.
23. Caraballo, A.A.M.; Arruda, N.M.; Nunes, B.P.; Lopes, G.R.; Casanova, M.A. TRTML—A Tripleset Recommendation Tool Based on Supervised Learning Algorithms. In *Proceedings of the Semantic Web: ESWC 2014 Satellite Events, Anissaras, Crete, Greece, 25–29 May 2014*; Presutti, V., Blomqvist, E., Troncy, R., Sack, H., Papadakis, I., Tordai, A., Eds.; Springer International Publishing: Cham, Switzerland, 2014; pp. 413–417.
24. Molli, P.; Skaf-Molli, H.; Grall, A. *SemCat: Source Selection Services for Linked Data*; Research Report; Universite de Nantes: Nantes, France, 2020.
25. Schmachtenberg, M.; Bizer, C.; Paulheim, H. Adoption of the Linked Data Best Practices in Different Topical Domains. In *Proceedings of the Semantic Web—ISWC 2014, Riva del Garda, Italy, 19–23 October 2014*; Mika, P., Tudorache, T., Bernstein, A., Welty, C., Knoblock, C., Vrandečić, D., Groth, P., Noy, N., Janowicz, K., Goble, C., Eds.; Springer International Publishing: Cham, Switzerland, 2014; pp. 245–260.
26. Polleres, A.; Kamdar, M.R.; Fernández, J.D.; Tudorache, T.; Musen, M.A. A more decentralized vision for Linked Data. *Semant. Web* **2020**, *11*, 101–113. [CrossRef]
27. LOD Laundromat | SEMANTiCS 2018. Available online: https://2018.semantics.cc/lod-laundromat (accessed on 20 March 2021).
28. Röder, M.; Ngonga Ngomo, A.C.; Ermilov, I.; Both, A. Detecting Similar Linked Datasets Using Topic Modelling. In *Proceedings of the 13th International Conference on The Semantic Web. Latest Advances and New Domains—Volume 9678, Heraklion, Greece, 29 May–2 June 2016*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 3–19.
29. Glaser, H.; Jaffri, A.; Millard, I. Managing Co-reference on the Semantic Web. In Proceedings of the WWW2009 Workshop: Linked Data on the Web (LDOW2009), Madrid, Spain, 20 April 2009.
30. Nentwig, M.; Soru, T.; Ngomo, A.C.N.; Rahm, E. LinkLion: A Link Repository for the Web of Data. In *Proceedings of the ESWC (Satellite Events), Anissaras, Greece, 25–29 May 2014*; Presutti, V., Blomqvist, E., Troncy, R., Sack, H., Papadakis, I., Tordai, A., Eds.; Springer: Berlin/Heidelberg, Germany, 2014; Volume 8798, pp. 439–443.
31. Mountantonakis, M.; Tzitzikas, Y. LODsyndesis: Global Scale Knowledge Services. *Heritage* **2018**, *1*, 23. [CrossRef]
32. Alexander, K.; Cyganiak, R.; Hausenblas, M.; Zhao, J. Describing Linked Datasets—On the Design and Usage of voiD, the 'Vocabulary of Interlinked Datasets'. In Proceedings of the WWW 2009 Workshop: Linked Data on the Web (LDOW2009), Madrid, Spain, 20 April 2009.
33. Kopsachilis, V.; Vaitis, M.; Mamoulis, N.; Kotzinos, D. Recommending Geo-semantically Related Classes for Link Discovery. *J. Data Semant.* **2021**, *9*, 151–177. [CrossRef]
34. W3C. Data Catalog Vocabulary (DCAT). Available online: https://www.w3.org/TR/2020/SPSD-vocab-dcat-20200204/ (accessed on 20 March 2021).
35. Akar, Z.; Halaç, T.G.; Ekinci, E.E.; Dikenelli, O. Querying the Web of Interlinked Datasets using VOID Descriptions. In Proceedings of the CEUR Workshop Proceedings, LDOW, Lyon, France, 16 April 2012; Bizer, C., Heath, T., Berners-Lee, T., Hausenblas, M., Eds.; Volume 937.
36. Böhm, C.; Lorey, J.; Naumann, F. Creating voiD descriptions for Web-scale data. *J. Web Semant.* **2011**, *9*, 339–345. [CrossRef]
37. voiD Store. Available online: http://void.rkbexplorer.com/ (accessed on 20 March 2021).
38. Langegger, A.; Woss, W. RDFStats—An Extensible RDF Statistics Generator and Library. In Proceedings of the 2009 20th International Workshop on Database and Expert Systems Application, Linz, Austria, 31 August–4 September 2009; pp. 79–83. [CrossRef]
39. Khatchadourian, S.; Consens, M. ExpLOD: Summary-Based Exploration of Interlinking and RDF Usage in the Linked Open Data Cloud. In *The Semantic Web: Research and Applications*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 272–287.

40. Demter, J.; Auer, S.; Martin, M.; Lehmann, J. LODStats—An Extensible Framework for High-performance Dataset Analytics. In *Proceedings of the EKAW 2012, Galway City, Ireland, 8–12 October 2012*; Lecture Notes in Computer Science (LNCS); Springer: Berlin/Heidelberg, Germany, 2012; p. 7603, 29% acceptance rate.
41. Mäkelä, E. Aether—Generating and Viewing Extended VoID Statistical Descriptions of RDF Datasets. In *Proceedings of the Semantic Web: ESWC 2014 Satellite Events, Anissaras, Greece, 25–29 May 2014*; Presutti, V., Blomqvist, E., Troncy, R., Sack, H., Papadakis, I., Tordai, A., Eds.; Springer International Publishing: Cham, Switzerland, 2014; pp. 429–433.
42. Mihindukulasooriya, N.; Poveda-Villalón, M.; García-Castro, R.; Gómez-Pérez, A. Loupe—An Online Tool for Inspecting Datasets in the Linked Data Cloud. In Proceedings of the International Semantic Web Conference (Posters and Demos), Bethlehem, PA, USA, 11–15 October 2015; Villata, S., Pan, J.Z., Dragoni, M., Eds.; Volume 1486.
43. Palmonari, M.; Rula, A.; Porrini, R.; Maurino, A.; Spahiu, B.; Ferme, V. ABSTAT: Linked Data Summaries with ABstraction and STATistics. In *Proceedings of the ESWC (Satellite Events), Portoroz, Slovenia, 31 May–4 June 2015*; Gandon, F., Guéret, C., Villata, S., Breslin, J.G., Faron-Zucker, C., Zimmermann, A., Eds.; Springer: Berlin/Heidelberg, Germany, 2015; Volume 9341, pp. 128–132.
44. Abedjan, Z.; Grütze, T.; Jentzsch, A.; Naumann, F. Profiling and mining RDF data with ProLOD++. In Proceedings of the ICDE, Chicago, IL, USA, 31 March–4 April 2014; Cruz, I.F., Ferrari, E., Tao, Y., Bertino, E., Trajcevski, G., Eds.; pp. 1198–1201.
45. Benedetti, F.; Bergamaschi, S.; Po, L. Visual Querying LOD sources with LODeX. In Proceedings of the 8th International Conference on Knowledge Capture, K-CAP, Palisades, NY, USA, 7–10 October 2015; Barker, K., Gómez-Pérez, J.M., Eds.; pp. 12:1–12:8.
46. Neto, C.B.; Kontokostas, D.; Hellmann, S.; Müller, K.; Brümmer, M. LODVader: An Interface to LOD Visualization, Analytics and DiscovERy in Real-time. In Proceedings of the 25th WWW Conference, Montreal, QC, Canada, 11–15 April 2016.
47. Pietriga, E.; Gözükan, H.; Appert, C.; Destandau, M.; Cebiric, S.; Goasdoué, F.; Manolescu, I. Browsing Linked Data Catalogs with LODAtlas. In *Proceedings of the International Semantic Web Conference (2), Monterey, CA, USA, 8–12 October 2018*; Vrandecic, D., Bontcheva, K., Suárez-Figueroa, M.C., Presutti, V., Celino, I., Sabou, M., Kaffee, L.A., Simperl, E., Eds.; Springer: Berlin/Heidelberg, Germany, 2018; Volume 11137, pp. 137–153.
48. DCMI Metadata Terms. Available online: https://www.dublincore.org/specifications/dublin-core/dcmi-terms/ (accessed on 20 March 2021).
49. Beek, W.; Rietveld, L.; Bazoobandi, H.R.; Wielemaker, J.; Schlobach, S. LOD Laundromat: A Uniform Way of Publishing Other People's Dirty Data. In *Proceedings of the International Semantic Web Conference (1), Riva del Garda, Trento, Italy, 19–23 October 2014*; Mika, P., Tudorache, T., Bernstein, A., Welty, C., Knoblock, C.A., Vrandecic, D., Groth, P., Noy, N.F., Janowicz, K., Goble, C.A., Eds.; Springer: Berlin/Heidelberg, Germany, 2014; Volume 8796, pp. 213–228.
50. Vandenbussche, P.Y.; Umbrich, J.; Matteis, L.; Hogan, A.; Aranda, C.B. SPARQLES: Monitoring public SPARQL endpoints. *Semant. Web* **2017**, *8*, 1049–1065. [CrossRef]
51. Hasnain, A.; Mehmood, Q.; e Zainab, S.S.; Hogan, A. SPORTAL: Profiling the Content of Public SPARQL Endpoints. *Int. J. Semant. Web Inf. Syst.* **2016**, *12*, 134–163. [CrossRef]
52. Baron Neto, C.; Kontokostas, D.; Kirschenbaum, A.; Publio, G.; Esteves, D.; Hellmann, S. IDOL: Comprehensive & Complete LOD Insights. In Proceedings of the 13th International Conference on Semantic Systems (SEMANTiCS 2017), Amsterdam, The Netherlands, 11–14 September 2017.
53. re3data.org. Available online: http://re3data.org (accessed on 20 March 2021).
54. Hasnain, A.; Decker, S.; Deus, H.F. Cataloguing and Linking Life Sciences LOD Cloud. 2012. Available online: https://aran.library.nuigalway.ie/bitstream/handle/10379/4841/Cataloguing_and_linking_Life_Sciences_LOD_cloud%28Final_Resubmission%29.pdf?sequence=1&isAllowed=y (accessed on 20 March 2021).
55. Umaka Data. Available online: https://yummydata.org/ (accessed on 20 March 2021).
56. Nentwig, M.; Hartung, M.; Ngonga Ngomo, A.C.; Rahm, E. A survey of current Link Discovery frameworks. *Semant. Web* **2015**, *8*, 419–436. [CrossRef]
57. Leme, L.A.P.P.; Lopes, G.R.; Nunes, B.P.; Casanova, M.A.; Dietze, S. Identifying Candidate Datasets for Data Interlinking. In *Proceedings of the Web Engineering, Aalborg, Denmark, 8–12 July 2013*; Daniel, F., Dolog, P., Li, Q., Eds.; Springer: Berlin/Heidelberg, Germany, 2013; pp. 354–366.
58. Ben Ellefi, M.; Bellahsene, Z.; Dietze, S.; Todorov, K. Dataset Recommendation for Data Linking: An Intensional Approach. In *Proceedings of the 13th International Conference on The Semantic Web. Latest Advances and New Domains–Volume 9678, Heraklion, Crete, Greece, 29 May–2 June 2016*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 36–51.
59. Mehdi, M.; Iqbal, A.; Hogan, A.; Hasnain, A.; Khan, Y.; Decker, S.; Sahay, R. Discovering Domain-specific Public SPARQL Endpoints: A Life-sciences Use-case. In *Proceedings of the 18th International Database Engineering & Applications Symposium, Porto, Portugal, July 14; IDEAS '14*; ACM: New York, NY, USA, 2014; pp. 39–45. [CrossRef]
60. Emaldi, M.; Corcho, Ó.; de Ipiña, D.L. Detection of Related Semantic Datasets Based on Frequent Subgraph Mining. *IESD@ISWC* **2014**, *5*, 7.
61. Liu, H.; Wang, T.; Tang, J.; Ning, H.; Wei, D.; Xie, S.; Liu, P. Identifying Linked Data Datasets for sameAs Interlinking Using Recommendation Techniques. In *Proceedings of the Web-Age Information Management, Nanchang, China, 3–5 June 2016*; Cui, B., Zhang, N., Xu, J., Lian, X., Liu, D., Eds.; Springer International Publishing: Cham, Switzerland, 2016; pp. 298–309.
62. Mountantonakis, M.; Tzitzikas, Y. Scalable Methods for Measuring the Connectivity and Quality of Large Numbers of Linked Datasets. *J. Data Inf. Qual.* **2018**, *9*, 15:1–15:49. [CrossRef]

63. Mera Caraballo, A.A.; Nunes, B.P.; Lopes, G.R.; Paes Leme, L.A.P.; Casanova, M.A.; Dietze, S. TRT—A Tripleset Recommendation Tool. In Proceedings of the 12th International Semantic Web Conference (ISWC2013), Sydney, Australia, 21–25 October 2013.
64. Wagner, A.; Haase, P.; Rettinger, A.; Lamm, H. Entity-Based Data Source Contextualization for Searching the Web of Data. In *Proceedings of the Semantic Web: ESWC 2014 Satellite Events, Anissaras, Greece, 25–29 May 2014*; Presutti, V., Blomqvist, E., Troncy, R., Sack, H., Papadakis, I., Tordai, A., Eds.; Springer International Publishing: Cham, Switzerland, 2014; pp. 25–41.
65. Group, W.S.W.I. Basic Geo (WGS84 Lat/Long) Vocabulary. Available online: https://www.w3.org/2003/01/geo (accessed on 20 March 2021).
66. GeoVocab.org. Available online: http://geovocab.org/ (accessed on 20 March 2021).
67. OGC. GeoSPARQL—A Geographic Query Language for RDF Data. Available online: https://www.ogc.org/standards/geosparql (accessed on 20 March 2021).
68. GeoNames Ontology. Available online: http://www.geonames.org/ontology (accessed on 20 March 2021).
69. W3C Geospatial Vocabulary. Available online: https://www.w3.org/2005/Incubator/geo/XGR-geo (accessed on 20 March 2021).
70. Linked Open Vocabularies. Available online: https://lov.linkeddata.es/dataset/lov (accessed on 20 March 2021).
71. Linked Open Vocabularies for Internet of Things (IoT). Available online: https://lov4iot.appspot.com/?p=lov4iot-location (accessed on 20 March 2021).
72. CKAN API Guide. Available online: https://docs.ckan.org/en/2.9/api/ (accessed on 20 March 2021).
73. React A JavaScript Library for Building User Interfaces. Available online: https://reactjs.org/ (accessed on 20 March 2021).
74. Node.js. Available online: https://nodejs.org/en/ (accessed on 20 March 2021).
75. Fetch-Sparql-Endpoint. Available online: https://www.npmjs.com/package/fetch-sparql-endpoint (accessed on 20 March 2021).
76. Mapbox. Static Images. Available online: https://docs.mapbox.com/api/maps/static-images (accessed on 20 March 2021).
77. React Leaflet. Available online: https://react-leaflet.js.org/ (accessed on 20 March 2021).
78. OpenLayers. Available online: https://openlayers.org/ (accessed on 20 March 2021).
79. PostGIS—Spatial and Geographic Objects for PostgreSQL. Available online: https://postgis.net/ (accessed on 20 March 2021).
80. Ngomo, A.C.N.; Auer, S.; Lehmann, J.; Zaveri, A. Introduction to Linked Data and Its Lifecycle on theWeb. In *Reasoning on the Web in the Big Data Era. Reasoning Web 2014*; Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2011; Volume 6848, pp. 250–325.
81. Brooke, J. *SUS-A Quick and Dirty Usability Scale. Usability Evaluation in Industry*; CRC Press: Boca Raton, FL, USA, 1996; ISBN 9780748404605.
82. Bangor, A.; Kortum, P.; Miller, J. Determining What Individual SUS Scores Mean: Adding an Adjective Rating Scale. *J. Usability Stud.* **2009**, *4*, 114–123.
83. Buil-Aranda, C.; Hogan, A.; Umbrich, J.; Vandenbussche, P.Y. SPARQL Web-Querying Infrastructure: Ready for Action? In *Proceedings of the Semantic Web—ISWC 2013, Sydney, NSW, Australia, 21–25 October 2013*; Alani, H., Kagal, L., Fokoue, A., Groth, P., Biemann, C., Parreira, J.X., Aroyo, L., Noy, N., Welty, C., Janowicz, K., Eds.; Springer: Berlin/Heidelberg, Germany, 2013; pp. 277–293.
84. Janowicz, K.; Hu, Y.; McKenzie, G.; Gao, S.; Regalia, B.; Mai, G.; Zhu, R.; Adams, B.; Taylor, K.L. Moon Landing or Safari? A Study of Systematic Errors and Their Causes in Geographic Linked Data. In *Proceedings of the Annual International Conference on Geographic Information Science, GIScience, Montreal, QC, Canada, 27–30 September 2016*; Miller, J.A., O'Sullivan, D., Wiegand, N., Eds.; Springer: Berlin/Heidelberg, Germany, 2016; Volume 9927, pp. 275–290.
85. OGC. Web Feature Service. Available online: https://www.ogc.org/standards/wfs (accessed on 20 March 2021).

Review

Semantic Trajectory Analytics and Recommender Systems in Cultural Spaces

Sotiris Angelis [1,*], Konstantinos Kotis [1] and Dimitris Spiliotopoulos [2]

[1] Intelligent Systems Lab, Department of Cultural Technology and Communication, University of the Aegean, University Hill, 81100 Mytilene, Greece; kotis@aegean.gr

[2] Department of Management Science and Technology, University of the Peloponnese, 22100 Tripoli, Greece; dspiliot@uop.gr

* Correspondence: sotiris@aegean.gr; Tel.: +30-697-971-0389

Abstract: Semantic trajectory analytics and personalised recommender systems that enhance user experience are modern research topics that are increasingly getting attention. Semantic trajectories can efficiently model human movement for further analysis and pattern recognition, while personalised recommender systems can adapt to constantly changing user needs and provide meaningful and optimised suggestions. This paper focuses on the investigation of open issues and challenges at the intersection of these two topics, emphasising semantic technologies and machine learning techniques. The goal of this paper is twofold: (a) to critically review related work on semantic trajectories and knowledge-based interactive recommender systems, and (b) to propose a high-level framework, by describing its requirements. The paper presents a system architecture design for the recognition of semantic trajectory patterns and for the inferencing of possible synthesis of visitor trajectories in cultural spaces, such as museums, making suggestions for new trajectories that optimise cultural experiences.

Keywords: semantic trajectories; recommender systems; big data analytics; user experience; cultural space

Citation: Angelis, S.; Kotis, K.; Spiliotopoulos, D. Semantic Trajectory Analytics and Recommender Systems in Cultural Spaces. *Big Data Cogn. Comput.* **2021**, *5*, 80. https://doi.org/10.3390/bdcc5040080

Academic Editor: Giuseppe Maria Luigi Sarnè

Received: 2 October 2021
Accepted: 13 December 2021
Published: 16 December 2021

1. Introduction

Visitors of cultural spaces (e.g., museums, archaeological sites) are usually offered a rather static and less personalised experience, e.g., a group-organised guided tour of exhibits in museum rooms. To overcome this problem, there have been numerous studies that utilise advancements in recent technologies, such as IoT and pervasive computing technologies, to monitor and analyse visitor movement and interactions within cultural spaces [1,2]. Analysed data (with high volume, velocity, and variety) gathered from sensors (streaming/dynamic data) and datastores/databases (historical/static data) are used to recognise/infer visitor preferences and personal interests to propose and eventually deliver an enhanced cultural experience. This is achieved either by providing personalised and enriched content or by suggesting personalised navigation in the cultural space. A related example is found at the Rijksmuseum Amsterdam, which offers a real-time routing system for implementing a mobile museum tour guide for personalised tours [3].

An already effective approach for discovering preferences and needs for moving users in cultural spaces is through the analysis of their trajectories (big movement data), as they contain rich explicit and implicit information and knowledge. Due to the evolution of mobile computing, wireless networking, and related technologies, such as GPS, mobile applications can monitor and share information about user position during movement, e.g., while the user is visiting a cultural space. The existing infrastructure enables applications to produce a vast amount of streaming data that include information not only about locations and places that users are visiting but also the paths/routes/trajectories the users are following, as an aggregation of connected spatial points in specific time-lapses [4–7].

Semantic Web technologies offer powerful representation tools for pervasive applications. The convergence of location-based services and Semantic Web standards allows easier interlinking and semantic annotation of trajectories, resulting in semantic trajectories. Trajectory-based operations, which involve spatiotemporal data of moving entities, are becoming increasingly important in related studies and applications, as they provide insights about human movement and the ability to extract patterns and predict future behaviours. As described in [8], a semantic trajectory-based recommender system (RS) is designed on the basis of the observation that users with similar trajectories would have similar preferences for the available objects, and outperform traditional recommendation methods that do not consider trajectory or environment information.

The motivation for this research is to explore human movement and behaviour in cultural spaces to provide optimised cultural experiences. Based on this motivation, we propose a framework that represents visitor movement as enriched semantic trajectories to extract useful information required as input to a recommender system that would provide optimum alternatives to their cultural experiences. The main requirements of the framework (described in Section 5) are summarised as follows:

a. Exploitation of raw spatiotemporal trajectory data
b. Semantic segmentation and annotation of the trajectory
c. Trajectory description using suitable ontologies
d. Semantic trajectory enrichment with Linked Open Data (LOD)
e. Semantic annotation of cultural spaces and points of interest (POI) to provide context and capability for semantic integration with user trajectories
f. Trajectory analytics for pattern recognition and classification
g. Future location prediction
h. Dynamic user profiling
i. Integration of User Knowledge Graph (UKG)
j. Integration of Cultural Space (CS) and POI Knowledge Graph (KG)
k. Integration of KG-Based recommender system (RS) for path-based and KG-based recommendations
l. Integration of context-aware RS
m. Integration of hybrid RS
n. Integration of collaborative filtering RS
o. Inference and proposal of a possible synthesis of visitor trajectories

The specific set of requirements was derived from an extensive study of the related work (pros and cons of related approaches/systems), as well as from our motivation to select the most appropriate techniques and methods related to the synthesis of both paradigms, i.e., semantic trajectories and recommender systems. The aim was to develop a novel framework that will eventually (a) enhance the functionality and efficiency of a recommender system for visiting cultural spaces, and (b) connect user trajectories with optimised cultural experiences.

The aim of this paper is two-fold: (a) to present a systematic literature review of state-of-the-art approaches related to semantic trajectories and recommender systems, with an emphasis on the cultural domain, and (b) to introduce a high-level framework for the recognition of trajectory patterns, inferencing possible syntheses of visitor trajectories in cultural spaces and the combination of trajectory analysis results with visitor dynamic profiling data. The RS is used to suggest optimal alternative trajectories in real-time to enhance the visitor experience. The framework is presented through a use case scenario of a trajectory that takes place in the city of Athens and the Acropolis Museum.

The structure of this paper is as follows: Section 2 describes the basic concepts of Semantic Trajectories, Recommender Systems, and Knowledge Graphs. Section 3 discusses the survey methodology and the decisions made for the final selection of related studies. Section 4 presents the reviewed state-of-the-art related work regarding Semantic Trajectories and Recommender Systems. Section 5 critically discusses the related work based on a set of requirements towards a framework that enhances cultural experiences. Section 6 presents the proposed system architecture design. Finally, Section 7 concludes the paper and reports on future work.

2. Preliminaries

2.1. Semantic Trajectories (STs)

A trajectory is defined as the composition of the sections of connected traces and points that express a meaningful movement in space and time by an object or entity of interest. The study of trajectories is fundamental for the comprehension of moving object/entity behaviour, as there is a plethora of useful information in the path/route the object/entity follows to navigate between start and end points. The behaviour of a trajectory is the sum of the characteristics that identify the essential details of a moving object/entity or a group of moving objects/entities. A set of such unique characteristics creates a short description of a group of trajectories which are called patterns [9,10]. Data analytics based on trajectories of moving objects/entities, such as trajectory clustering and construction, could provide advantages in the solutions of several common or more complex problems [11–13]. Trajectory analysis can be performed either using raw spatiotemporal data or semantically annotated movement data. Although data mining plays a significant role in this domain, as these algorithms are used for the extraction of trajectory patterns, most of the mining algorithms are developed for raw data implementation and, as a result, they are not effective in recognising patterns for specific domains. Moreover, the exclusive usage of spatiotemporal data provided by movement tracking sources lacks significant information about the context of the movement [4,14].

One of the main issues is the difficulty to correlate the patterns with movement behaviours to extract and expand the knowledge about them. The challenge is in the enrichment of the spatiotemporal data of the trajectories with semantic, context-based information, relevant to the moving objects and the contextual association of the trajectories with related low- or high-level events. One way to address this challenge is the use of ontologies and linked data (LD) to semantically connect the trajectory patterns and behaviours with the broader context of the movement. As stated by Dodge et al. [15], the movement behaviour depends on the general context in which it takes place, as every movement has a specific meaning in the moment and in the space/environment that it is happening. Semantic trajectories are the trajectories that have been enriched with semantic annotations and one or more complementary segmentations [10]. Annotations of segmented parts of a trajectory (episodes) could be "stop" or "move", or, in other cases, could be points or regions of interest. An example semantic trajectory of a touristic walk is depicted in Figure 1.

Knowledge discovery tasks can be performed in semantic trajectories to extract patterns based on characteristics, such as changes or stops, POIs, or specific behaviours that can be recognised in a single or a group of trajectories and used to create classes and classify or identify future trajectories [9].

2.2. Recommender Systems (RS)

RSs are software tools or AI applications that are designed to predict the user's interests and preferences based on statistics, data mining algorithms, and machine learning techniques, to suggest/recommend products, services, locations, routes, etc. RSs handle the problem of information overload that users normally encounter and affect the way users make decisions through the recommendation of suitable actions or objects of interest. An example of a Cultural Recommender System is depicted in Figure 2. The main tasks of a RS are to (a) gather and process data, (b) create models based on the available data, (c) apply the models to existing and future data, and (d) receive feedback and re-evaluate the models. Common characteristics that a RS should possess for it to be considered efficient are: accuracy, coverage, relevance, novelty, serendipity, and recommendation diversity [16]. Most recommendation strategies are defined based on three main relation types: (a) User–Item relation, (b) Item–Item relation and (c) User–User relation.

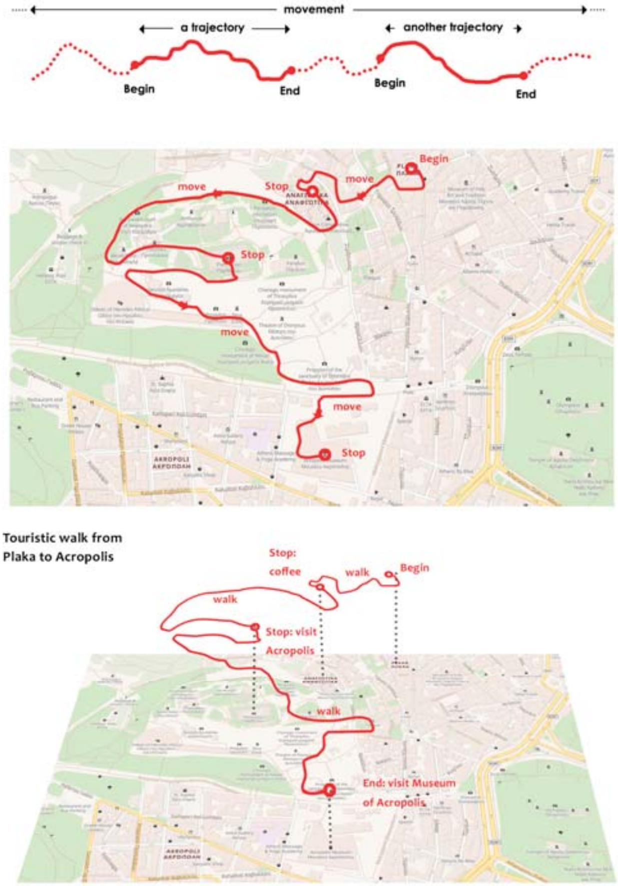


Figure 1. Example semantic trajectory depicting a tourist behaviour in Athens, starting from a simple trajectory, and resulting in a semantic trajectory.

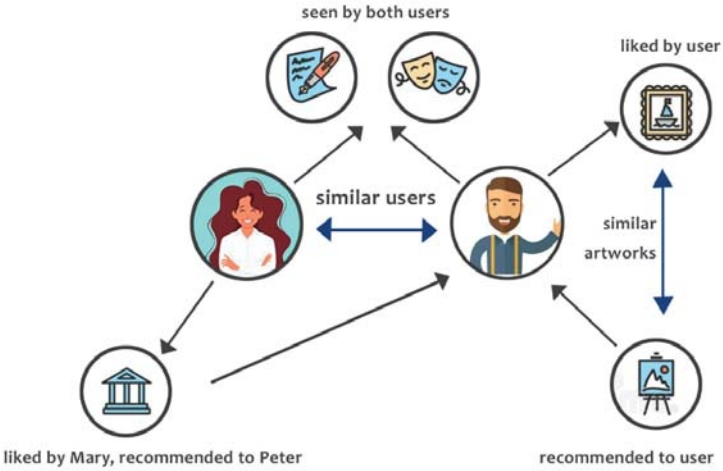


Figure 2. Example of a Cultural Recommender System.

- User–Item relation: This relation is based on the user profile and the explicitly documented preferences of the user towards a specific type of Item.
- Item–Item relation: This relation occurs based on the similarity or the complementarity of the attributes or descriptions of the Items.
- User–User relation: This relation describes the Users that possibly have similar tastes with respect to specific Items, such as mutual friends, age group, location, etc.

User preferences are calculated with the use of explicit ratings provided by user evaluations or feedback, and implicit ratings that are inferred from the users' interactions with the products or the aforementioned relations.

Depending on the available data, there are two main methods of categorisation of objects (supervised and unsupervised) based on the similarity of their features. Classification is the supervised method. It uses predefined tags and classes to categorise a group of inputs. Commonly used algorithms for classification are K-Nearest Neighbours (kNN), Decision Trees, and Naïve Bayes. In the case of Clustering, the unsupervised method, labels or categories are unknown in advance and the task is to effectively categorise given inputs and discover similarities based on certain criteria. Examples of clustering algorithms are K-Means Clustering and DBSCAN (Density-based Spatial Clustering).

One of the most common issues that RSs deal with is the rating sparsity, known also as the cold start problem, which causes inefficiency and low accuracy on recommendations. RSs face that condition when the number of available items, i.e., candidates for recommendation, is much greater than the number of users rating them. This situation occurs especially when a RS has recently started collecting ratings or has not been exploited by a lot of users [17].

Popularity-based RSs are often used to bypass the cold start problem that recommend the most selected products by other users. Demographic RSs, use information like age, gender, etc., to classify users for future recommendations. Demographic techniques are sometimes included in Hybrid RSs to increase robustness [18].

Collaborative Filtering (CF) RS use models that focus on User–User relations or Item–Item relations to infer ratings about products. The methods used in CF are equivalent to those of a classifier that creates a training model from labelled data. The basic idea of CF methods is that unspecified ratings can be imputed because the observed ratings are often highly correlated across various users and items. The main challenge in those methods is that the matrices of the user ratings are very sparse, as the users usually rate a fraction of the available products. When the user preferences are specified, the model tries to discover similarities to other users. If the similarity discovery is successful, the ratings of similar users are used to infer values to complete the rating matrices [16].

Content-based RSs are based on the idea that the user will prefer products similar to those already used and rated highly. This type of RS creates representations of products based on their features and descriptions and matches them to similar attributes of other products to suggest them to the target user. The content-based methodology calculates user profiles and specifies the interests and preferences to compute a relevance score that predicts the user's level of interest in a specific product. The product attributes for representations are usually extracted by metadata or textual descriptions, but there is an increasing interest in the advantages of Semantic Web technologies to approach content-based recommendations. As there is a plethora of open knowledge sources that provide semantic information, recent research studies shift from keyword-based to concept-based representations of products and users [19].

Apart from the Item ratings, context may be anything that might affect the desirability of particular recommendations at the time of the generation of the recommendations [19]. The context-aware RS (CARS) is a new trend in recommender systems. The CARS considers user profiles as dynamic and evaluates user preferences and interests along with other factors that may occur in the current situation of the target user, like the user's location, the user's companionship, the weather, etc. The CARS aims to provide personalised recommendations according to both user profiles and their current contextual conditions [20].

The knowledge-based RS utilises domain knowledge either provided by experts in the form of domain-specific rules and ontologies or by the usage of the knowledge available on the Web as structured LOD. KGs can be used to exploit explicit connections between user entities and product entities or infer implicit connections to create suggestions for the users. As mentioned in [21], there are several studies on ontology-based, LOD-based, path-based, and KG-based recommendation approaches that perform better than traditional recommendation approaches, especially in cases with small amounts of sample ratings and sparse rating matrices.

The hybrid RS exploits the advantages of different approaches, like the effectiveness of KG-based RSs in sparse data, or the collaborative methods when multiple user ratings are available. The hybrid RS leverages the strengths of several approaches, allowing recommendation methods to produce separate ranked lists of recommendations, and then merging their results to produce a suggestion list.

2.3. Knowledge Graphs

KGs are increasingly getting attention from academic and industry organisations as they provide several advantages compared to relational databases, regarding the representation and management of big and heterogeneous data. As defined by Hogan [22], a KG is a graph of data intended to accumulate and convey knowledge of the real world. The nodes represent entities of interest, and the edges represent relations between these entities. While there is a conceptual overlap between KGs and ontologies, because both are formed to be "an explicit specification of a conceptualisation", KGs can be considered more as "a graph of data with the intent to compose knowledge" [23,24].

As described in [24], the requirements for a graph to be considered a KG are:

a. Knowledge graph meaning is expressed as structure.
b. Knowledge graph statements are unambiguous.
c. Knowledge graphs use a limited set of relation types.

In terms of inferencing and entity representation, a KG can accumulate simple statements as edges in the data graph, but for more advanced tasks, more expressive ways are required, such as the use of ontologies and rules [22].

A KG stores and manages relations about entities. These relations could be expanded, and new relations could be created with the use of inference algorithms, e.g., rule-based reasoning, OWL/RDFS reasoning, or combinations of these approaches, that can infer knowledge and enrich the KG.

Figure 3 depicts an example KG that includes a user sub-graph describing the visitors, a POI sub-graph describing museums and exhibits in the city of Athens, and their connections. A sample set of RDF triples [25] of this KG is the following:

:Peter	rdf:type	foaf:Person.
:Peter	foaf:knows	:Mary.
:Mary	:visited	db:Parthenon.
db:Parthenon	rdf:type	:POI.
db:Parthenon	dbo:location	dbr:Athens.

Apart from KG enrichment, several techniques can be applied to a KG to provide insights and perform analysis of the encoded data. The most widely used techniques and approaches for analysis are listed below:

- Centrality: discovers the nodes with the most connections and the biggest impact in the graph.
- Community Detection: discovers sub-graphs that are more closely connected internally, compared to the rest of the graph.
- Connectivity: evaluates the quality of the connections in the graph, in terms of resilience, reachability, etc.
- Node Similarity: measures which neighbour nodes are in a specific area of the graph, based on their features and connections.
- Path Finding: discovers possible reachable paths between predefined terminal nodes.

- KG Embeddings: transforms graph representations to a low-dimensional vector space (graph embeddings), to allow ML applications to handle them efficiently.
- KG Recommendations: KGs, by design, provide the technical means to integrate various heterogeneous information sources, for instance, POIs and user preferences. Thus, feature similarity discovery algorithms can be applied to enhance recommendation techniques.

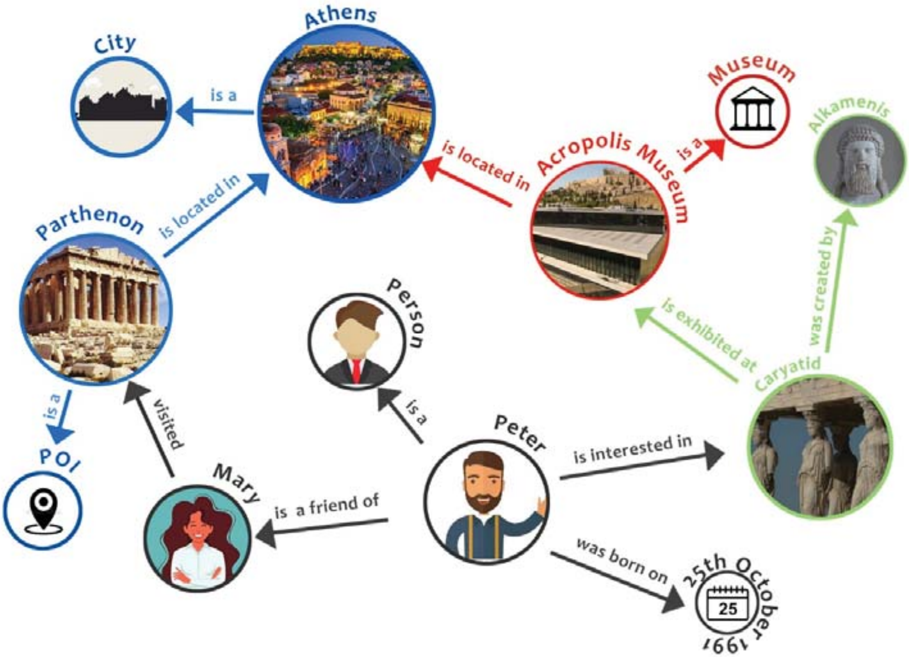


Figure 3. An example knowledge graph representing knowledge about the Acropolis and related entities, e.g., museums, visitors.

3. Survey Methodology

The research methodology followed in this paper focuses mainly on the collection of information sources related to semantic trajectories and cultural recommender systems. The research was conducted in a period of six months, examining academic articles, relevant literature, and web resources published between 2016 and 2021 (5 years).

The research for articles was conducted on academic web portals such as Scopus, Google, and Semantic Scholar, ResearchGate, ACM Digital Library, IEEEXplore, and SpringerLink.

The specific search terms used in various combinations were:

- semantic recommender systems
- trajectory-based recommender systems
- semantic trajectory-based recommender systems
- cultural recommender systems
- semantic trajectories
- trajectory annotation
- trajectory segmentation
- POI extraction and annotation
- trajectory enrichment
- human movement trajectories
- cultural semantic trajectories

The papers that were collected were mainly related to the domains of museum studies, cultural computing, and computing.

We have included articles that mostly cover human movement and trajectories that are primarily related to the domains of our field of study. Although a significant amount of the published research is on semantic trajectories and trajectory analytics in a wide range of scientific fields, we did not include works applied to other domains, such as the works related to human road safety or autonomous vehicles. However, we have taken into consideration a selection of non-human trajectory-related studies that propose noteworthy approaches regarding the semantic annotation and management of trajectories [13,26–37].

RSs are increasingly applied to a variety of use cases. Therefore, there are numerous articles about their use in several domains. We have limited the selected works to those that utilise knowledge and semantic approaches for recommendations and those that focus on cultural applications and trajectories [2,8,38–51].

4. State of the Art in STs and RSs

4.1. Semantic Trajectories (ST)

4.1.1. Annotation of Trajectories

In de Graaff et al. [26], a novel algorithm for automating the detection of visited POIs is proposed. They describe a POI as “a location where goods and services are provided, geometrically described using a point, and semantically enriched with at least an interest category”. A Polygon of Interest (POLOI) has a similar definition to a POI, except that the location is described by a polygon. The proposed method tries to identify visited POIs both in outdoor and indoor trajectories from raw smartphone GPS data but is specifically designed for urban indoor trajectory analysis. It focuses on detecting stops that take place at known and predefined POIs. The challenges overcome by the proposed algorithm are the non-detection of indoor visited POIs and the false positive detection due to lack or instability of the GPS signal. Because of the unavailability of the GPS signal for the indoor segments of a trajectory, the proposed algorithm selects points before and after the users get into a building and projects them to a polygon. The POI visit extraction algorithm considers the accuracy of the location, reductions in speed, changes in direction, and projection of signals onto polygons to extract the staypoints (the centroids of stop sequences) from a trajectory. An experiment with students in the city of Hengelo was set up to validate the proposed approach and concluded that the algorithm outperformed several existing approaches.

Another method for annotating trajectories, following a different approach, is the one presented by Nogueira et al. [13]. In this work, it is stated that the focus is on modelling mobile object trajectories in the context of the Semantic Web, and it is based on an ontological approach to extract episodes from semantic trajectories. An episode is defined as the smallest semantic unity in Semantic Trajectory Episodes (STEP) [52]. Episodes encapsulate values that a Feature of Interest (FOI) or Contextual Element (CE) may assume during the trajectory. They expand the already published STEP ontology to represent generic spatiotemporal episodes. It is claimed that, with the use of Semantic Web technologies, it is possible to integrate various data sources, data, and metadata and also support reasoning by inference, in order to enrich movement data. To validate the expanded ontology, a framework called FrameStep is proposed. The framework contains a graph–object mapping layer that enables the creation of episode instances that can be then transformed and serialised into triples. It utilises the ontology along with annotation algorithms to form semantic trajectories from raw GPS data by detecting episodes and enriching the trajectories. The framework also integrates with LD cloud and OpenStreetMap [53] to retrieve information about the context and the environment of the analysed trajectories.

In the work of Chen et al. [30], the challenge of organising raw spatial trajectories and fusing them with semantic data is examined. They present a structured and self-described way to annotate trajectory episodes with sentiments, events, or topic words. To achieve this goal, a model is implemented, where each user has multiple trajectories divided into episodes. Each episode contains semantic information like events or sentiments extracted by a combination of spatial and temporal information, with the context semantic information derived from posted texts by the users, using Natural Language Processing (NLP).

4.1.2. Semantic Trajectory Management

The work of AL-Dohuki et al. [31] focuses on an approach to interact with trajectory data through visualisations, enriched with semantic information about the trajectories. The approach was designed and evaluated for taxi trajectories. The trajectories are converted to documents through a textualisation transformation process where the GPS points are mapped to street names or POIs, and the speed is described quantitatively. After the transformation, each document is described by a meta-summary and indexed to enable queries over a text-based search engine. The system is data-structure agnostic, and the results are integrated with visualisations and interactions to promote easy understanding. The evaluation of a prototype claimed to be successful, and its ease of use is demonstrated appropriately.

Motivated by the need to exploit data from disparate and heterogeneous sources in an integrated manner to increase the predictability of moving object trajectories, Santipantakis et al. [32] propose a framework for semantic integration of big mobility data with other data sources, providing a unified representation and support analysis tasks by exploiting trajectories at various levels of analysis. One of the major challenges of this work is to enrich surveillance data providing meaningful information about moving entity trajectories, and annotating trajectories with related events, therefore creating enriched trajectories. To meet these challenges, they introduce the SPARTAN approach for providing enriched streams of mobility data, incorporating online compression, data transformation, and link discovery functionality. The domains of interest are marine and aviation. Streaming spatiotemporal data are the input to the proposed framework. It performs data cleaning and summarisation, transforms data to Resource Description Framework (RDF) [25] in compliance with a generic ontology (namely, datAcron ontology [54]) for trajectories, and performs integration with other streaming and archival data sources. The output of the framework is a stream of linked RDF data that contains enriched trajectories of moving objects. The experimental evaluation of the proposed framework demonstrates efficiency and scalability when using large, real-life datasets.

The first step to semantic annotation and enrichment of the trajectories is to perform the necessary trajectory segmentation. This can be done either with supervised or unsupervised ML techniques. In the first case, the criteria are predefined, and the segmentation can be implemented with ML techniques trained with labelled segments or by applying algorithms that segment the trajectory based on a threshold. In the latter case, the applied methods must discover similarities and homogeneity without specified criteria, based on point density or a cost function. In the work of Amilcar Soares Júnior et al. [33], a semi-supervised algorithm is proposed that implements the Minimum Description Length (MDL) principle to measure homogeneity inside segments. This algorithm uses a small set of labelled trajectory data during the trajectory segmentation task to drive the unsupervised segmentation of unlabelled trajectory data. The semi-supervised approach takes advantage of both the supervised and unsupervised techniques, so the user must annotate a small set of trajectories to help the algorithm recognise meaningful segments for the rest of the trajectories. As mentioned in the study, the main advantage of this method, compared to pure supervised ones, is that it reduces the human effort to label the number of trajectories. A few examples can be used to target the segmentation task to specific domains. The proposed algorithm is characterised as reactive, for automatically defining the values of the input parameters by analysing the results of previously provided examples. As stated in the study, for the experiments conducted using real-world datasets, the proposed algorithm outperformed the state-of-art competitors.

4.1.3. Semantic Trajectory Modelling

Semantic modelling is the step after the annotation for utilising semantic trajectories. In Vassilakis et al. [34], authors present SemMR, a semantic framework for modelling interactions between human and non-human entities, managing reusable and optimised cultural experiences towards a shared cultural experience ecosystem that may seamlessly accommodate mixed reality experiences. The proposed framework is based on the concept of cultural experience as a semantic trajectory (eX-trajectory). It is designed to utilise Mixed Reality (MR) technologies and applications for creating and managing eX-trajectory content and tools. Methods are proposed for monitoring and analysing user interactions in MR spaces for behaviour pattern extraction for optimising eX-trajectories at runtime by enriching and augmenting them with useful information from heterogeneous sources. To evaluate the framework, simulation experiments ran with artificial users as inputs. As stated, the eX-trajectories that were generated by the system were parsed and evaluated for appropriateness to each user profile and visitor path and were found to be in alignment with the user visiting style and preferences.

The related work for modelling human movement behavioural knowledge from GPS traces for categorising mobile users [35] is motivated by the challenge of using extracted knowledge from trajectory analysis of a specific place to cluster users and discover user behaviour patterns in a target place, without analysing it. That work proposes a framework that forms trajectories from raw GPS data, creates a movement behaviour model, and applies pattern mining methods to transfer knowledge regarding the human movement to geographical regions. To achieve the knowledge transfer, they cluster user behaviour in a specific region of interest and apply the results to semantically similar regions. To model the user movement, a Bayesian network is used that can encapsulate measures of the randomness of movement. The classifier of the Bayesian network provides the probability of a target user to match multiple categories that are predefined based on the features and the trajectory analysis. As an experimental evaluation, a real-life dataset of monitored GPS trajectories was used to transfer knowledge to a region with insufficient data and the results were close to those of the analysed area.

As mentioned by most of the reviewed related works, the mainstream trajectory representation methods focus mainly on the spatiotemporal information of the trajectory and not on the context or the semantic information that could be extracted from them. Gao et al. [36] present a novel representation of semantic trajectories that takes into account domain knowledge, in addition to spatial and temporal data, in order to enhance semantic trajectory retrieval. They propose a synchronisation-based clustering model to transform raw GPS points to multi-resolution Regions of Interest (ROIs). They deploy a tree-shaped hierarchical network that captures each ROI in a set of GPS trajectories. This leads to the replacement of raw trajectories with sequences of ROIs. A hierarchical embedding model transforms the ROI sequences to continuous vectors, based on geographical and semantic trajectory features in a way that the similarity measures between two trajectories can be computed by the Euclidean distance of two vectors directly. The embedding model emphasises their context of movement to extract semantic relations among target objects. The results presented after evaluation experiments showed that the proposed method had superior performance in semantic ROI/trajectory retrieval tasks compared to state-of-the-art methods and deep network embedding models.

Most of the studies in the field of trajectory representation and analytics focus on GPS data of outdoor activities and movements, while those that capture indoor POIs present them as parts or stop points of a broader trajectory. Motivated by the lack of indoor trajectory research, Kontarinis et al. [37] combine aspects of semantic outdoor trajectory models with a semantically-enabled hierarchical symbolic representation of the indoor space, in alignment with the OGC IndoorGML standard [55]. The proposed model for enriched indoor semantic trajectories utilises a standardised indoor space modelling framework that contains the semantic trajectories alongside the semantically enriched representations of indoor space. The indoor space is described as a layered multigraph.

The nodes of the multigraphs represent spatial regions, while the edges represent the topological relations between spatial regions. The model was deployed and evaluated for a dataset of spatially aggregated timestamped points of visitors of the Louvre Museum, to present its expressiveness. The study also presents the application of standard and advanced pattern mining methods to provide a formalisation for indoor trajectory mining and express the combination of semantic and spatiotemporal data.

In the work of Krisnadhi et al. [56], a pattern for modelling semantic trajectories is presented. This model adds a spatiotemporal expansion to a previously published model for the representation of trajectories. The work is motivated by the lack of published patterns for simple spatiotemporal extents. The pattern indicates that a trajectory should be modelled as a sequence of fixes that are connected by segments and that every fix should have a spatial and a temporal extent. The model follows a set of axiomatic rules. Specifically, every trajectory must have one starting and one ending fix, trajectories for the same spatiotemporal extent cannot have temporal overlap, and every spatiotemporal extent must have at least one trajectory.

4.1.4. Semantic Trajectory Analytics

While there are several works in trajectory analysis and enriching trajectories and managing them as semantic trajectories provides better insights into the target movement behaviour, limited research has been conducted on the use of Convolutional Neural Networks (CNNs) in connection with modelling human movement patterns. In Karatzoglou et al. [27], a novel CNN-based approach for representing semantic trajectories and predicting future locations is introduced. The CNN approach design was based on an NLP use case similar to the proposed, explaining that the data were also in one-dimensional format and that relevant use cases with the use of CNNs have already been studied in NLP. The CNN takes semantic trajectories as input and assigns a unique index to every semantic location. Trajectory indexes are passed to a hash table which assigns a feature vector to them. These feature vectors are task-specific representations that the system extracted in the training phase from the available data by discovering the optimal semantic location representations. The results are used as input to the core model for predicting the next semantic location. To evaluate the approach, they have worked with semantically enriched data of a single-user model and multiuser models that contained the trajectories of 100 users. The evaluation results showed that, although the proposed model is sensitive to sparse data, it outperforms other reference systems in terms of accuracy and is capable of modelling semantic trajectories and predicting future semantic locations.

In Zhang et al. [28], a system is proposed to extract the semantic trajectory patterns of data produced by users' positioning devices. As stated in the paper, the already proposed mined trajectory patterns are unable to reflect the semantic information hidden in the trajectory because a user's trajectory not only contains the physical movement track but also embodies the user's purpose for moving. Motivated by that, they propose a probabilistic generative model that annotates and clusters the pre-processed trajectory. Raw spatiotemporal data with no semantic information are divided into two parts: the moving and the stop data. They formulated a spatiotemporal threshold and clustering-based method to extract the stopover points. A probabilistic generative model was implemented to identify starting and ending points on a trajectory, connect them to POIs and, by annotating those points, discover the visiting purpose of the trajectory. Finally, the PrefixSpan algorithm [57] is applied to discover patterns over trajectories by finding suffixes for multiple prefix sequences. Evaluating experiments of the proposed method, in contrast to widely used algorithms in pattern mining like K-Means and DBSCAN, showed that it outperforms the other methods in all cases.

Several privacy issues can emerge when tracking and analysing human movement, especially when enriching the movement with context information and discovering patterns that are frequently followed. In order to reduce the risks of invasion of privacy, the work of Khoroshevsky and Lerner [29] is based on the assumption that there is no data sharing among users and that the user data are stored on the client. By performing the analysis on the user device, they prevent the exchange of sensitive location information between servers and users. An algorithm that combines spatial and semantic information for movement pattern discovery and location prediction is proposed. They retrieve semantic data from the OpenStreetMap API to specify semantic places around points that are clustered in geographic locations. To achieve point clustering in semantic locations, they propose two clustering evaluation metrics. After point clustering, the terminal and intermediate stop points form the user location history, which can be transformed into a sequence of visit locations. To cluster trajectories, they use a similarity metric for string sequences. To discover trajectory patterns, each sequence is assigned to the closest detected cluster of trajectories. Experiments that compared the proposed algorithm with previous works revealed that it provides accuracy for a reasonable number of location sequences.

The work of Liu and Wang [8] is motivated by the challenge of community detection over a set of trajectories. In contrast to most proposed methods that evaluate clustering by proximity-related metrics, they propose a framework that exploits semantic information from multiple sources, where the trajectories in a specific cluster exhibit similarity in one or more movement-related features. They defined the difference between clustering and community detection as the difference between a set of objects related purely through spatial proximity and a set of objects whose proximity or movement similarity is likely a manifestation of some underlying mutual interaction or shared relationship. The proposed framework leverages information markers underlying the raw trajectory data to detect groups based on movement information of users and semantic information of the space where the movement takes place. The framework learns the consistent graph Laplacians by constructing the multi-modal diffusion process and optimising the heat kernel coupling on each pair of similarity matrices from multiple information sources. It models the trajectory similarity based on semantic-level movement, spatiotemporal proximity, and velocity, then computes similarity measurements to determine the communities derived from the computed values. After communities are formed, they apply a collaborative filtering recommendation method based on the observation that similar users that belong to the same community have similar preferences. After experiments were conducted on selected datasets, it is stated that the community detection system and the recommendation method outperformed other clustering and recommendation algorithms.

4.2. Recommender Systems (RS)

4.2.1. Cultural RS

In Amato et al. [44], a methodology that combines recommendation with agent-based planning techniques to implement a planner of routes within cultural spaces is proposed. The problem of finding a scheduled path of visitors in a cultural space or site is handled as a reachability problem and uses multi-agent models to achieve the goal of accessing POIs within certain deadlines. First, the approach analyses user preferences to provide an accurate list of cultural items. Then, the multi-agent planning methods calculate the paths that follow sequence steps to meet the goal of visiting the suggested items. For each pair of users and items, the recommender can compute a rank that measures the expected interest of the user in an item, using a knowledge base and a ranking algorithm. The ranking algorithm integrates information about preferences and past behaviours of the target user and the user community, user feedback, and contextual information, to create the list of suggested items. The browsing system is represented as a directed labelled graph and depicts the sequence of chosen items to increase the similarity measure between them. Finally, the agents compute and recommend the path that meets the requested goals or state that it is unreachable.

Su et al. [45] propose and develop a Big Data architecture that leverages edge intelligence in addition to cloud computing for a more scalable and user-centric analysis of the cultural data. The architecture consists of a data ingestion stage, a knowledge base, a data process stage, and applications. Apart from the architecture, a user-centred recommendation edge-intelligence strategy is proposed for cultural recommendations. The RS relies on a context-aware hybrid recommendation strategy deployed on a multilayer architecture based on Big Data and edge computing technologies. It considers user preferences, location, and items' semantic features of POIs, to generate touristic routes as a sequence of POIs. The recommendation algorithm relies on features of cultural items, the user's past itineraries, and behavioural information captured by the stream processing from social networks. As a proof of concept, they developed an application for suggesting POIs, such as museums, to city visitors. Evaluation results demonstrated that the recommender system outperformed other proposals. This is explained using the Knowledge Base that stores information about objects and users, combining them with those obtained from social networks during the ranking phase.

The work of Cardoso et al. [46] presents the implementation of an application that suggests routes for cultural heritage visits. The application is designed with an adaptive user interface where routing and augmented reality are connected to acknowledge the needs of user categories, such as elders, kids, experts, or general users. The proposed application aims to suggest the optimal route of POIs between terminal nodes in a cultural environment. The suggested route is computed considering the maximum visit time and a vector of the user preferences. The design of the proposed application is based on the optimisation problems of user preference extraction, the number of visiting POIs, and time spent exploring them. The methods designed for the navigation problem were based on the ant colony optimisation algorithms and weighted function strategy. They computed the optimal paths inside a network of POIs, in near real-time, considering the user preferences and given limitations. The application is in the final stages of development and under testing with real users in a real museum environment.

The system presented in Smirnov et al. [47] deals with tasks related to the info-mobility concept: user action analysis, preference revealing, and cultural heritage recommendation based on the preferences and current situation. The system suggests possible touristic paths for visiting cultural POIs. The application architecture is based on the Smart-M3 information sharing platform and consists of a set of joined services by a smart space that provides semantic-based information exchange. The RS calculates the item ranking based on the CF method to provide the list of POIs. The ranking algorithm considers the ratings set by all the users of the system concerning the similarity to the current user, the user preferences, the current situation in the target location, and the reachability of the POIs. The user situation while using the application is modelled by ontology-based context. The application provides detailed information about cultural heritage POIs that are retrieved from internet sources in real-time. It also estimates the reaching path and presents it in an interactive map. The evaluation shows that the application is efficient in finding the cultural POIs in the user's area in a reasonable time and can be used for on-the-fly tourist support during a trip.

An interesting research challenge is how the cultural factors influence RS efficiency, whether cultural differences could be a technology barrier, or whether there are universal factors for RSs. Motivated by that challenge, Hong et al. [48] proposed the novel concept of cross-cultural contextualisation and a model to compute the cross-cultural factor affecting user preferences. They propose a contextualisation model for computing the cross-cultural factor, which influences user preferences in RSs by using Matrix Factorisation and clustering techniques. As mentioned in the study, a systematic analysis of the dataset and the experimental results suggested that individual users could be considered as country-wise groups for the model to analyse the cross-cultural factors. The users' cultural preferences are modelled to a rating matrix that contains vectors of cultural preferences on different items. Matrix factorisation is applied to interpolate the sparse matrix. Experiments using a real-world dataset, which contains ratings by visitors from different countries, showed the effectiveness of the proposed model and supported that there are cultural factors that influence user rating behaviour in recommendations.

In Loboda et al. [49], the authors demonstrate the impact of the museum RS and implement a content-based RS to generate personalised museum tours to enhance visitors' experience by providing a personalised way to engage with museum collections. The feedback received from a study conducted to evaluate the improvement of a visit by a RS focused on the provided information about the objects and the accessibility of the museum collection. The developed RS was based on the content-based filtering method regarding the feature similarities across the items of the collection. To address the cold-start problem, the users had to select preferred items from a list before the application provides a personalised tour. The application provides detailed information about the recommended items. Evaluation by real visitors reported that the RS made the visit more structured and helped the discovery of interesting objects. Another aspect of the findings of the user evaluation was that diversity in museum RSs might be favoured over accuracy.

In related works about social recommendation services for cultural heritage [50], it is stated that we can guess the affinity of an artwork choice between two users from the users' artwork-watching histories and the artwork features. Motivated by that, Hong et al. presented a novel recommender system based on a method for discovering and exploiting social affinity between users based on artwork features and user experience. The system is designed based on a use case scenario where the museum that exploits the RS is equipped with an IoT system relying on intelligent sensors and services that can identify the user behaviour during the visit. The RS follows a hybrid recommendation approach, where the suggestions are computed from the results of a social-based and context-based recommendation method. The social recommendation ranks the list of artworks based on social affinity by requests of individuals or groups. The affinity between users is estimated using an affinity graph, which is created from filtering the information of the user history and the features of the artworks. For artwork similarity measurement, the Jaccard similarity coefficient and Euclidean distance were calculated. The context-based method recommends artworks based on user interaction. The proposed approach is about to be evaluated through a developed application in a museum in Naples that offered the necessary infrastructure.

4.2.2. Semantic and Knowledge-Based Recommender Systems

Interactive RSs are modelled as a multistep decision-making process to capture the dynamic changes of user preferences. Zhou et al. [38] present a recommendation approach that utilises reinforcement learning methods and KG to provide semantic information to an Interactive RS. Reinforcement learning methods face an efficiency issue when provided with a small sample of data. To address the issue, they leverage prior knowledge of the item relations in the KG for better candidate item retrieval, enrich the representation of items and user states, and propagate user preferences among the correlated items. Interactions between the user and the system last for a defined time period. At each period, the system dynamically generates a list of items based on historical interaction data and item similarity from the KG, suggests them to the user, and receives feedback in order to update the recommendations. The introduced model consists of a graph convolution module, a state representation module, a candidate selection module, and the Q-learning network module. Evaluation experiments demonstrated that the proposed approach outperformed the state-of-the-art method.

In Sansonetti et al. [2], the authors introduce a hybrid RS empowered by social media interactions and LOD. The first step in the recommendation method is the collection of data relevant to the user by analysing the social profiles to retrieve preferences and past interactions with other users and with cultural places. The collected information is stored in a Neo4j [58] graph database and disambiguation tasks are applied to the graph through LOD tools. The proposed recommendation approach integrates both social and semantic recommender methods. The social recommendation method performs CF to suggest items that users with similar interests from the social network prefer. Semantic recommender leverages the DBpedia [59] and Europeana [60] knowledge bases for suggesting cultural places that share similar semantic features. The system also considers the contextual information of the suggested places, such as accessibility and weather conditions. The recommended POIs are formed in an itinerary with user preferences and contextual constraints. The routes are modelled as a directed graph where nodes represent the POIs, and the weighted edges represent the time required to reach the next POI in the sequence. The graph is filtered based on the location of the POIs. POIs are annotated with information from the LinkedGeoData [61] dataset using SPARQL [62] queries. Experimental results on real users showed the effectiveness of the modules of the proposed RS.

Minkov et al. [39] propose a graph-based recommender framework, to help museum visitors deal with information overload. A KG is constructed to model the museum environment into classes of entities. The nodes represent entities of users, multimedia presentations, physical positions of artworks, or semantic themes, while edges represent structured relations between entities or viewed relations between users and artworks. To infer similarity between nodes, the Personalised Page Rank (PPR) algorithm [63] with a random walk is applied to the graph to rank items based on their relevance to the target user profile. The user profile is generated by using the previous interactions of the user with other entities in the graph. So, user feedback is given on viewed multimedia presentations or POIs while not-yet-watched presentations are ranked up according to the user interactions and preferences. As reported in the study, the results of experiments conducted using data collected at the Hecht Museum showed that graph-based recommendation using the PPR measure outperformed a set of classical collaborative and content-based recommendation methods, justifying the superiority of the graph-based approach.

Qassimi and Abdelwahed [51] present a graph-based recommendation approach that utilises semantic information extracted from collaborative tagging of cultural heritage places to enhance cultural heritage visits and suggest semantically related places that are most likely to be of interest to the visitors. The recommender system is based on the emerging graphs representing the semantic relation of similar cultural heritage places and their related tags. The emerging graphs form a multilayer graph. Its nodes represent the cultural heritage places. The edges represent their relations. Descriptive metadata are extracted by exploring a folksonomy of shared resources to augment the cultural places. The augmented places are clustered based on the tags used to annotate them. The user is provided with suggestions of similar places to those previously visited, tagged, or rated. Evaluation of the system shows that it achieves better results compared to content-based approaches.

4.2.3. Trajectory-Based Recommender Systems for Cultural Spaces

Rodriguez-Hern et al. [40] introduce a trajectory and user-based collaborative filtering approach and implement a context-aware recommender system in order to provide the user with visiting routes in a museum. The system considers several contextual aspects, such as user preferences, choices of other visitors, time constraints, current location, and trajectory. To evaluate the proposed approach, they used a real dataset of the artwork of the Museum of Modern Art collection and reproduced the layout of six floors by converting map images available on the Web to graph structures by using the tool WebPlotDigitizer [64]. The DataGenCARS [65] tool was used for user rating generation, producing ratings provided by synthetic users for artworks already visited, based on the users' profiles, as well as random trajectories that were assigned to the users. The exchange of opinions among visitors relies on a central information service that allows feeding the recommendation process with data to pro-actively suggest changes in the recommended route in the museum. The recommendation approach detects visitors with similar preferences to the user and uses their ratings to estimate the potential ratings of the user for various artworks. If those ratings exceed a threshold, an artwork is considered a candidate for suggestion. The highest-rated items are ordered in a way that minimises the overall distance and the generated path that consists of their optimal sequence is suggested to the user.

DeepTrip is an end-to-end neural network method for understanding the underlying human mobility and modelling of the POI transitional distribution in human moving patterns. DeepTrip is proposed by Gao et al. [41] as an implementation of a trajectory embedding approach for a low dimensional representation of POI contextual features. A trip encoder that leverages a recurrent neural network is responsible for the route embeddings and a trip decoder for reconstructing the routes. A Generative Adversarial Network (GAN) is defined as a learning model consisting of a generator and a discriminator that compete in a two-player min-max game. The framework incorporates a GAN to enhance the generation ability of the trip decoder for POI sequence recommendations. Evaluation experiments show that DeepTrip outperforms the state-of-the-art baseline resulted from various evaluation metrics, although it expends more effort in understanding human mobility through learning implicit trajectory distributions.

Geo-tagged photos can be used to construct users' trajectories as they contain spatiotemporal information in sequential order and are useful for mining patterns of human movement. Cai et al. [42] present an itinerary RS based on semantic trajectory pattern mining from geo-tagged photos in order to provide a suggestion of POIs. Sequences of geo-tagged photos that capture POIs provide spatial and contextual information like weather conditions and travel duration. Semantic itineraries are built by annotating raw trajectories with application-dependent contextual and spatial semantics extracted from geo-tagged photos and environmental data like day, type, time, weather conditions, and place names. Then, the semantically enriched trajectories are mined by performing a variation of the Prefixspan algorithm for patterns of frequent sequences of semantic stops in the progress of the trajectories. The system consists of the offline trajectory pattern mining part and the online itinerary recommendation part. The recommendation part suggests itineraries based on a user query by filtering and ranking candidate itineraries from the semantic trajectory pattern database. The experimental results on real datasets from Flickr [66] support the effectiveness and efficiency of the proposed system over traditional approaches.

In Xu and Han [43], the authors introduce a framework for next location recommendation based on trajectory analysis and user behaviour. The framework is based on a Recurrent Neural Network (RNN) and a Similarity-based Markov Model (SMM) that combines inferred user behaviour and spatial information to suggest future locations. Raw trajectory data are converted offline to semantic sequences. Then, the users are added to clusters based on semantic similarity of their trajectories. The neural network is trained from the trajectory features of the user clusters. For the online location prediction part, the trajectory of the target user is transformed to a semantic sequence by applying the Word2vec algorithm [48] to embed spatiotemporal and semantic information into a universal space, and the user is attached to the most similar cluster. The SMM creates a state transition matrix from the correlation of the user with others in the cluster and historical trajectory data. The output of the model is an ordered sequence of candidate locations, where users can choose preferably. The proposed framework is reported to be superior to other tested models in terms of prediction performance.

5. Evaluation and Discussion

In Table 1, the evaluation of related works based on the requirements of the proposed framework is presented. The columns represent the requirements (as listed below) and the rows show the referenced studies. The order of the presented works in Table 1 is based on the structure and order of their presentation in Section 4. This rationale, along with the arrangement of the evaluation criteria (first half are related to ST, second half are related to RS), are the reasons for the effect in Table 1, i.e., quadrants I and III are comparatively empty.

Table 1. Comparative table of reviewed related works.

*	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
[26]	x	x													
[13]	x	x	x	x											
[30]	x	x													
[31]	x	x													
[32]	x	x	x	x											
[33]	x	x													
[34]	x	x	x	x	x	x		x							
[35]	x	x				x									
[36]	x					x									
[37]	x	x	x		x	x									
[56]		x	x												
[27]						x	x								
[28]	x	x				x									
[29]	x	x		x		x	x								
[8]	x	x				x									
[44]								x						x	x
[45]				i	x			x	x	x		x		x	x
[46]			ii		x			x				x		x	x
[47]								x				x		x	x
[48]								x						x	
[49]								x							x
[50]								x	x			x	x	x	
[51]					x			x		x	x				
[38]								x		x	x				
[2]				iii	x			x	x	x		x	x	x	x
[39]								x	x	x	x	x			
[40]	x							x		x		x		x	x
[41]	x				x	x	x	x							x
[42]	x	x			x	x									x
[43]	x	x						x						x	

i: enrichment of POIs with LOD, ii: ontological description of the context, iii: enrichment of POIs with LOD.
* Based on the order presented in Section 4.

The main requirements of the framework are listed as follows:

a. Exploitation of raw spatiotemporal trajectory data: raw trajectory data include useful spatial information combined with time-specific stops, speed, and direction of the visitor, needed for the initial segmentation.

b. Semantic segmentation and annotation of the trajectory: for raw trajectories to be converted to semantic trajectories and analysed as such, segmentation of the trajectory and annotation of the parts are necessary.

c. Trajectory description using suitable ontologies: Ontologies provided a structured and unified way of semantically describing instances of entities and fuse them with domain knowledge

d. Semantic trajectory enrichment with linked open data (LOD): LOD grant a plethora of continuously updated information from different data sources regarding the context of the trajectory
e. Semantic annotation of cultural spaces and points of interest (POI) to provide context and capability for semantic integration with user trajectories: semantically described POIs and spaces can make trajectory segmentation and recommendations more effective, and the interlinking of POIs and trajectories possible
f. Trajectory analytics for pattern recognition and classification: the main goal of the process is the ability to discover features and recognise patterns in trajectories to categorise them and extract meaningful information about visitor movement
g. Future location prediction: effective trajectory analysis and classification can lead to future location prediction, which is a useful input for the RSs
h. Dynamic user profiling: updating user profile based on explicit and implicit feedback of user behaviour
i. Integration of User Knowledge Graph (UKG): describes user profiles semantically and represents them as nodes in a KG
j. Integration of Cultural Space (CS) and POI Knowledge Graph (KG): represents semantically annotated and enriched POIs and CS as a KG
k. Integration of KG-Based recommender system (RS) for path-based and KG-based recommendations: performs path finding and connectivity methods for discovering possible recommendation lists in the optimal ranking order
l. Integration of context-aware RS: provides suggestions considering contextual information to enhance final recommendations
m. Integration of hybrid RS: merges multiple recommendations to achieve maximum efficiency and accuracy
n. Integration of collaborative filtering RS: leverages user similarity to produce meaningful suggestions
o. Inference and proposal of a possible synthesis of visitor trajectories: evaluation and combination of RS suggestions with respect to user preferences for generating and proposing optimised trajectories

By reviewing the state-of-the-art method in semantic trajectories and recommender systems in recent literature, it appears that a significant amount of related work is partially related to the topics of semantic trajectories and semantic cultural recommender systems. However, as our research indicates, none of these related works fully exploit the semantic information and the insights of trajectory analytics to (a) enhance the functionality and efficiency of a recommender system for cultural spaces, and (b) connect user trajectories with cultural experiences. Some of the related works [40,41,46] propose an integration of RS approaches with trajectory extracted information, but either there is no semantic annotation or analysis of the trajectories, and the focus is only on GPS points, or the recommendation approach uses only the user current position and not the full semantic representation of user trajectory to define the context. Last but not least, in most of the related works [2,40–42,44–47,49], the outcome is a suggestion of a path generated based on user interest. That path usually is the shortest path of connected POIs, without considering the user visiting style.

6. Proposed System Architecture Design

In this paper, we present the architectural design of the proposed framework (Figure 4) along with a real-life use case scenario. The framework is designed to overcome the limitations of existing related work and fully meet the requirements to provide recommendations for optimum alternatives to visitor cultural experiences based on semantic trajectory analytics, as these were discussed in Section 5.

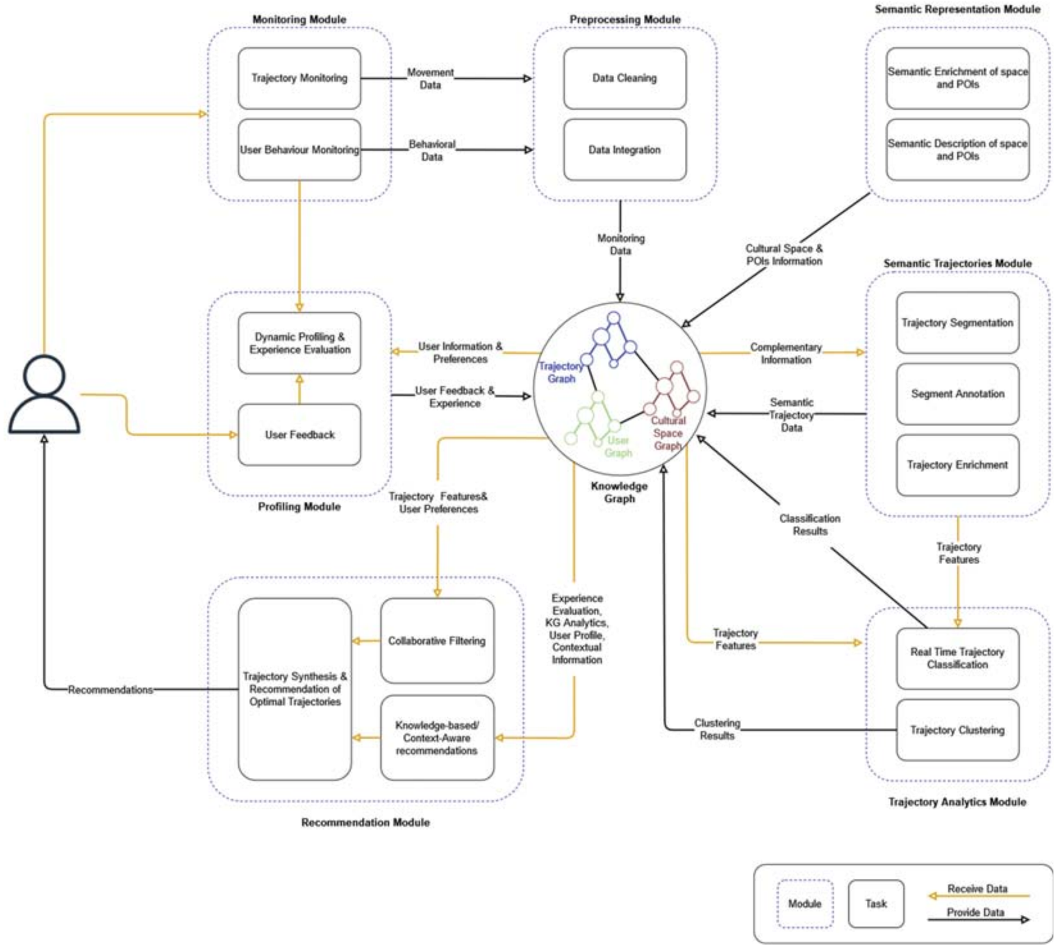

Monitoring Module
Trajectory Monitoring
User Behaviour Monitoring
Movement Data
Behavioral Data
Preprocessing Module
Data Cleaning
Data Integration
Semantic Representation Module
Semantic Enrichment of space and POIs
Semantic Description of space and POIs
Monitoring Data
Cultural Space & POIs Information
Semantic Trajectories Module
Trajectory Segmentation
Segment Annotation
Trajectory Enrichment
Dynamic Profiling & Experience Evaluation
User Feedback
Profiling Module
User Information & Preferences
User Feedback & Experience
Trajectory Graph
User Graph
Cultural Space Graph
Knowledge Graph
Complementary Information
Semantic Trajectory Data
Classification Results
Trajectory Features
Trajectory Features& User Preferences
Experience Evaluation, KG Analytics, User Profile, Contextual Information
Collaborative Filtering
Knowledge-based/ Context-Aware recommendations
Trajectory Synthesis & Recommendation of Optimal Trajectories
Recommendations
Recommendation Module
Trajectory Features
Real Time Trajectory Classification
Trajectory Clustering
Clustering Results
Trajectory Analytics Module
Module
Task
Receive Data
Provide Data

Figure 4. A system architecture design on the collaboration and information exchange between the Knowledge Graph and trajectory-oriented and recommendation-oriented components to provide optimal trajectory recommendations to the user.

The framework is designed to be trajectory-centred and to express the integration of user experience and movement in an enriched semantic trajectory. Results of analytics on that trajectory and the use of information of similar trajectories in the same cluster will provide valuable input for meaningful recommendations. Furthermore, the suggestions for the future visiting locations and the information about POIs are designed to be tailored to user preferences and continuously re-evaluated based on user choices and their physical location. As a trajectory evolves, it can be classified in a more effective manner and can be more accurately matched to the visiting style(s) of a user.

6.1. Use Case Scenario

The recording of the purpose of a cultural trip to a city is a semantic annotation at the trajectory level, while the recording of the presence of a person at a specific location, such as a visit to a temporal art exhibition, is a semantic annotation at the position level. Movement analysis of a tourist who engages in cultural related activities in Athens results in a trajectory for the whole movement in the city as a 'tourist inside Athens'. One or more distinct trajectories of daily cultural experiences, such as a tour of the Museum of

Acropolis on Friday morning, are also recorded. The distinct areas of the museum and the exhibits are semantically described. Visitor data, such as age and gender, personal preferences regarding art, music, types of museums a user visits, and the ways of touring in an exhibition space have been collected and evaluated. During the visit to the Acropolis Museum, that information is combined to produce semantically richer and more accurate trajectories, as well as to suggest routes and alternatives to improve the visitor experience. Trajectory-based personal recommendations are provided via a smartphone application on a user's phone screen, or in embedded screens near exhibits or artworks. The recommendations provide either short or detailed descriptions based on calculated user profiles and suggestions for relevant and related exhibits along with the most efficient routes to reach them.

The tourist in our scenario uses the application on a smartphone device. The application records the GPS signal to monitor the current location of the user (Trajectory Monitoring). The user has provided personal information and interests to build an initial version of a profile. As depicted in Figure 1, the user starts the walk from Plaka and, after a while, stops for a coffee break. This information is stored and used to annotate the trajectory with preferences in sightseeing and social activities (Figure 4: Trajectory Segmentation, Trajectory Annotation). As the user moves in Athens, the trajectory is taking shape and it is compared with stored trajectories in order to be grouped in a cluster with trajectories that have similar characteristics (Figure 4: Trajectory Clustering). The analysed trajectory is classified as a 'touristic walk inside Athens' (Figure 4: Trajectory Classification). The user gets a notification from the application to continue the walk to Acropolis to visit the Parthenon and receives information about this POI. The user provides feedback for the suggestion and evaluates the related information (Figure 4: explicit User Feedback). The system discovers that similar trajectories follow the path to Acropolis Museum, so it suggests the museum as the next visiting location (Figure 4: Collaborative Filtering Recommendations). The user is provided with a basic path that covers the main exhibits of the museum, based on their current profile preferences. Inside the museum, Bluetooth beacons, installed near the exhibits, provide information to the application about the proximity of the user and the time spent near them. That information, along with the interaction of the user to the provided contextual details about the targeted exhibit, are recorded as interesting information for them (Figure 4: Implicit User Feedback, Dynamic Profiling). The system is actively computing new recommendations and provides them to the user. The recommendations are based on the user preferences, the available time span, and the visiting style. The visiting style is evaluated by comparing the current trajectory with others stored in the KG. Data from the beacons and the application show that the user expresses more interest in statues rather than other exhibits. The application suggests that visitors with this preference follow a route from the Archaic Acropolis on the 1st floor to the Caryatids and the Athena Nike on the 2nd floor (Figure 4: Collaborative Filtering Recommendations). The recommendations consider avoiding crowded exhibits and rearrange the order of the suggestions (Figure 4: Context-Aware Recommendations). The calculated tour in the museum is estimated to last approximately two hours. After the first hour, the user is detected to behave differently, i.e., not devoting enough attention to the exhibits, bypassing most of the system suggestions. This behaviour is recognised as boredom or exhaustion (Figure 4: Behaviour Monitoring, Experience Evaluation). The application re-evaluates the suggested route and proposes the following (in sequence): (a) to skip the following exhibits on the 2nd floor, (b) to move to the 3rd floor, (c) to take a 20-min break at the museum café, and (d) to visit the temporal exhibition of the Acropolis Museum. The suggestion about the 3rd floor, where the artworks from the Parthenon are exhibited, was based on the interest and positive feedback received from the user about the Parthenon earlier in the tour (Figure 4: Knowledge-based/Context-Aware Recommendations).

6.2. System Architecture Modules and Tasks

In Figure 4, the proposed system architecture design for the proposed framework is depicted. The framework is divided to interconnected modules and information exchangers. The distinct modules are Monitoring, Pre-processing, Semantic Representation, Profiling, Semantic Trajectories, Trajectory Analytics, and Recommendation.

The Monitoring Module includes trajectory and user behaviour monitoring tasks. It is assumed that the cultural space is equipped with proper monitoring infrastructure (e.g., IoT devices such as Bluetooth beacons or RFIDs, and cameras) for tracking the visitor movement and behaviour. The Trajectory Monitoring task is responsible for collecting information about the physical movement of the user. This information is generated from GPS signals of portable devices such as smartphones and smartwatches, and proximity signals from Bluetooth beacons or RFID tags. The User Behaviour Monitoring task collects information about the behaviour of the user. This information is collected from the interactions of users with POIs, recorded by the developed application or by the evaluation of the user's mood (e.g., bored, interested). Movement and facial expressions can also be recorded by cameras in inner (e.g., museum) or open spaces (e.g., open museums/archaeological sites).

The Pre-processing Module receives data from the Trajectory Monitoring and Behaviour Monitoring tasks. At this module, the data is cleaned and integrated for further processing. Data integration is implemented by semantically describing the data with suitable ontologies. For instance, a data model for the system could use the datAcron Ontology [54] to describe trajectories, EDM [67] or CIDOC-CRM [68] to describe artworks, and FOAF [69] or User Profile Ontology [70] to describe users. Data is cleaned and transformed to RDF triples with tools like Karma [71] and eventually stored as a KG. This data conversion is necessary to (a) handle the heterogeneous data in a unified manner, and (b) for further enrichment and linking with related LOD.

The Semantic Trajectories Module is responsible for the transformation of raw trajectories to semantic trajectories. After data transformation, the trajectories formed by raw spatiotemporal data are segmented based on stop/move parts, velocity, or predefined POIs. The segments are semantically annotated with contextual information (day, time, place name, weather, etc.) and domain knowledge (e.g., abstract concept about the artwork). The semantic trajectories are also enriched with LOD (e.g., cultural POIs linked with a DBpedia or Europeana entity), complementary information stored in the KG, such as data provided by the Semantic Representation Module, or features visiting style evaluated by the comparison of stored trajectories in the KG.

The Semantic Representation Module is responsible for the representation of the cultural space. To provide predefined POIs and ROIs for more efficient trajectory segmentation, the cultural space and its exhibits must be described in a manner that is in alignment with the movement data and persist useful features of artworks and their peripheral area. POIs and ROIs are then semantically described with a suitable ontology that covers semantic and spatial information, converted to RDF, and stored as main entities to the cultural space KG. The entities are now linked and enriched with external information from LOD. Furthermore, they are interlinked with the POIs recognised in the trajectory segmentation part.

The Trajectory Analytics Module is responsible for the analysis and categorization of semantic trajectories. It consists of two tasks: the "online" Real-Time Trajectory Classification and the "offline" Trajectory Clustering task. The stored semantic trajectories are then analysed for pattern extraction and for the formation of semantically similar trajectory clusters. Each cluster contains trajectories with discovered visiting styles that are spatially or semantically similar. For example, one may spend the whole visit exploring all artworks and artifacts of a temporary exhibition while another is selectively visiting the essential and mainstream exhibits of a museum and spend the rest of the visit in the museum cafe. Clusters are necessary for the "online" part, where the partially constructed trajectories are assigned to the most similar one in real-time and provide to the semantic segmentation and annotation part useful information about entities' visiting style and preferences.

The Profiling Module is responsible for updating the user profile and evaluating user state. It consists of the Feedback task and the Dynamic Profiling and Experience Evaluation task. Implicit (user actions in the monitored space, visited exhibits and time spent near them) and explicit (direct ratings through forms/questionnaires) user feedback, alongside the classified trajectory, provide essential information for dynamic profiling. The Dynamic Profiling & Experience Evaluation task combines information about user preferences with feedback and behaviour monitoring data to evaluate user experience and dynamically update the user profile. The recording profiles, containing personal information and provided or inferred preferences, are stored in the User KG.

The Recommendation Module consists of the Collaborative Filtering task, the Knowledge-based/Context-Aware recommendations task, and the Trajectory synthesis and Recommendation of optimal trajectories task. CF RS methods applied to the User KG provide suggestion lists based on user similarity. A Knowledge-based/Context-aware RS provides suggestions based on semantic object similarity measurements and user preferences while considering contextual information like crowd density, weather, and minimum time needed to explore a region. This module is also responsible for providing complementary semantic and multimedia information about the visited artifacts. For instance, the user (e.g., Peter in Figure 3) is interested in a specific artwork that is described in the KG (Caryatid). The artwork is linked with information and related entities in the KG, such as the creator and the museum that it is exhibited at. The museum (Acropolis Museum) is related to the city (Athens) that it is located, while the city is related to its POIs. If other users with similar preferences (Mary) have shown interest in POIs (Pantheon) in this city, a recommended path will occur from one POI to the other, based on the feature similarity, community detection, and reachability derived from the KG.

The Trajectory synthesis and Recommendation of optimal trajectories task integrates a hybrid RS that merges the recommendation provided by the CF RS and the Knowledge-based RS. This RS leverages the merged recommendations to achieve trajectory synthesis and provide optimal personalised routes that guide visitors to preferred exhibits, according to the preferred visiting style and the available timespan. A crucial part of the described architecture is the continuous feedback and the experience evaluation that affect the RS state and create the potential to dynamically update the provided recommendations and the complementary information.

The proposed architecture is designed to meet the abovementioned requirements and cover the partial absence of ST exploitation for empowering RS towards cultural experience optimisation. Furthermore, the architecture is designed to capture the entire process of handling raw spatiotemporal data, converting them to enriched ST, clustering and classifying ST based on features, creating user and cultural space KG, and integrating different types of RS to effectively recommend enhanced trajectories.

7. Conclusions and Future Work

Cultural spaces like museums are increasingly emphasising a more personalised, optimised, and enhanced visiting experience. A way to achieve that is through efficiently and effectively understanding human movement in cultural spaces. Movement can be effectively represented and evaluated by semantic trajectory analysis, while personalisation can be realised by user/visitor profiling and by providing meaningful and interesting suggestions utilising specialised, software-like recommender systems. In this paper, we conducted a systematic review of the state-of-the-art related work, focusing on the intersection of the semantic trajectories and recommender systems, and on the advantages of the Semantic Web Technologies and KGs in both research fields. Subsequently, a framework and a system for the collection, annotation, and analysis of trajectory data, along with the integration of a hybrid knowledge-based RS, is proposed for optimising cultural experiences.

Future plans for this work involve the implementation of the proposed system by developing a set of methods and tools that meet the presented framework requirements by (a) the effective transformation of raw trajectories to semantic trajectories, (b) the performing of analytic tasks to extract meaningful information about visitors, and (c) the integration of a hybrid RS that combines results from a KG-based, a CF-based, and context-aware RS, for optimal suggestions. Future work will also include the design and evaluation of a use case scenario and experimentation with real-life users to receive useful feedback on the efficiency of the framework.

Author Contributions: Conceptualisation: S.A. and K.K.; methodology: K.K.; formal analysis: S.A.; investigation: S.A.; resources: D.S.; writing—original draft preparation: S.A.; writing—review and editing: K.K. and D.S.; visualisation: S.A.; supervision: K.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors would like to thank Alexandra Kallitsa for her valuable contribution.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ruotsalo, T.; Haav, K.; Stoyanov, A.; Roche, S.; Fani, E.; Deliai, R.; Mäkelä, E.; Kauppinen, T.; Hyvönen, E. SMARTMUSEUM: A mobile recommender system for the Web of Data. *J. Web Semant.* **2013**, *20*, 50–67. [CrossRef]
2. Sansonetti, G.; Gasparetti, F.; Micarelli, A.; Cena, F.; Gena, C. Enhancing cultural recommendations through social and linked open data. *User Model. User-Adapt. Interact.* **2019**, *29*, 121–159. [CrossRef]
3. Van Hage, W.R.; Stash, N.; Wang, Y.; Aroyo, L. Finding your way through the Rijksmuseum with an adaptive mobile museum guide. In Proceedings of the 7th Extended Semantic Web Conference, ESWC 2010, Heraklion, Greece, 30 May–3 June 2010; Volume 9088, pp. 46–59. [CrossRef]
4. Andrienko, G.; Andrienko, N.; Fuchs, G.; Raimond, A.M.O.; Symanzik, J.; Ziemlicki, C. Extracting semantics of individual places from movement data by analyzing temporal patterns of visits. In Proceedings of the First ACM SIGSPATIAL International Workshop on Computational Models of Place, Orlando, FL, USA, 5–8 November 2013; pp. 9–15. [CrossRef]
5. Zhang, D.; Lee, K.; Lee, I. Hierarchical trajectory clustering for spatio-temporal periodic pattern mining. *Expert Syst. Appl.* **2018**, *92*, 1–11. [CrossRef]
6. Ying, J.J.C.; Lu, E.H.C.; Lee, W.C.; Weng, T.C.; Tseng, V.S. Mining user similarity from semantic trajectories. In Proceedings of the 2nd ACM SIGSPATIAL International Workshop on Location Based Social Networks (LBSN-10), San Jose, CA, USA, 2 November 2010; pp. 19–26. [CrossRef]
7. Giannotti, F.; Nanni, M.; Pedreschi, D.; Pinelli, F.; Renso, C.; Rinzivillo, S.; Trasarti, R. Unveiling the complexity of human mobility by querying and mining massive trajectory data. *VLDB J.* **2011**, *20*, 695–719. [CrossRef]
8. Liu, S.; Wang, S. Trajectory Community Discovery and Recommendation by Multi-Source Diffusion Modeling. *IEEE Trans. Knowl. Data Eng.* **2017**, *29*, 898–911. [CrossRef]
9. Parent, C.; Spaccapietra, S.; Renso, C.; Andrienko, G.; Andrienko, N.; Bogorny, V.; Damiani, M.L.; Gkoulalas-Divanis, A.; Macedo, J.; Pelekis, N.; et al. Semantic trajectories modeling and analysis. *ACM Comput. Surv.* **2013**, *45*, 1–32. [CrossRef]
10. Spaccapietra, S.; Parent, C.; Damiani, M.L.; de Macedo, J.A.; Porto, F.; Vangenot, C. A conceptual view on trajectories. *Data Knowl. Eng.* **2008**, *65*, 126–146. [CrossRef]
11. Nanni, M.; Trasarti, R.; Renso, C.; Giannotti, F.; Pedreschi, D. Advanced knowledge discovery on movement data with the GeoPKDD system. In Proceedings of the 13th International Conference on Extending Database Technology, Lausanne, Switzerland, 22–26 March 2010; pp. 693–696. [CrossRef]
12. Bao, J.; Zheng, Y.; Wilkie, D.; Mokbel, M. Recommendations in location-based social networks: A survey. *Geoinformatica* **2015**, *19*, 525–565. [CrossRef]
13. Nogueira, T.P.; Braga, R.B.; de Oliveira, C.T.; Martin, H. FrameSTEP: A framework for annotating semantic trajectories based on episodes. *Expert Syst. Appl.* **2018**, *92*, 533–545. [CrossRef]
14. Maarala, A.I.; Su, X.; Riekki, J. Semantic Reasoning for Context-Aware Internet of Things Applications. *IEEE Internet Things J.* **2017**, *4*, 461–473. [CrossRef]
15. Dodge, S.; Weibel, R.; Lautenschütz, A.K. Towards a taxonomy of movement patterns. *Inf. Vis.* **2008**, *7*, 240–252. [CrossRef]
16. Kembellec, G.; Chartron, G.; Saleh, I. *Recommender Systems*; John Wiley & Sons: Hoboken, NJ, USA, 2014; ISBN 9781119054252.

17. Pavlidis, G. Recommender systems, cultural heritage applications, and the way forward. *J. Cult. Herit.* **2019**, *35*, 183–196. [CrossRef]
18. Bobadilla, J.; Ortega, F.; Hernando, A.; Gutiérrez, A. Recommender systems survey. *Knowl.-Based Syst.* **2013**, *46*, 109–132. [CrossRef]
19. Ricci, F.; Rokach, L.; Shapira, B. *Recommender Systems Handbook*; Springer: Berlin/Heidelberg, Germany, 2011; ISBN 9780387858203.
20. Barranco, M.J.; Noguera, J.M.; Castro, J.; Martínez, L. A context-aware mobile recommender system based on location and trajectory. *Adv. Intell. Syst. Comput.* **2012**, *171 AISC*, 153–162. [CrossRef]
21. Chicaiza, J.; Valdiviezo-Diaz, P. A comprehensive survey of knowledge graph-based recommender systems: Technologies, development, and contributions. *Information* **2021**, *12*, 232. [CrossRef]
22. Hogan, A.; Blomqvist, E.; Cochez, M.; D'Amato, C.; De Melo, G.; Gutierrez, C.; Kirrane, S.; Gayo, J.E.L.; Navigli, R.; Neumaier, S.; et al. Knowledge graphs. *ACM Comput. Surv.* **2021**, *54*, 1–257. [CrossRef]
23. Bonatti, P.; Decker, S.; Polleres, A.; Presutti, V. Knowledge Graphs: New Directions for Knowledge Representation on the Semantic Web (Dagstuhl Seminar 18371). *Dagstuhl Rep.* **2019**, *8*, 29–111.
24. Kejriwal, M. What Is a Knowledge Graph. In *Domain-Specific Knowledge Graph Construction*; SpringerBriefs in Computer Science; Springer: Cham, Switzerland, 2019. [CrossRef]
25. Lassila, O.; Swick, R.R. Resource Description Framework (RDF) Model and Syntax Specification. World Wide Web Consortium Recommendation. 1999. Available online: https://www.w3.org/TR/1999/REC-rdf-syntax-19990222/ (accessed on 16 November 2021).
26. De Graaff, V.; De By, R.A.; De Keulen, M. Automated semantic trajectory annotation with indoor point-of-interest visits in urban areas. In Proceedings of the 31st Annual ACM Symposium on Applied Computing, Pisa, Italy, 4–8 April 2016; pp. 552–559. [CrossRef]
27. Chen, Z.; Wang, X.; Li, H.; Wang, H. On Semantic Organization and Fusion of Trajectory Data. In Proceedings of the 2020 IEEE 44th Annual Computers, Software, and Applications Conference (COMPSAC), Madrid, Spain, 13–17 July 2020; pp. 1078–1081. [CrossRef]
28. Al-Dohuki, S.; Wu, Y.; Kamw, F.; Yang, J.; Li, X.; Zhao, Y.; Ye, X.; Chen, W.; Ma, C.; Wang, F. SemanticTraj: A New Approach to Interacting with Massive Taxi Trajectories. *IEEE Trans. Vis. Comput. Graph.* **2017**, *23*, 11–20. [CrossRef]
29. Santipantakis, G.M.; Glenis, A.; Patroumpas, K.; Vlachou, A.; Doulkeridis, C.; Vouros, G.A.; Pelekis, N.; Theodoridis, Y. SPARTAN: Semantic integration of big spatio-temporal data from streaming and archival sources. *Futur. Gener. Comput. Syst.* **2020**, *110*, 540–555. [CrossRef]
30. Soares, A.; Times, V.; Renso, C.; Matwin, S.; Cabral, L.A.F. A semi-supervised approach for the semantic segmentation of trajectories. In Proceedings of the 2018 19th IEEE International Conference on Mobile Data Management (MDM), Aalborg, Denmark, 25–28 June 2018; pp. 145–154. [CrossRef]
31. Vassilakis, C.; Kotis, K.; Spiliotopoulos, D.; Margaris, D.; Kasapakis, V.; Anagnostopoulos, C.N.; Santipantakis, G.; Vouros, G.A.; Kotsilieris, T.; Petukhova, V.; et al. A semantic mixed reality framework for shared cultural experiences ecosystems. *Big Data Cogn. Comput.* **2020**, *4*, 6. [CrossRef]
32. Ghosh, S.; Ghosh, S.K. Modeling of human movement behavioral knowledge from GPS traces for categorizing mobile users. In Proceedings of the 26th International Conference on World Wide Web Companion, Perth, Australia, 3–7 April 2017; pp. 51–58. [CrossRef]
33. Gao, C.; Zhang, Z.; Huang, C.; Yin, H.; Yang, Q.; Shao, J. Semantic trajectory representation and retrieval via hierarchical embedding. *Inf. Sci. (NY)* **2020**, *538*, 176–192. [CrossRef]
34. Kontarinis, A.; Zeitouni, K.; Marinica, C.; Vodislav, D.; Kotzinos, D. Towards a semantic indoor trajectory model: Application to museum visits. *GeoInformatica* **2021**, *25*, 311–352. [CrossRef] [PubMed]
35. Karatzoglou, A.; Schnell, N.; Beigl, M. A convolutional neural network approach for modeling semantic trajectories and predicting future locations. In Proceedings of the 27th International Conference on Artificial Neural Networks, Rhodes, Greece, 4–7 October 2018; Volume 11139, pp. 61–72. [CrossRef]
36. Zhang, W.; Wang, X.; Huang, Z. A system of mining semantic trajectory patterns from GPS data of real users. *Symmetry* **2019**, *11*, 889. [CrossRef]
37. Khoroshevsky, F.; Lerner, B. Human mobility-pattern discovery and next-place prediction from GPS data. In Proceedings of the 4th IAPR TC 9 Workshop, MPRSS 2016, Cancun, Mexico, 4 December 2016; Volume 10183, pp. 24–35. [CrossRef]
38. Amato, F.; Moscato, F.; Moscato, V.; Pascale, F.; Picariello, A. An agent-based approach for recommending cultural tours. *Pattern Recognit. Lett.* **2020**, *131*, 341–347. [CrossRef]
39. Su, X.; Sperl, G.; Moscato, V.; Picariello, A. An Edge Intelligence Empowered Recommender System Enabling Cultural Heritage Applications. *IEEE Trans. Ind. Inform.* **2019**, *15*, 4266–4275. [CrossRef]
40. Cardoso, P.J.S.; Rodrigues, J.M.F.; Pereira, J.; Nogin, S.; Lessa, J.; Ramos, C.M.Q.; Bajireanu, R.; Gomes, M.; Bica, P. Cultural heritage visits supported on visitors' preferences and mobile devices. *Univers. Access Inf. Soc.* **2020**, *19*, 499–513. [CrossRef]
41. Smirnov, A.V.; Kashevnik, A.M.; Ponomarev, A. Context-based infomobility system for cultural heritage recommendation: Tourist Assistant—TAIS. *Pers. Ubiquitous Comput.* **2017**, *21*, 297–311. [CrossRef]
42. Hong, M.; An, S.; Akerkar, R.; Camacho, D.; Jung, J.J. Cross-cultural contextualisation for recommender systems. *J. Ambient Intell. Humaniz. Comput.* **2019**, *10*, 1–12. [CrossRef]
43. Loboda, O.; Nyhan, J.; Mahony, S.; Romano, D.M.; Terras, M. Content-based Recommender Systems for Heritage: Developing a Personalised Museum Tour. In Proceedings of the DSRS-Turing 2019: 1st International 'Alan Turing' Conference on Decision Support and Recommender Systems, London, UK, 21–22 November 2019.
44. Hong, M.; Jung, J.J.; Piccialli, F.; Chianese, A. Social recommendation service for cultural heritage. *Pers. Ubiquitous Comput.* **2017**, *21*, 191–201. [CrossRef]

45. Qassimi, S.; Abdelwahed, E.H. Towards a semantic graph-based recommender system. A case study of cultural heritage. *J. Univers. Comput. Sci.* **2021**, *27*, 714–733. [CrossRef]
46. Zhou, S.; Dai, X.; Chen, H.; Zhang, W.; Ren, K.; Tang, R.; He, X.; Yu, Y. Interactive Recommender System via Knowledge Graph-enhanced Reinforcement Learning. In Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, China, 25–30 July 2020; pp. 179–188. [CrossRef]
47. Minkov, E.; Kahanov, K.; Kuflik, T. Graph-based recommendation integrating rating history and domain knowledge: Application to on-site guidance of museum visitors. *J. Assoc. Inf. Sci. Technol.* **2017**, *68*, 1911–1924. [CrossRef]
48. Rodríguez-Hernández, M.D.C.; Ilarri, S.; Hermoso, R.; Trillo-Lado, R. Towards Trajectory-Based Recommendations in Museums: Evaluation of Strategies Using Mixed Synthetic and Real Data. *Procedia Comput. Sci.* **2017**, *113*, 234–239. [CrossRef]
49. Gao, Q.; Zhou, F.; Zhang, K.; Zhang, F.; Trajcevski, G. Adversarial Human Trajectory Learning for Trip Recommendation. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *32*, 1–13. [CrossRef]
50. Cai, G.; Lee, K.; Lee, I. Itinerary recommender system with semantic trajectory pattern mining from geo-tagged photos. *Expert Syst. Appl.* **2018**, *94*, 32–40. [CrossRef]
51. Xu, M.; Han, J. Next Location Recommendation Based on Semantic-Behavior Prediction. In Proceedings of the 2020 5th International Conference on Big Data and Computing, Chengdu, China, 28–30 May 2020; pp. 65–73. [CrossRef]
52. Semantic Trajectory Episodes—Report Generated by Parrot. Available online: http://talespaiva.github.io/step/ (accessed on 16 November 2021).
53. OpenStreetMap. Available online: https://www.openstreetmap.org/#map=16/37.9704/23.7300&layers=H (accessed on 16 November 2021).
54. Santipantakis, G.M.; Vouros, G.A.; Doulkeridis, C.; Vlachou, A.; Andrienko, G.; Andrienko, N.; Fuchs, G.; Garcia, J.M.C.; Martinez, M.G. Specification of semantic trajectories supporting data transformations for analytics: The datacron ontology. In Proceedings of the 13th International Conference on Semantic Systems, Amsterdam, The Netherlands, 11–14 September 2017; pp. 17–24. [CrossRef]
55. IndoorGML OGC. Available online: http://indoorgml.net/ (accessed on 16 November 2021).
56. Krisnadhi, A.; Hitzler, P.; Janowicz, K. A spatiotemporal extent pattern based on semantic trajectories. *Adv. Ontol. Des. Patterns* **2017**, *32*, 47–53.
57. Pei, J.; Han, J.; Mortazavi-Asl, B.; Pinto, H.; Chen, Q.; Dayal, U.; Hsu, M.C. PrefixSpan: Mining sequential patterns efficiently by prefix-projected pattern growth. In Proceedings of the 17th International Conference on Data Engineering, Heidelberg, Germany, 2–6 April 2001; pp. 215–224. [CrossRef]
58. Graph Data Platform | Graph Database Management System | Neo4j. Available online: https://neo4j.com/ (accessed on 16 November 2021).
59. Home—DBpedia Association. Available online: https://www.dbpedia.org/ (accessed on 16 November 2021).
60. Discover Inspiring European Cultural Heritage | Europeana. Available online: https://www.europeana.eu/en (accessed on 16 November 2021).
61. Home—LinkedGeoData. Available online: http://linkedgeodata.org/ (accessed on 16 November 2021).
62. SPARQL 1.1 Query Language. Available online: https://www.w3.org/TR/sparql11-query/ (accessed on 16 November 2021).
63. Haveliwala, T.H. Topic-sensitive PageRank. In Proceedings of the Eleventh International Conference on World Wide Web—WWW '02, Honolulu, HI, USA, 7–11 May 2002; ACM Press: New York, NY, USA, 2002; p. 517.
64. WebPlotDigitizer—Extract Data from Plots, Images, and Maps. Available online: https://automeris.io/WebPlotDigitizer/ (accessed on 16 November 2021).
65. DataGenCARS. Available online: http://webdiis.unizar.es/~{}silarri/DataGenCARS/ (accessed on 16 November 2021).
66. Find Your Inspiration. | Flickr. Available online: https://flickr.com/ (accessed on 16 November 2021).
67. Europeana Data Model | Europeana Pro. Available online: https://pro.europeana.eu/page/edm-documentation (accessed on 16 November 2021).
68. Home | CIDOC CRM. Available online: http://www.cidoc-crm.org/ (accessed on 16 November 2021).
69. FOAF Vocabulary Specification. Available online: http://xmlns.com/foaf/spec/ (accessed on 16 November 2021).
70. User Profile Ontology. Available online: http://iot.ee.surrey.ac.uk/citypulse/ontologies/up/up.html (accessed on 16 November 2021).
71. Karma: A Data Integration Tool. Available online: https://usc-isi-i2.github.io/karma/ (accessed on 16 November 2021).

big data and
cognitive computing

MDPI

Article

Context-Aware Explainable Recommendation Based on Domain Knowledge Graph

Muzamil Hussain Syed [1,*], Tran Quoc Bao Huy [2] and Sun-Tae Chung [2,3]

[1] Department of Information and Telecommunication, Graduate School, Soongsil University, Seoul 06978, Korea
[2] Department of Intelligent Systems, Graduate School, Soongsil University, Seoul 06978, Korea; huy.tsusak@gmail.com (T.Q.B.H.); cst@ssu.ac.kr (S.-T.C.)
[3] School of Artificial Intelligence Convergence, Soongsil University, Seoul 06978, Korea
* Correspondence: engr.muzamilshah@gmail.com

Abstract: With the rapid growth of internet data, knowledge graphs (KGs) are considered as efficient form of knowledge representation that captures the semantics of web objects. In recent years, reasoning over KG for various artificial intelligence tasks have received a great deal of research interest. Providing recommendations based on users' natural language queries is an equally difficult undertaking. In this paper, we propose a novel, context-aware recommender system, based on domain KG, to respond to user-defined natural queries. The proposed recommender system consists of three stages. First, we generate incomplete triples from user queries, which are then segmented using logical conjunction ($\wedge$) and disjunction ($\vee$) operations. Then, we generate candidates by utilizing a KGE-based framework (Query2Box) for reasoning over segmented logical triples, with $\wedge$, $\vee$, and $\exists$ operators; finally, the generated candidates are re-ranked using neural collaborative filtering (NCF) model by exploiting contextual (auxiliary) information from GraphSAGE embedding. Our approach demonstrates to be simple, yet efficient, at providing explainable recommendations on user's queries, while leveraging user-item contextual information. Furthermore, our framework has shown to be capable of handling logical complex queries by transforming them into a disjunctive normal form (DNF) of simple queries. In this work, we focus on the restaurant domain as an application domain and use the Yelp dataset to evaluate the system. Experiments demonstrate that the proposed recommender system generalizes well on candidate generation from logical queries and effectively re-ranks those candidates, compared to the matrix factorization model.

Keywords: domain knowledge graph; natural language query; recommendation system

Citation: Syed, M.H.; Huy, T.Q.B.; Chung, S.-T. Context-Aware Explainable Recommendation Based on Domain Knowledge Graph. *Big Data Cogn. Comput.* **2022**, *6*, 11. https://doi.org/10.3390/bdcc6010011

Academic Editors: Konstantinos Kotis, Dimitris Spiliotopoulos and Min Chen

Received: 14 November 2021
Accepted: 13 January 2022
Published: 20 January 2022

1. Introduction

Recommender systems provide personalized recommendations for a set of products or items that may be of interest to a particular user [1]. In today's digital world, recommender systems have shown to be extremely valuable and essential tools. Numerous applications, ranging from e-commerce to social networks, are developed with enhanced algorithms to make intelligent recommendations [2–4]. Although numerous efforts have been made toward more personalized recommendations, these recommender systems remain unable to address context and other challenges, such as data sparsity and cold start problems. Recently, recommendations based on the knowledge graph (KG) have attracted considerable interest as a source of context information. This approach not only alleviates the problems mentioned above for a more accurate recommendation but also provides explanations for recommended items [5–7]. The KG is a graph representation of real-world knowledge, whose nodes represent entities and edges illustrate the semantic relation between them [8]. KGs explain the semantic and attribute relationships between concepts, in order to facilitate reasoning about them. Moreover, node and edge vectors derived from KG embedding (KGE), where semantics of KG are preserved, are used for training and inferencing machine learning (ML) models, which is useful for analytics, deeper queries, and more accurate recommendation. However, processing and making an efficient context-aware recommender system based on user-defined complex queries, such as "*Recommend best Chinese restaurants in Toronto which serve sweet Noodles or Spicy Chicken Biryani*", on large scale incomplete KGs still remains a challenging task. Finding candidate entities that satisfy queries can be approached by reasoning on KG.

Reasoning is the process of consciously applying logic to draw new conclusions from new or existing information. Logical reasoning mainly relies on first-order predicate logic (FOL), which uses propositions as the basic unit for reasoning. Simple propositions in FOL are declarative sentences that do not contain a connection and are represented in a simple form of triple (*h, r, t*). A simple query is a one-hop triple query, in the form of ($v_?$ *r, t*), where $v_?$ denotes the target entity answering the query (e.g., *which restaurant has noodles on the menu?*) or (*h, r,* $v_?$) (e.g., *which menu does the Alps restaurant serve?*). Complex queries in FOL can be decomposed into combinations of simple queries connected by logical operations (*AND* ($\wedge$), *OR* ($\vee$)) with an existential quantifier (*there exists;* $\exists$) and allows *negation* ($\neg$). Link prediction on the KG embedding space can be used to locate target entities that satisfy simple queries. Thus, complex queries consisting of any combination of simple queries with the same target entity can be answered by set operations (intersection, union) on sets of target entities that answer each simple query. However, complex queries that involve intermediate unknown entities to answer target entities, such as "*Which famous dishes Chinese restaurants serves in Toronto?*", illustrated in Figure 1, cannot be handled by link prediction.

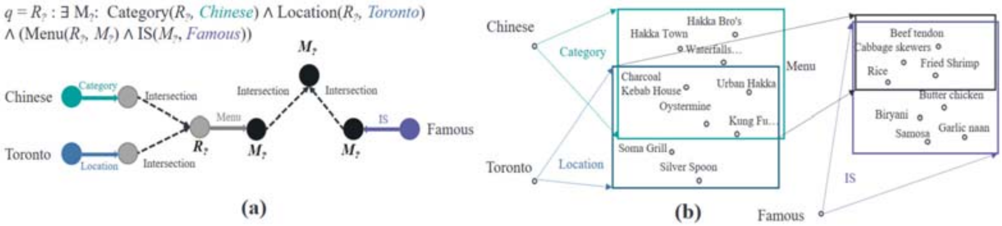


Figure 1. FOL query and its (**a**) computation graph and (**b**) vector space representation for a domain specific natural query "*Which famous dishes Chinese restaurants serves in Toronto?*".

In this work, we propose a novel, context-aware recommender system based on user-defined complex queries. Context-aware recommendation is accomplished by translating natural language queries into complex logical queries, reasoning over KGE space to identify candidate entities satisfying complex logical queries, and then re-ranking those candidates by incorporating contextual information. As for a concrete setting of the proposed system, we construct a KG for the restaurant domain and utilize the Yelp [9] review dataset to perform experiments. The proposed recommender system consists of three modules: (i) the triple generator module: generates logical triple segments from natural query; (ii) the candidate generator module: generates candidate item set with a *relevance score* (*z*), using the Query2Box (Q2B) model [10], which embeds logical queries, as well as KG entities, into a low-dimensional vector space, such that entities that satisfy the query are embedded close to the query in a hyper-rectangle boxes—candidate finder locates entities that satisfy or closely approximate existential positive first order (EPFO) logical queries, by reasoning over the KGE space and logical operations ($\wedge$, $\vee$, and $\exists$), which are processed by set operations (intersection, union and projection); and (iii) the ML-based re-ranker module: to incorporate interaction and content features for personalized recommendation—similar to the factorization machine [11], we train a neural collaborative filtering (NCF) model [12] and feed the contextual embeddings obtained from the GraphSAGE [13], which incorporates user and item node and attribute information to generate contextual embedding. The final candidate items, with relevance scores *z*, obtained in the second step, are further passed through the trained NCF model to predict a *ranking score* (r) between a pair of user and candidate items and re-rank the candidates by calculating a weighted average between ranking (r) and relevance scores (z). Since our approach considers the content and collaborative features for recommendation, and KG provides a natural explanation of the recommended candidates, the proposed approach is more user-friendly and produces accurate explainable recommendations based on defined queries.

For reasoning over EPFO logical queries, we utilize the Q2B model [10]. We prepare a domain-specific, ontology-based triple dataset, from the Neo4j graph database, for training and reasoning over the Q2B model. However, the Q2B model does not support the conversion of complex natural queries into the EPFO logical query. In this work, we provide the generation of EPFO logical queries from user-defined queries.

To the best of our knowledge, there are only a few research works that address context-aware recommendation by incorporating side features based on query expansion. However, prior works lack in handling and providing recommendation based on user-defined natural language queries.

The following is the demonstration of the major contributions of this work:

- We propose a novel framework for providing context-aware explainable recommendations based on domain KG by utilizing the existing model and embedding framework.
- We carefully design the domain ontology to capture the semantic meaning of entities and relations to make relevant recommendations, in response to user queries.
- We develop a template-based framework to transform users' natural queries into logical triple segments and extract entity concepts and their relationships using the knowledge base.

The remainder of the paper is structured as follows. In Section 2, we review the related work. In Section 3, we present the implementation and techniques of the proposed system and its components. In Section 4, we describe the data, system evaluation, and experiment results. Finally, in Section 5, we present the conclusion of the paper.

2. Related Work

Substantial work has been conducted on contextual recommendation based on KGs. Our system is built on a foundation of prior research and query-based logical reasoning.

2.1. Knowledge Graph (KG) and Knowledge Graph Embedding (KGE)

The knowledge graph (KG) is a graph representation of real-world knowledge, whose nodes represent entities and edges that illustrate the semantic relation between them [8]. A KG is based on the facts that are typically represented as "SPO" triples (subject entity, predicate (relation), and object entity) and are denoted as (h, r, t) or r (h, t). A domain KG defines the entities and relationships of a specific domain. Constructing a domain KG entails creating ontology, designing rules, extracting relations, and storing semantic data. The KG has revolutionized how information is retrieved, in the traditional sense [14]. Additionally, KGs explain the semantic and attribute relationships between concepts, in order to facilitate reasoning about them. One of the key benefits of a KG is that it can easily integrate the newly discovered facts from the information retrieval process, which captures more detailed facts and attributes on an ongoing basis and significantly resolves the KGs incompleteness problem. Moreover, structured data, in the form of KG triples, reveals numerous interesting patterns that are useful for analytics, deeper queries, and more accurate recommendation. KG is effectively used for a variety of crucial AI tasks, including semantic search [15], intelligent question answering [16], and recommendation systems [5,6].

Formally, a KG is defined as follows: $G = \{(sub, pred, obj)\} \subseteq E \times R \times E$ is a set of $(sub, pred, obj)$ triples, each including a subject $sub \in E$, predicate $pred \in R$, and object $obj \in E$. E and R are the sets of all entities and relation types of G [17]. KGE models embed entities E and relations R into a low-dimensional real or complex vector space, while preserving the structure of the KG and its underlying semantic information [18].

Such KGE models have proven to be extremely useful for a range of prediction and graph analysis tasks [11]. KGE models are classified into three categories: translational distance-, semantic matching-, and neural network-based models. Translational distance models or additive models TransE [19], TransH [20], and TransR [21], use distance-based scoring functions to calculate the similarity between the different entities and relations, thus build the embedding accordingly [22]. On the other hand, semantic matching models, or multiplicative models RESCAL [23], DistMult [24], and ComplEx [25], employ a similarity-based scoring function. In translational-based models, given a triple (h, r, t), the relation r translates the head entity h to the tail entity t. These models define a scoring function to measure the correctness of the triple in the embedding space. However, these models are reported to have low expressive power and do not capture semantic information [18]. The multiplicative models outperform the additive models by capturing more semantic information [18]. The third category includes models built on graph neural networks (GNNs), such as ConvE [26], ConvKB [27], and GraphSAGE [13]. These models consider the type of entity or relation, temporal information, path information, and substructure information [18]. Among these models, GraphSAGE is a framework for learning inductive representations on large graphs [13]. Furthermore, it achieves low-dimensional vector representations of nodes, while utilizing their attribute information, which makes GraphSAGE more appropriate for generating contextual embedding for recommendation and prediction tasks than any other KGE models. Thus, we adopt this model for obtaining user and item latent vector for training and inferencing our NCF model.

2.2. Recommendation

Two main techniques, collaborative filtering (CF) and content-based filtering (CBF) are traditionally used for recommendation [28]. The CF approach recommends items for a user by predicting the ranking score $\hat{r}(u,\ i)$ of item (i), based on users' (u) profiles, common preferences, and historical interactions. A number of research efforts have been devoted to the CF approach for implementing efficient recommender systems that utilize user and item side information, by transforming them into feature vectors to predict the rating score. Matrix factorization (MF) [29] is a well-known collaborative filtering technique that projects users and items into a shared latent space using a latent feature vector. Thereafter, the interaction between a user and item is modelled as the inner product of their latent vectors. However, CF techniques usually suffer from data-sparse and cold-start problems. Numerous studies have been conducted to improve MF [12,29–33] and addressing the cold-start problem by incorporating content (side) features to represent the user and item in latent space. Therefore, exploiting context information has great importance in resolving such problems and enhancing recommender systems. The conventional recommendation methods (CRMs) recommend items for a given user and utilize context information, such as *When?* (day, night, weekday, weekend, etc.), *What conditions?* (whether, climate, region, etc.), and with *Whom?* (family, lover, or friend). Given this information, the CRMs learn more contextual prediction rating for recommendation between user and item. However, these methods are incapable of adequately capturing the context information and are lacking in handling and providing context-aware explicable recommendations based on user-defined natural queries. Our system is capable of capturing context information from users' queries and recommending by reasoning over KG, as well as employing contextual information, similar to that found in CRMs, to add more context and re-rank the recommended candidates.

Recent studies have started to consider KGs as a source of contextual information, since they can provide valuable context information for recommendation. For example, entities, and their associated attributes, can be incorporated and mapped into the KG, in order to comprehend their mutual relationships [34]. Additionally, the heterogeneous relations in a KG help to improve the accuracy of recommender systems and increase the diversity of recommended items. Moreover, KGs facilitate the interpretation of recommender systems. In general, the majority of the known techniques for developing KG-based recommender systems can be classified into two categories: path- and knowledge graph embedding (KGE)-based approaches. Path-based methods build a user-item graph and utilize it to discover path-level similarity for items, either by predefining meta-paths or mining connective patterns automatically. However, the disadvantage of these methods is that they require domain knowledge to specify the types and number of meta-paths [35]. On the other hand, KGE-based methods expand the representation of entities and relations by embedding them in a continuous vector space. Hence, the representation of entities, and their associated relations, can be obtained from KGE space. These methods typically require a two-module approach for recommendation systems (RS) [18,22,35–37]: a KGE module for obtaining the user and item latent vectors and recommendation module for inferring recommendation (usually by computing candidate ranking score from user and item latent vectors). The advantages of these methods lie in the simplicity and scalability; however, since the two modules are loosely coupled, the approach might not be suitable for recommendation tasks. There are a few research works that deals with recommendations based on query context. Sitar-Tăut et al. [38] suggested a knowledge-driven product recommendation, using a digital nudging mechanism, to tackle the cold-start problem. They utilized a KG to integrate managerial preferences, along with product attributes that enable semantic reasoning using SPARQL queries. Zhu et al. [39] addressed the context of interactive recommendation in e-commerce by utilizing user interaction history with the e-commerce site. Bhattacharya et al. [40] utilized queries as context and found candidate items by collecting similar items while the user was searching for a target item. For candidate items, [40] calculated recommendation rank scores via a gradient boosted decision tree (GBDT). Both [39] and [40] dealt with query context by user interaction and query expansion, which is different from the context defined by a user's queries in natural language, as proposed in this paper.

2.3. Semantic Search and Reasoning over KG

FOL queries can be expressed as directed acyclic graphs (DAGs) and reasoned over KG to obtain a set of candidates. A conjunctive query performs a conjunction operation ($\wedge$) between two one-hop queries, such as ($v_?$, r_1, t) $\wedge$ (h, r_2, $v_?$). The path query, on the other hand, needs to traverse the path in a KG, such as "$\exists v$: ($v_?$, r_1, v) $\wedge$ (v, r_2, t)" or "($\exists v$: (v, r_1, $v_?$) $\wedge$ (v, r_2, t)", where $v_?$ denotes the target entity, v denotes an intermediate unknown entity with a known type, and $\exists$ denotes the existential quantifier. One concrete example of a path query is, "*Where did Turing Award winners graduate?*" ($\exists v$: (v, *Graduate*, $v_?$) $\wedge$ (v, *Win*, *Turing Award*). EPFL query allows disjunction ($\vee$) of queries, in addition to conjunction ($\wedge$) with existential quantifier $\exists$.

With the evolution of KGs, query-based methods for recommendations [10,16,19,41] have become a more intriguing topic of research. These approaches decouple entities and concepts from the query and create recommended candidates from KGs. We build our recommender framework on the concept of semantic search and reasoning over KG, since this enables the generation of explainable and context-aware query-based recommended candidates. Pirro [42] developed RECAP, based on a similar concept; the tool utilizes RDF and SPARQL for determining explainable knowledge for a given pair of entities in KG. Zhu et al. [15], Feddoul et al. [43], and Yan et al. [44] proposed searching techniques using query processing and expansion over KG. For an incomplete knowledge base (KB), some real-world queries fail to answer. The purpose of query embedding (QE) on KGs is to represent KB queries in an embedding space. A promising approach to this problem is to embed KG entities, as well as the query, into a vector space, such that entities that answer the query are embedded close to the query [34]. GQE (graph query embedding) [45] embeds graph nodes in a low-dimensional space and represents logical operators ($\wedge$ with$\exists$), modeled as learned geometric operations (e.g., translation and rotation). Ren et al. [10] further extended the concept and proposed Q2B to represent logical queries as hyper-rectangles, instead of points in a KGE vector space. Geometrical operations on those rectangles allowed for answering queries with disjunction ($\vee$), re-written in the disjunctive normal form (DNF). This technique of modeling the answering entities of logical queries resolves the candidate generation problem for each incomplete triple and finds candidate entities of a given query, without traversing the KG. Additionally, the Q2B model [10] is capable of handling EPFO logical queries. Changing the query representation form to beta distributions enabled Beta-E [41], which can further tackle queries with negation (¬) and, hence, is able to handle FOL queries. However, since our recommendation framework is constrained to a certain template for parsing natural language queries into logical triple segments, and since the domain ontology graph lacks deeper linkages, it is not possible to train and answer very complex and arbitrary queries. Therefore, we use the Q2B model [10] to train the KG triples with limited query structures, as described in the original paper.

2.4. Translating Natural Language Query to Triple

In order to process users' queries expressed in natural language, we need to convert them into triples that reflect the semantic structures of domain KGs, based on defined ontology. PAROT [46] provides a dependency-based framework for converting user's queries into user and ontology triples to construct SPARQL queries. The framework processes compound sentences by employing negation, scalar adjectives, and numbered lists.

3. Methods

3.1. Ontology Design

Ontologies represent the backbone of the formal semantics of a knowledge graph. They can be seen as the schema of the KG. A well-defined ontology ensures a shared understanding of KG entities, attributes, and their relationships. Moreover, it enables deeper queries and reasoning over KG for efficient recommendations. Figure 2 shows the ontology of our restaurant domain KG. We apply natural language processing (NLP) techniques to extract information and produce KG triples. We utilized the Neo4j database for KG storage and visualization. The description of entities, relations, and their attributes is given in Table 1. The 'Time', 'Weather', and 'Date' nodes in the ontology do not exist in domain KGs. However, by extending multi-source data acquisition and implementing more robust rules, the KG can be extended to obtain additional knowledge.

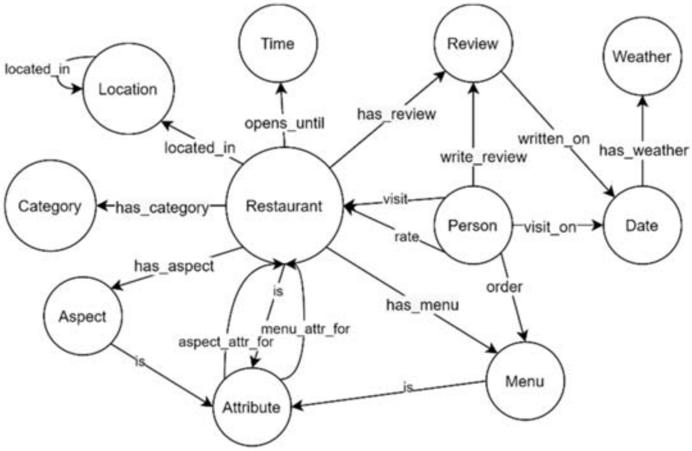


Figure 2. Conceptual graph schema (ontology) for restaurant domain knowledge graph.

Table 1. Description of domain KG entities and relations.

Entity	No. of Nodes	Relationship Types		Attributes
		Incoming	Outgoing	
Aspect	15	[HAS_ASPECT]	[IS]	aspect_id, name
Attr	241	[IS]	[ASPECT_ATTR_FOR, MENU_ATTR_FOR]	attr_id, name
Category	56	[HAS_CATEGORY]	[]	category_id, name
City	31	[LOCATED_IN]	[LOCATED_IN]	city_id, name
Country	2	[LOCATED_IN]	[]	country_id, name
Menu	624	[HAS_MENU]	[IS]	menu_id, name
Restaurant	102	[ASPECT_ATTR_FOR, MENU_ATTR_FOR, VISIT, RATE]	[LOCATED_IN, HAS_CATEGORY, IS, HAS_ASPECT, HAS_MENU, HAS_REVIEW]	rest_id, name, address, postal_code, rating
Review	3452	[HAS_REVIEW, WRITE_REVIEW]	[]	review_id, text
User	3227	[HAS_FRIEND]	[VISIT, ORDER, RATE, WRITE_REVIEW]	user_id, name, gender, age, review_count, avg_star, fans

3.2. *Proposed KG-Based Context-Aware Recommendation*

Our proposed recommender system, illustrated in Figure 3, is composed of the following three components: (i) triple generator module: generates template-based logical triple segments from a natural query; (ii) candidate generator module: comprises of the Q2B model; and (iii) machine learning-based re-ranker module: comprises of the NCF model. The NCF model is trained to learn user-item interactions by embedding latent vectors obtained from GraphSAGE framework. As a result, the model is capable of re-ranking the candidates generated in the second step of the candidate generation process by incorporating contextual features.

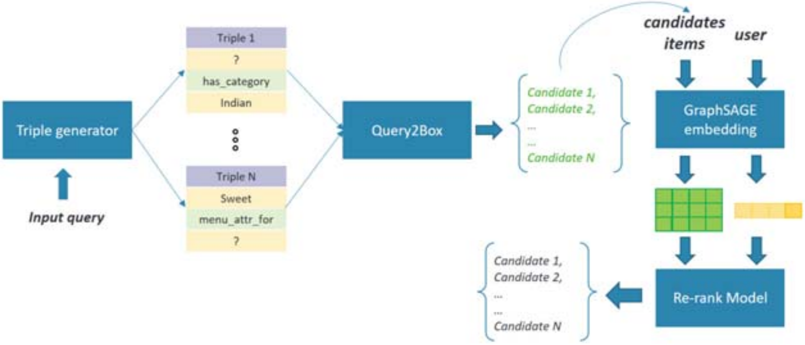


Figure 3. Proposed recommendation system.

3.2.1. Triple Generation from Natural Language Query

The triple generator module, illustrated in Figure 4, is a template-based framework that employs a simple, lexical-based technique to convert natural language query into logical triple segments. It first identifies the target word from the given user query. For our restaurant domain, we target restaurant and menus for recommendations (i.e., *recommend best* ***restaurant*** *in . . . , most* ***popular dishes*** *of Silver Spoon restaurant*, etc.). It then recognizes the entity concepts, i.e., ‘*Category*’, ‘*City*’, and ‘*Menu*’, etc., from user query, maps them to their associated relations, using the knowledge base (KB), and generates a set of triples. The goal is to convert natural language queries to logical triple segments that can be mapped to Q2B query structures and are created within the scope of the defined rules and KB entity concepts and relationships. Therefore, the framework is restricted to a specific template and can only process any arbitrary FOL queries that satisfy the template-specific rules.

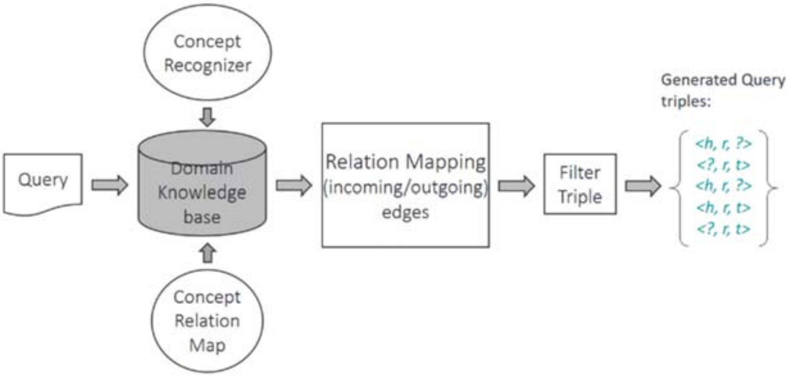


Figure 4. Triple generator module.

Table 2 illustrates the results of triple extraction, filtration, and logical query conversion for a given user query: *“Recommend best Chinese restaurant in Toronto which serves sweet Noodles or spicy Chicken Biryani”*. The extracted entity concept is mapped to its associated incoming and outgoing relations, in order to generate triples with logical conjunction ($\wedge$) and disjunction ($\vee$) (AND, OR) segments. Additionally, the module eliminates extraneous triples (in red) and filters the created triples to establish logical groupings of triple segments for reasoning and candidate retrieval. The strategy for filtration is described below:

- We eliminate all triples that do not have a direct relation to the target word or triples that do not describes the attribute of the entity concepts given in the user queries. The triple, for instance, (‘Toronto’, ‘LOCATED_IN’, ?), which represents the LOCATED_IN (‘City’, ‘Country’) triple, neither has a direct relationship to the target word nor defines the attribute of any entity concepts given in the user query. Therefore, this triple gets eliminated. On the other hand, the triple (‘?’ LOCATED_IN, ‘Toronto’) represents the LOCATED_IN (‘Restaurant’, ‘City’) triple, which has a direct relation with the restaurant concept. Similarly, the triple (?, ‘ORDER’, ‘Noodles’), which represents ORDER (‘User’, ‘Menu’), gets eliminated.
- The triples that do not belong to the similar concept are eliminated. For example, the ‘Attribute’ entity has two outgoing relationships, i.e., ‘MENU_ATTR_FOR’ and ‘ASPECT_ATTR_FOR’. Therefore, the attribute ‘Spicy’ generates two triples: (‘Spicy’, ‘MENU_ATTR_FOR’, ‘?’) and (‘Spicy’, ‘ASPECT_ATTR_FOR’, ‘?’). However, the former does not fall under the ‘Menu’ concept and, hence, the triple is eliminated.
- The two triples that result in a complete triple fact are considered a true fact and removed from the incomplete triple segments, i.e., [(‘Noodles’, ‘IS’, ‘?’), (‘?’, ‘IS’, ‘Sweet’)] or [(‘Chicken Biryani’, ‘IS’, ‘?’), (‘?’, ‘IS’, ‘Spicy’)] generates (‘Noodles’, ‘IS’, ‘Sweet’) or (‘Chicken Biryani’, ‘IS’, ‘Spicy’).

Furthermore, we transform query segments to a logical query format for querying the Q2B model for candidate retrieval.

- The logical operations ($\wedge$, $\vee$) are defined between triple segments.

Table 2. Results of user query “*Recommend best Chinese restaurant in Toronto which serves sweet Noodles or spicy Chicken Biryani*” conversion to logical query.

Extracted Triples	[{‘triple_segment’: [(‘?’, ‘HAS_CATEGORY’, ‘Chinese’)], ‘concept’: ‘Category’, ‘op’: None}, {‘triple_segment ‘: [(‘?’, ‘LOCATED_IN’, ‘Toronto’), (‘Toronto’, ‘LOCATED_IN’, ‘?’)], ‘concept’: ‘City’, ‘op’: None}, {‘triple_segment ‘: [(‘?’, ‘ORDER’, ‘Noodles’), (‘?’, ‘HAS_MENU’, ‘Noodles’), (‘Noodles’, ‘IS’, ‘?’), (‘?’, ‘IS’, ‘Sweet’), (‘Sweet’, ‘MENU_ATTR_FOR’, ‘?’), (‘Sweet’, ‘ASPECT_ATTR_FOR’, ‘?’)], ‘concept’: ‘Menu’, ‘op’: None}, {‘triple_segment ‘: [(‘?’, ‘ORDER’, “Chicken Biryani ‘), (‘?’, ‘HAS_MENU’, ‘Chicken Biryani’), (‘Chicken Biryani’, ‘IS’, ‘?’), (‘?’, ‘IS’, ‘Spicy’), (‘Spicy’, ‘MENU_ATTR_FOR’, ‘?’), (‘Spicy’, ‘ASPECT_ATTR_FOR’, ‘?’)], ‘concept’: ‘Menu’, ‘op’: ‘AND’}]
Filtered Triples	[{‘triple_segment ‘: [(‘?’, ‘HAS_CATEGORY’, ‘Chinese’)], ‘concept’: ‘Category’, ‘op’: None}, {‘triple_segment ‘: [(‘?’, ‘LOCATED_IN’, ‘Toronto’)], ‘concept’: ‘City’, ‘op’: None}, {‘triple_segment ‘: [(‘?’, ‘HAS_MENU’, ‘Noodles’), (‘Sweet’, ‘MENU_ATTR_FOR’, ‘?’)], ‘concept’: ‘Menu’, ‘op’: None}, {‘triple_segment ‘: [(‘?’, ‘HAS_MENU’, ‘Chicken Biryani’), (‘Spicy’, ‘MENU_ATTR_FOR’, ‘?’)], ‘concept’: ‘Menu’, ‘op’: ‘AND’}]
Logical Query	$q = R_?: \exists R: Category(R_?, Chinese) \wedge Location(R_?, Toronto) \wedge (((Menu(R_?, Noodles) \wedge MenuAttrFor(Sweet, R_?)) \vee (Menu(R_?, Butter\ Chicken) \wedge MenuAttrFor(Spicy, R_?)))$

The resultant logical query shows the converted output of the framework from the user query. To simplify logical query triples, the relationship aliases are converted to their simple form (i.e., HAS_CATEGORY to Category, LOCATE_IN to Location, HAS_MENU to Menu, MENU_ATTR_FOR to MenuAttrFor, etc.).

3.2.2. Query2Box (Q2B) Model

In KG, the entity that answers a query (e.g., $\{v_?; (v_?,\ r_1,\ t) = True\}$) is typically a set, not a point. If we take a query to be equivalent to a set of answer entities, then logical operations on queries are equivalent to operations on set of answer entities. A set of answer entities is embedded in a hyper-rectangle (box), formed by vectors corresponding to the entities in the KGE space. The Q2B [10] model is an embedding-based framework for reasoning over KGs that is capable of handling arbitrary existential positive first-order (EPFO) logical queries (i.e., queries having any set of ∧, ∨, and ∃) in a scalable manner. It embeds queries as hyper-rectangles (boxes) in a KGE space and logical operations, such as existential quantifier (∃), conjunctive (∧), and disjunctive operations (∨), as box operations over queries (*projection*, *intersection*, and *union*).

The Q2B operations are illustrated in Figure 5, which is accomplished by defining a box with its center and offset $b = (Cen(b), Off(b)) \in R^{2d}$ (d is the dimension of KGE space) as:

$$Box_p \equiv \left\{ v \in R^d\ : Cen(p) - Off(p) \le v \le Cen(p) + Off(p) \right\}$$

where $\le$ is an element-wise inequality, $Cen(p) \in R^d$ is the center of the box, and $Off(p) \in R^d_{\ge 0}$ is the positive offset of the box, representing the size of the box. Then, it describes the box projection and intersection operations, as well as the entity-to-box distance $dist_{box}$ (box distance).

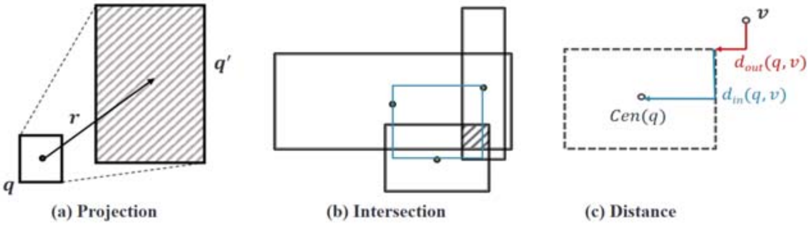


Figure 5. The geometric intuition of the projection and intersection operations in Q2B.

Q2B represents each entity $v \in V$ as a single point (zero-volume box): $v = (Cen(v), 0)$ and relation (r) as an embedding in R^{2d} and performs relation projection and box intersection, i.e., $\mathcal{P}$: Box × Relation → Box and $\mathcal{J}$: Box × ... × Box → Box in the embedding space, respectively.

Each relation (r) takes a box and produces a new box. The projection operation ($\mathcal{P}$) (Figure 5a) takes the current box as input and uses the relation embedding to project and expand the box. The geometric intersection operation ($\mathcal{J}$) (Figure 5b) takes the multiple boxes $\{\mathbf{p}_1, \ldots, \mathbf{p}_n\}$ as input $\mathbf{p}_{inter} = (Cen(\mathbf{p}_{inter}), Off(\mathbf{p}_{inter}))$ and produces the intersection box. The center of the new box should be close to the centers of the input boxes, accomplished by shrinking the offset of the input boxes. The center of the new box (in blue Figure 5b) is a weighted sum of the input box centers, given as follows:

$$Cen(\mathbf{p}_{inter}) = \sum_i w_i \odot Cen(\mathbf{p_i}), \therefore w_i = \frac{\exp(MLP(\mathbf{p_i}))}{\Sigma_j \exp\left(MLP\left(\mathbf{p_j}\right)\right)}$$

$$Off(\mathbf{p}_{inter}) = Min(\{Off(\mathbf{p}_1), \ldots, Off(\mathbf{p_n})\}) \odot \sigma(DeepSets(\{\mathbf{p}_1, \ldots, \mathbf{p_n}\}))$$

where $\odot$ is the dimension-wise product, $w_i \in R^d$ is calculated by a multi-layer perceptron (with trainable weights), and w_i represents a self-attention score for the center of each input $Cen(\mathbf{p_i})$. It takes the minimum of the offset of the input box and makes the model more expressive by introducing a $DeepSets(.)$ function to extract the representation of the input boxes with a sigmoid function to guarantee shrinking.

The entity-to-distance (Figure 5c) function defines the scoring function, in terms of distance. Given a query box (q) and entity embedding (box) ($\mathbf{v}$), the distance is defined as:

$$dist_{box}(\mathbf{v}, q) \equiv dist_{out}(\mathbf{v}, q) + \alpha \cdot dist_{in}(\mathbf{v}, q) \ \therefore 0 < \alpha < 1$$

If the point is enclosed in the box, the distance should be downweighted. Q2B handles conjunctive queries in a natural way by taking projection and intersection. However, allowing disjunction (union) over arbitrary queries requires high-dimensional embeddings, as it cannot be embedded in a low-dimensional vector space. To handle EPFL queries that contain disjunction, it makes use of a well-known fact—any FOL query may be translated into its corresponding disjunctive normal form (DNF), which has a form of disjunction of conjunctive queries. Thus, in order to answer to an EPFO queries, Q2B first applies box projection and intersection to calculate the intersected box for each conjunctive and compute the final answer entities by utilizing the aggregated distance function to a set of intersected boxes. The aggregated distance function between an EPFO query (q) and an entity ($\mathbf{v}$) utilizing the box distance $dist_{box}$ and the fact that a complex query (q) in FOL is logically equivalent to $q^{(1)} \vee \cdots \vee q^{(N)}$, as follows:

$$dist_{agg}(\mathbf{v}, q) \equiv Min\left(\left\{dist_{box}\left(\mathbf{v}, q^{(1)}\right), \cdots, dist_{box}\left(v, q^{(N)}\right)\right\}\right)$$

As long as **v** is the answer to one conjunctive query (q^i), it should also be the answer to q, and as long as **v** is the close to one conjunctive query (q^i), it should also be close to q in the embedding space.

For the purpose of learning entity embeddings, as well as geometric projection and intersection operators, Q2B optimizes a negative sampling loss, in order to successfully optimize the distance-based model. During optimization process, Q2B randomly samples a query (q) from the training graph (G_{train}), answer ($v \in [|q|]_{train}$), and a negative sample ($v' \notin [|q|]_{train}$). The negative sample entity (v') is the entity of same type as v but not the answer entity. After sampling, the query is embedded into the vector space and calculates the score $dist_{box}(v;q)$ and $dist_{box}(v'_i;q)$ and optimizes the loss (L) to maximize $dist_{box}(v;q)$, while minimizing the $dist_{box}(v'_i;q)$;

$$L = -log\sigma(\gamma - dist_{box}(v;q)) - \sum_{i=1}^{k} \frac{1}{k} log\sigma(dist_{box}(v'_i;q) - \gamma)$$

where σ is the sigmoid function, γ represents a fixed scalar margin, v is a positive entity (answer to the query q), v'_i is the i-th negative entity (non-answer to the query q), and k is the number of negative entities.

3.2.3. ML-Based NCF Re-Rank Model

Our ML-based collaborative re-ranking model, illustrated in Figure 6, provides a ranking score function $\hat{r}\ (u,\ i)$ for recommendation between the user (u) and candidate items (i) obtained from the Q2B model. The ranking score (r) is used to re-rank the obtained candidate items, based on user and item interaction and side (auxiliary) information.

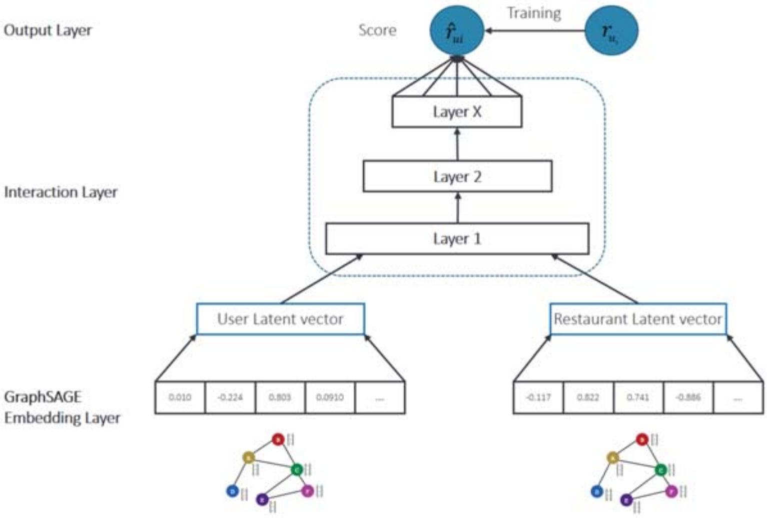


Figure 6. NCF-based re-rank model.

The NCF model leverages both the bipartite interaction and side information of users and items, as shown in Figure 7. We utilized the GraphSAGE framework to obtain the contextualized latent vectors from the domain KG. GraphSAGE generates node embeddings, while incorporating side features. The auxiliary features of user and restaurant nodes are given in the 'Attributes' column in Table 1. By combining content information and bipartite interactions into the NCF model, on top of the Q2B model, we can not only recommend candidates based on user-defined queries but also re-rank the recommended candidates to provide more contextual recommendations and address the cold-start problem.

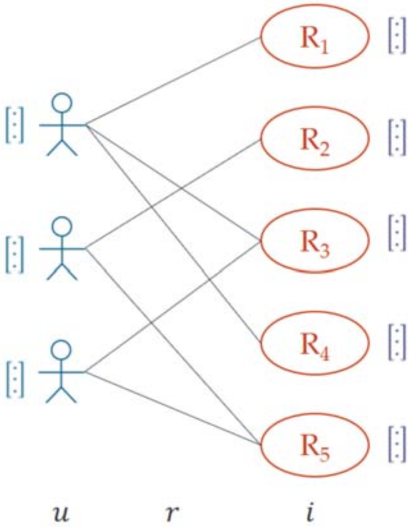


Figure 7. Example of bipartite interaction between user and item. The brackets represent the side features of users and items.

We leverage the Graph Data Science (GDS) library, provided by Neo4j, to generate GraphSAGE embeddings for nodes with context features from the domain KG. GraphSAGE is a framework for learning inductive representations on large graphs [13]. It generates low-dimensional vector representations of nodes, while utilizing their attribute information, making GraphSAGE more appropriate for training and learning KG representation for recommendation and other tasks than any other KGE model. Furthermore, it preserves the latent structural information of the network. Previous matrix factorization-based embedding frameworks, which are transductive in nature and computationally expensive, due to the fact that they can generate embeddings only for a single fixed graph and do not generalize well on unseen nodes. GraphSAGE performs sampling on the neighboring nodes and aggregates their feature representations to generate node embedding. GraphSAGE generates node embeddings using the pool aggregator function, which performs element-wise max-pooling operations to aggregate information across neighbor nodes [13].

$$AGGREGATE_k^{pool} = max\left(\left\{\sigma\left(W_{pool}h_{u_i}^k + b\right), \forall_{u_i} \in N(v)\right\}\right)$$

The learned contextual embeddings are used to train the NCF model for re-ranking the candidates. We process the final candidates, returned by the Q2B model, by utilizing user and restaurant node and attribute information to predict a rating score (r). The final recommendation score (s) for each candidate restaurant is calculated by the weighted average between relevance score (z) of the Q2B model and predicted rating score (r) of the NCF model to re-rank the final restaurant candidates.

$$s = (\alpha z + (1-\alpha)r) \ \therefore 0 \ < \ \alpha \ < \ 1$$

The α factor is the weight that determines the importance of model results to be considered for final ranking of the candidates. The relevance score (z) of the generated candidate items is obtained from the Q2B model, based on the given user query. On the other hand, the NCF model predicts the rating score (r) between the user and generated candidate items by incorporating the interaction and side information. Therefore, assigning a higher weight (α) to the rating score (r) leads in re-ranking candidates, based on more accurate contextual information.

4. Evaluation

In this part, we describe the dataset and experiments. We conducted experiments to evaluate the candidate generation and re-ranking, using the proposed recommendation framework.

4.1. Dataset

We utilized the YELP dataset [9] and stored it in the MongoDB database. We filtered out only restaurant categories to make a restaurant domain dataset. We considered 100 restaurants, with at least 20 reviews and extract entities (User, Category, Reviews, City, etc.), associated with those restaurants. For instance, we stored only those users who had left a review for those restaurants. We extracted entities and their relations that constituted a KG triple. First, we processed the structured data to extract the factual information (entity and relation), based on the defined ontology, shown in Figure 2. For example, the restaurant data contains information about *location*, *category*, etc., while the review data contains information about *reviewer* (i.e., ID, name), *review_for*, *review_text*, and *date*. Then, we manipulated this structured data, based on domain ontology, and stored it in our Neo4j graph database. Then, we applied different NLP techniques to process unstructured text, in the form of review text. Table 3 shows the data description of the restaurant domain KG.

Table 3. No. of nodes and relation in the restaurant domain KG.

Total No. of Nodes	Total No. of Relations
7750	39,158

4.2. Evaluation on Natural Query Conversion

The triple generator module was evaluated on domain-specific user queries. For evaluation settings, we defined the natural query, corresponding to the query structures used to evaluate the Q2B model on the domain KG. We also evaluated the conversion of user-defined natural queries, corresponding to the arbitrary FOL query (having any set of $\wedge$, $\vee$, and $\exists$) that satisfied the template-specific rules. The results depict that the framework generalizes well on the queries that adhere to the template rules.

Failure case analysis: The framework generated an incorrect query that lacked a valid target word (restaurant or menu). The natural query with the 'pi' structure (Table 4 (6)) does not contain a valid target word. As a result, the framework produced an incorrect query. Additionally, the query (Table 4 (10)) also failed to convert to the valid logical query segments. The 'Aspect' node in the domain KG contains 'Delivery' and 'Service' as two distinct aspects of a restaurant. Therefore, the framework extracted two aspects and produced the wrong result, containing an extra triple: 'Aspect ($R_?$, Service)', which is invalid.

4.3. Evaluation on Query2Box (Q2B) Query and Candidate Generation

To train our system on the restaurant domain KG build from the Yelp dataset, we simulated [10] the construct of a set of queries and their answers during training time and then learned the entity embeddings and geometric operators, in order to enable accurate query response. The model examines nine distinct types of query structures, as seen in Figure 8. The first five query structures were used to train the model and were evaluated on all nine query structures. The technique demonstrates a high degree of generalization, when applied to queries and logical structures not seen during training.

Table 4. Results of natural query conversion to logical triple segments. $R_?$ denotes the restaurant target entity, and $M_?$ denotes the menu target entity.

#	User Query	Query Structure	Logical Query Segments	Result
1	Recommend best Chinese restaurants	1p	Category ($R_?$, Chinese)	Correct
2	What special menus Chinese restaurants serve?	2p	Category ($R_?$, Chinese) ∧ Menu ($R_?$, $M_?$)	Correct
3	Recommend best Indian restaurant in Toronto	2i	Category ($R_?$, Indian) ∧ Location ($R_?$, Toronto)	Correct
4	Recommend best Indian restaurant in Toronto which serves Butter Chicken.	3i	Category ($R_?$, Indian) ∧ Location($R_?$, Toronto) ∧ Menu ($R_?$, Butter Chicken)	Correct
5	What special menus Indian restaurants serve in Toronto?	ip	Category ($R_?$, Indian) ∧ Location($R_?$, Toronto) ∧ Menu ($R_?$, $M_?$)	Correct
6	Who visited Indian restaurant and ordered Butter Chicken?	pi	Category ($R_?$, Indian) ∧Menu ($R_?$, Butter Chicken)	Wrong
7	Recommend restaurants which serve Butter Chicken or Chicken Biryani	2u	Menu ($R_?$, Butter Chicken) ∨ Menu ($R_?$, Chicken Biryani)	Correct
8	Which restaurants in Toronto serves Butter Chicken or Chicken Biryani	up	(Menu ($R_?$, Butter Chicken) ∨ Menu ($R_?$, Chicken Biryani)) ∧ Location ($R_?$, Toronto)	Correct
9	Recommend best Chinese restaurant in Toronto which serves sweet Noodles or spicy Chicken Biryani	arbitrary	Category ($R_?$, Chinese) ∧ Location ($R_?$, Toronto) ∧ (((Menu ($R_?$, Noodles) ∧ MenuAttrFor (Sweet, $R_?$)) ∨ (Menu ($R_?$, Butter Chicken) ∧ MenuAttrFor (Spicy, $R_?$)))	Correct
10	Which Chinese restaurants in Toronto have delivery service?	3i	Category ($R_?$, Chinese) ∧ Location ($R_?$, Toronto) ∧ Aspect ($R_?$, Delivery) ∧ Aspect ($R_?$, Service)	Wrong
11	Which Chinese restaurants in Toronto have delivery?	3i	Category ($R_?$, Chinese) ∧ Location ($R_?$, Toronto) ∧ Aspect ($R_?$, Delivery)	Correct

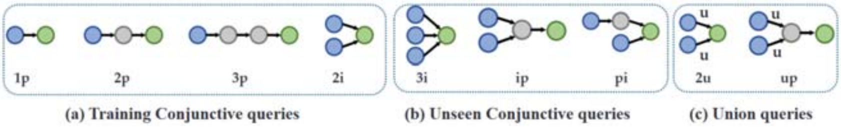

1p
2p
3p
2i
3i
ip
pi
u
u
2u
u
u
up
(a) Training Conjunctive queries
(b) Unseen Conjunctive queries
(c) Union queries

Figure 8. Query structures for experiments, where 'p', 'i', and 'u' stand for 'projection', 'intersection', and 'union', respectively.

We follow a similar evaluation protocol to that presented in [10], which is briefly stated here; the data preparation required splitting the KG edges into training, test, and evaluation sets and began by augmenting the KG to include inverse relations, thus doubling the amount of edges in the graph. Following that, we constructed three graphs: $G_{train} \subseteq G_{valid} \subseteq G_{test}$. While G_{train}, contains only training edges and is used to train node embeddings, as well as box operators, G_{valid} contains G_{train}, as well as the validation edges, and G_{test} includes G_{valid}, as well as the test edges. Considering the nine query structures presented in [10] for training the graphs, 80% of generated queries were utilized for training, 10% were used for validation, and the remaining 10% were used for testing. For a given query (q), the denotation sets (answer entities sets) $[|q|]_{train}$, $[|q|]_{valid}$, and $[|q|]_{test}$ were obtained by running subgraph matching of q on G_{train}, G_{valid}, and G_{test}, respectively. $[|q|]_{train}$ were utilized as positive samples for the query during training, while negative samples were generated from other random entities. However, for testing and validation, the method was validated against only those answers that have missing relations, instead of validating against whole validation $[|q|]_{valid}$ or test $[|q|]_{test}$ sets of answers. Table 5 summarizes the results of the Q2B model simulation on a restaurant domain dataset for different query structures. As stated in [10], the complex logical queries (particularly 2p, 3p, ip, pi, and up) require modeling a significantly greater number of answer entities (often more than 10 times) than the simple 1p queries do. Therefore, it is expected that the box embeddings will perform well when handling complex queries with a large number of answer entities; this was also observed in our experiments, particularly for queries with 2i, 3i, ip, and pi. According to our restaurant domain ontology triple dataset, queries with 2i 3i, ip, and pi structures are frequently observed with a greater number of entities than other query structures and, therefore, produced better results on the Yelp (domain-KG) dataset than other datasets. The model performs comparably to FB15k-237 and NELL995 but is inferior to FB15k on our generated restaurant domain KG.

Table 5. Results of Q2B on restaurant domain dataset for different query structures. The other dataset results are illustrated from the original paper for comparison.

Query	Avg	1p	2p	3p	2i	3i	ip	pi	2u	up
					Yelp (domain-KG) dataset					
MRR	0.286	0.309	0.164	0.144	0.402	0.604	0.234	0.38	0.171	0.168
Hits@1	0.188	0.19	0.062	0.05	0.334	0.545	0.143	0.26	0.012	0.099
Hits@3	0.347	0.392	0.234	0.207	0.429	0.627	0.287	0.46	0.289	0.201
Hit@10	0.44	0.46	0.362	0.303	0.498	0.699	0.399	0.535	0.393	0.308
					FB15k					
MRR	0.41	0.654	0.373	0.274	0.488	0.602	0.194	0.339	0.468	0.301
Hits@3	0.484	0.786	0.413	0.303	0.593	0.712	0.211	0.397	0.608	0.33
					FB15k-237					
MRR	0.235	0.4	0.225	0.173	0.275	0.378	0.105	0.18	0.198	0.178
Hits@3	0.268	0.467	0.24	0.186	0.324	0.453	0.108	0.205	0.239	0.193
					NELL995					
MRR	0.254	0.413	0.227	0.208	0.288	0.414	0.125	0.193	0.266	0.155
Hits@3	0.306	0.555	0.266	0.233	0.343	0.48	0.132	0.212	0.369	0.163

Our recommendation system enables generating results for the user-specific queries in an explainable way, based on user and query context.

> *"Recommend best Indian restaurant in Toronto which serves sweet Butter Chicken"*
> Query (1)

Q2B produces set of candidate restaurant for a given logical query. An equivalent illustration of the Q2B result for the Query (1) is given in Figure 9, using cypher in Neo4j.

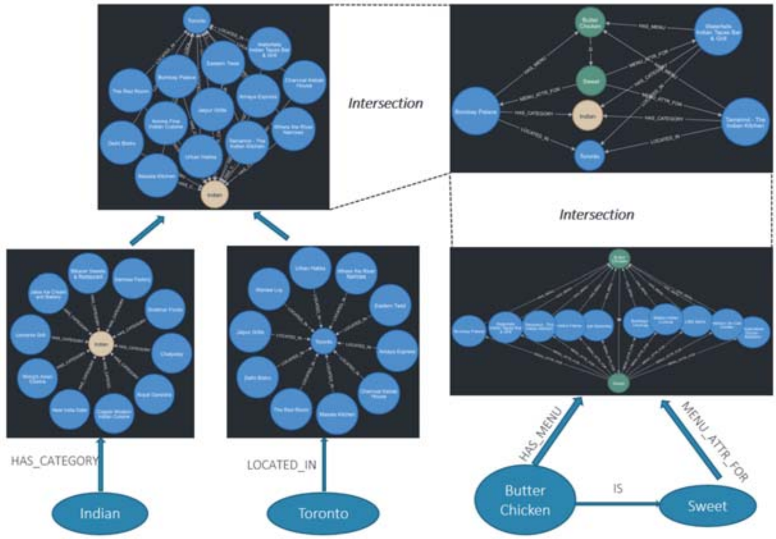


Figure 9. Visual explainable equivalent cypher query result from user Query (1).

The cypher query approach produces actual candidates from the domain KG. It cannot recommend candidates if no match is found. However, Q2B aims to recommend the item, not to perform a search from the database. Therefore, it produces *top-K*, the semantically similar and closest candidates for each query triple. Hence, it can recommend items, even if there is no match found in the KG. Table 6 depicts the Q2B results for the input Query (1). The resultant restaurant candidates in bold (Rank 1, 2, and 3) are the actual restaurants that satisfy the result for input Query (1) from the stored KG, as shown in Figure 10. The candidate restaurants with Ranks 4 and 5, on the other hand, missed some entities or relationships to match exact candidates based on the given query. However, because the recommender system's task is to recommend the *top-K* candidates, we consider the top five results, based on the semantic and contextual similarity of the query and KG. Additionally, this approach of producing candidates enables the generation of explainable findings. We define explainability, in terms of KG paths, for recommended candidates.

Table 6. Q2B results for input Query (1).

Rank	RestID	Q2B (z)
1	**4873**	**7.478498**
2	**4641**	**7.431061**
3	**3744**	**3.110984**
4	3926	1.378697
5	5324	0.376228

The result of the Q2B model can be defined for the explainable recommendation. We demonstrate the visual explainability of each generated restaurant candidate for input query (1) in Figure 11. The projection of the resulting restaurant candidates with connected paths determines the applicability of the recommendation. From the explicable approach, we show that candidate 4 does not match the attribute '*Sweet*', while candidate 5 does not match the menu, as well as attribute entity (*Butter Chicken*, *Sweet*). However, the other facts of these candidates are matched with higher semantic and structural similarity and obtained a relevance score lower than the exact match candidates of the query from the KG.

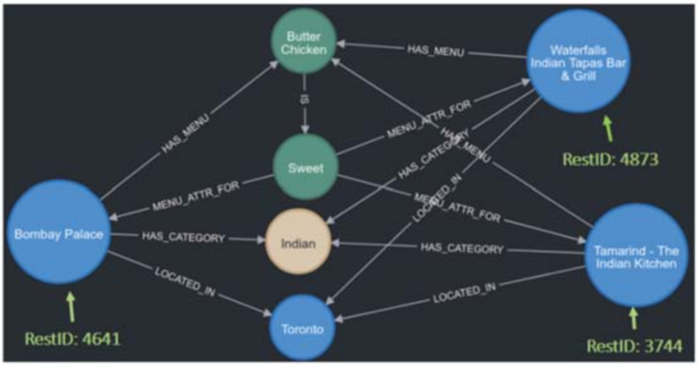


Figure 10. Comparison of Q2B results with native cypher query results for user Query (1).

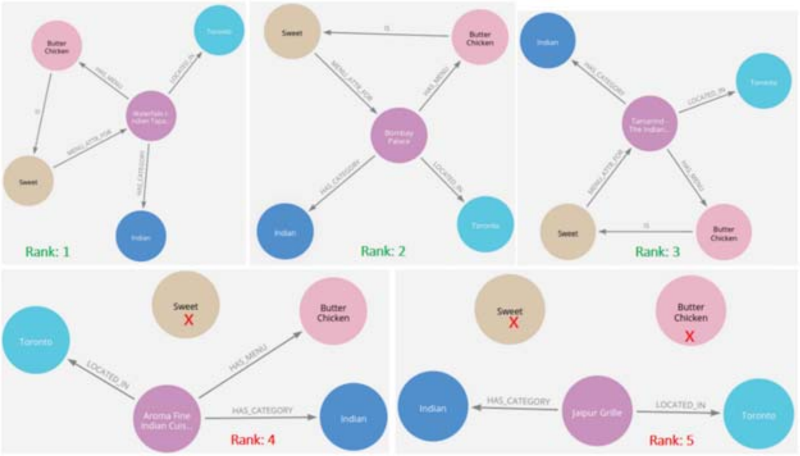


Figure 11. Visualization of explainable results from the domain KG for candidates, obtained from the Q2B model.

4.4. Evaluation of Neural Collaborative Filtering (NCF)-based Re-Rank Module

The scope of our recommender system is not only limited to domain KG-based explainable recommendation. It is a two-stage procedure, in which, similar to other factorization approaches, we employ a ML-based NCF model to re-rank the generated candidates from the previous stage, in order to incorporate user and candidate restaurant context information. We train a NCF model by leveraging contextual GraphSAGE embedding. We generate contextual GraphSAGE embedding by employing the entities and attributes from the domain KG, stored in the Neo4j database. The dataset for training the model is shown in Table 7.

Table 7. Dataset for model training.

Train	Test
30,000	4524

The proposed approach of utilizing GraphSAGE, embedding with the NCF network model, outperforms the classic matrix factorization method on the domain KG.

Table 8 shows the ablation study for model training, while in Table 9, we compare our proposed approach with the matrix factorization model.

Table 8. Ablation results of the NCF model training with GraphSAGE features.

Epochs	MAE	RMSE
5000	1.0690	1.5949
10,000	1.0674	1.5925
20,000	**1.0673**	**1.5917**

Table 9. NCF model comparison with matrix factorization technique.

Method	MAE	RMSE
MF	1.169	1.994
NCF Model	**1.0673**	**1.5917**

As discussed earlier, the NCF model processes candidates from the Q2B results and predicts a rating score (r) for the user and candidate restaurant pair. The final score (s) is then taken as a weighted average between the relevance score (z) of the Q2B model and predicted rating score (r) of the NCF model. The demonstration of the model results, between generated restaurant candidates in Table 6 and two users (*UserID:3178*, and *UserID:17*), pair is shown in Table 10. To calculate final score (s), we consider $\alpha = 0.3$ to assign more attention to r. The final candidate items (restaurants) are re-ranked, based on the final score.

Table 10. Re-rank model results.

Rank	RestID	Q2B (z)	UserID: 3178			UserID: 17		
			NCF (r)	Score (s)	Re-Ranking	NCF (r)	Score (s)	Re-Ranking
1	4873	7.478498	4.3	5.25	2	4.2	5.18	2
2	4641	7.431061	4.7	5.52	1	4.5	5.38	1
3	3744	3.110984	4.8	4.29	3	3.9	3.66	3
4	3926	1.378697	4.2	3.35	4	4.0	3.21	4
5	5324	0.376228	4.1	2.98	5	3.5	2.56	5

We further validate the NCF model results by applying cosine similarity on the GraphSAGE embeddings, learned using the Neo4j GDS framework. The cosine similarity computes the similarity between two vectors, which could be then utilized in a recommendation query. For instance, to obtain restaurant recommendations based on the preferences of users who have previously given similar ratings to other restaurants visited by the user. We measure the similarity between user (*UserID: 3178*) and resultant restaurant candidate pairs from the Q2B. The results in Table 11 show that the restaurant with (*RestID: 4641*) and (*RestID: 3744*) had the first and second highest similarity, respectively, which the NCF model also predicted as higher.

Table 11. NCF model results comparison, with the cosine similarity algorithm applied on GraphSAGE embedding, using the Neo4j GDS framework.

RestID	RestName	Similarity	NCF (r)
4641	Bombay Palace	0.99996	4.7
3744	Tamarind—The Indian Kitchen	0.99993	4.8
3926	OM Restaurant	0.99988	4.2
4873	Waterfalls Indian Tapas Bar & Grill	0.99985	4.3
5324	Pakwan Indian Bistro	0.99903	4.1

5. Conclusions

We proposed a novel, domain-specific, context-aware, explainable recommendation framework, based on a domain knowledge graph (KG) and machine learning model. Our proposed recommender system deals with user-defined natural language query. It consists of three modular stages: (1) decompose a complex natural language query into segments of logical triples that reflect the semantic and structural mapping over the domain knowledge graph; (2) find the final set of candidate restaurants with relevance score (z) by performing logical conjunction and or disjunction operations, using the Q2B model; and (3) predicts the rating score (r) for the final candidate items, through the neural collaborative filtering model, by incorporating user and candidate items latent vectors. Through the evaluation of its application to the restaurant domain, with Yelp data, it is shown that the proposed approach works effectively for a domain-specific system. We have shown that combining KG techniques with the traditional collaborative filtering approach can solve data sparsity cold-start problems and provide a content-aware explainable recommendation. Careful design of domain ontology is important for the decomposition of natural queries into triples for a domain KG. However, extending domain ontology with additional parameters and developing more robust rules for natural query processing could result in complex query answering and parsing. Moreover, replacing the candidate generation and re-rank processes with a single GraphSAGE framework could enable inductive learning and does not require the use of two distinct models for reasoning over KG and learning its representation (nodes, relation, and attributes) for creating latent vector separately.

Author Contributions: Conceptualization, M.H.S.; methodology, M.H.S.; software, M.H.S. and T.Q.B.H.; validation, M.H.S. and T.Q.B.H.; formal analysis, M.H.S.; data curation, M.H.S.; writing—original draft preparation, M.H.S.; writing—review and editing, S.-T.C.; supervision, S.-T.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the AI-based High Added Value New Product Technology Development Program (Project No. S2953591) and SME R&D project for the Start-up & Grow stage company (Project No. S3044682) of the Ministry of SMEs and Startups (MSS, Korea).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Wang, H.; Zhang, F.; Wang, J.; Zhao, M.; Li, W.; Xie, X.; Guo, M. RippleNet: Propagating User Preferences on the Knowledge Graph for Recommender Systems. In Proceedings of the 27th ACM International Conference on Information and Knowledge Management, Torino, Italy, 22–26 October 2018; pp. 417–426.
2. Covington, P.; Adams, J.; Sargin, E. Deep Neural Networks for YouTube Recommendations. In Proceedings of the 10th ACM Conference on Recommender Systems, Boston, MA, USA, 15–19 September 2016; ACM: New York, NY, USA, 2016; pp. 191–198.
3. Wang, H.; Zhang, F.; Xie, X.; Guo, M. DKN: Deep Knowledge-Aware Network for News Recommendation. In Proceedings of the 2018 World Wide Web Conference (WWW), Lyon, France, 23–27 April 2018; pp. 1835–1844.
4. Yu, X.; Ren, X.; Sun, Y.; Gu, Q.; Sturt, B.; Khandelwal, U.; Norick, B.; Han, J. Personalized entity recommendation: A heterogeneous information network approach. In Proceedings of the 7th International Conference Web Search and Data Mining (WSDM), New York, NY, USA, 24–28 February 2014; ACM: New York, NY, USA, 2014; pp. 283–292.
5. Wang, H.; Zhang, F.; Zhao, M.; Li, W.; Xie, X.; Guo, M. Multi-Task Feature Learning for Knowledge Graph Enhanced Recommendation. In Proceedings of the 2019 World Wide Web Conference (WWW), San Francisco, CA, USA, 13–17 May 2019; Association for Computing Machinery: New York, NY, USA, 2019; pp. 2000–2010.
6. Wang, X.; He, X.; Cao, Y.; Liu, M.; Chua, T.S. KGAT: Knowledge Graph Attention Network for Recommendation. In Proceedings of the 25th ACM SIGKDD International Conference of Knowledge Discovery & Data Mining (KDD), Anchorage, AK, USA, 4–8 August 2019; Association for Computing Machinery: New York, NY, USA, 2019; pp. 950–958.
7. Gao, Y.; Li, Y.F.; Lin, Y.; Gao, H.; Khan, L. Deep learning on knowledge graph for recommender system: A survey. *arXiv* **2020**, arXiv:2004.00387.

8. Hogan, A.; Blomqvist, E.; Cochez, M.; d'Amato, C.; Melo, G.; Gutierrez, C.; Gayo, J.E.L.; Kirrane, S.; Neumaier, S.; Polleres, A.; et al. Knowledge graphs. *ACM Comput. Surv.* **2021**, *54*, 1–37. [CrossRef]
9. Yelp.com. Available online: https://www.yelp.com/ (accessed on 23 October 2021).
10. Ren, H.; Hu, W.; Leskovec, J. Query2box: Reasoning over knowledge graphs in vector space using box embeddings. In Proceedings of the International Conference Learning Representations, New Orleans, LA, USA, 6–9 May 2019.
11. Rendle, S.; Gantner, Z.; Freudenthaler, C.; Thieme, L.S. Fast context-aware recommendations with factorization machines. In Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, Beijing, China, 24–28 July 2011; pp. 635–644.
12. He, X.; Liao, L.; Zhang, H.; Nie, L.; Hu, X.; Chua, T.-S. Neural collaborative filtering. In Proceedings of the 26th International Conference on World Wide Web (WWW 2017), Perth, Australia, 3–7 April 2017; pp. 173–182. Available online: https://dblp.org/rec/conf/www/HeLZNHC17 (accessed on 13 November 2021).
13. Hamilton, W.; Ying, Z.; Leskovec, J. Inductive representation learning on large graphs. In Proceedings of the NeuralPS, Long Beach, CA, USA, 4–9 December 2017; Curran Associates Inc.: Red Hook, NY, USA, 2017; pp. 1024–1034.
14. Hao, X.; Ji, Z.; Li, X.; Yin, L.; Liu, L.; Sun, M.; Liu, Q.; Yang, R. Construction and Application of a Knowledge Graph. *Remote Sens.* **2021**, *13*, 2511. [CrossRef]
15. Zhu, G.; Iglesias, C.A. Sematch: Semantic entity search from knowledge graph. In Proceedings of the SumPre 2015—1st International Workshop Extended Semantic Web Conference (ESWC), Portoroz, Slovenia, 1 June 2015.
16. Yasunaga, M.; Ren, H.; Bosselut, A.; Liang, P.; Leskovec, J. QA-GNN: Reasoning with Language Models and Knowledge Graphs for Question Answering. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Online, 6—11 June 2021; pp. 535–546. Available online: https://aclanthology.org/2021.naacl-main.45/ (accessed on 13 November 2021).
17. Papadopoulos, D.; Papadakis, N.; Litke, A. A Methodology for Open Information Extraction and Representation from Large Scientific Corpora: The CORD-19 Data Exploration Use Case. *Appl. Sci.* **2020**, *10*, 5630. [CrossRef]
18. Wang, M.; Qiu, L.; Wang, X. A Survey on Knowledge Graph Embeddings for Link Prediction. *Symmetry* **2021**, *13*, 485. [CrossRef]
19. Bastos, A.; Nadgeri, A.; Singh, K.; Mulang, I.O.; Shekarpour, S.; Hoffart, J.; Kaul, M. RECON: Relation Extraction using Knowledge Graph Context in a Graph Neural Network. In Proceedings of the World Wide Web, Ljubljana, Slovenia, 19–23 April 2021; pp. 1673–1685.
20. Arakelyan, E.; Daza, D.; Minervini, P.; Cochez, M. Complex Query Answering with Neural Link Predictors. *arXiv* **2011**, arXiv:2011.03459.
21. Bordes, A.; Usunier, N.; García-Durán, A.; Weston, J.; Yakhnenko, O. Translating embeddings for modeling multi-relational data. In Proceedings of the Advances in Neural Information Processing Systems 26 (NIPS 2013), Stateline, NV, USA, 5—10 December 2013; pp. 2787–2795. Available online: https://papers.nips.cc/paper/2013/hash/1cecc7a77928ca8133fa24680a88d2f9-Abstract.html (accessed on 13 November 2021).
22. Choudhary, S.; Luthra, T.; Mittal, A.; Singh, R. A survey of knowledge graph embedding and their applications. *arXiv* **2021**, arXiv:2107.07842.
23. Wang, Z.; Zhang, J.; Feng, J.; Chen, Z. Knowledge graph embedding by translating on hyperplanes. In Proceedings of the 28th AAAI Conference of Artificial Intelligence, Quebec City, QC, Canada, 27–31 July 2014; pp. 1112–1119.
24. Lin, Y.; Liu, Z.; Sun, M.; Liu, Y.; Zhu, X. Learning entity and relation embeddings for knowledge graph completion. In Proceedings of the 29th AAAI, Austin, TX, USA, 25–30 January 2015; pp. 2181–2187.
25. Nickel, M.; Tresp, V.; Kriegel, H.P. A three-way model for collective learning on multi-relational data. In Proceedings of the 28th International Conference on Machine Learning, Bellevue, WA, USA, 28 June –2 July 2011; pp. 809–816.
26. Yang, B.; Yih, W.T.; He, X.; Gao, J.; Deng, L. Embedding entities and relations for learning and inference in knowledge bases. In Proceedings of the International Conference on Learning Representations, San Diego, CA, USA, 7–9 May 2015; pp. 1–12.
27. Dettmers, T.; Minervini, P.; Stenetorp, P.; Riedel, S. Convolutional 2D knowledge graph embeddings. In Proceedings of the AAAI, Quebec City, QC, Canada, 27–31 July 2014; pp. 1811–1818.
28. Shah, L.; Gaudani, H.; Balani, P. Survey on Recommendation System. *Int. J. Comp. Appl.* **2016**, *137*, 43–49. [CrossRef]
29. Koren, Y.; Bell, R.; Volinsky, C. Matrix factorization techniques for recommender systems. *Computer* **2009**, *42*, 30–37. [CrossRef]
30. Zhang, H.; Ganchev, I.; Nikolov, N.S.; Ji, Z.; O'Droma, M. FeatureMF: An Item Feature Enriched Matrix Factorization Model for Item Recommendation. *IEEE Access* **2021**, *9*, 65266–65276. [CrossRef]
31. He, X.; Chua, T.S. Neural factorization machines for sparse predictive analytics. In Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, Tokyo, Japan, 7–11 August 2017; pp. 355–364.
32. Cheng, H.T.; Koc, L.; Harmsen, J.; Shaked, T.; Chandra, T.; Aradhye, H.; Anderson, G.; Corrado, G.; Chai, W.; Ispir, M. Wide & deep learning for recommender systems. In Proceedings of the 1st Workshop Deep Learning for Recommender Systems, Boston, MA, USA, 15 September 2016; pp. 7–10.
33. Lian, J.; Zhou, X.; Zhang, F.; Chen, Z.; Xie, X.; Sun, G. XDeepFM: Combining explicit and implicit feature interactions for recommender systems. In Proceedings of the 24th ACM SIGKDD International Conference of Knowledge Discovery and Data Mining, London, UK, 19–23 August 2018; pp. 1754–1763.

34. Zhang, F.; Yuan, N.J.; Lian, D.; Xie, X.; Ma, W.Y. Collaborative knowledge base embedding for recommender systems. In Proceedings of the 22nd. International Conference of ACM SIGKDD on KDD, San Francisco, CA, USA, 13–17 August 2016; Association for Computing Machinery: New York, NY, USA, 2016; pp. 353–362.
35. Guo, Q.; Zhang, F.; Qin, C.; Zhu, H.; Xie, X.; Xiong, H.; He, Q. A survey on knowledge graph-based recommender systems. *arXiv* **2020**, arXiv:2003.00911. [CrossRef]
36. Liu, C.; Li, L.; Yao, X.; Tang, L. A Survey of Recommendation Algorithms Based on Knowledge Graph Embedding. In Proceedings of the IEEE International Conference on Computer Science and Education informalization (CSEI), Xinxiang, China, 16–19 August 2019; pp. 168–171. [CrossRef]
37. Zhang, S.; Yao, L.; Sun, A.; Tay, Y. Deep learning based recommender system: A survey and new perspectives. *ACM Comput. Surv.* **2019**, *52*, 1–38. [CrossRef]
38. Sitar-Tăut, D.-A.; Mican, D.; Buchmann, R.A. A knowledge-driven digital nudging approach to recommender systems built on a modified Onicescu method. *Expert Syst. Appl.* **2021**, *181*, 115170. [CrossRef]
39. Zhu, Y.; Gong, Y.; Liu, Q.; Ma, Y.; Ou, W.; Zhu, J.; Wang, B.; Guan, Z.; Cai, D. Query-based interactive recommendation by meta-path and adapted attention-gru. In Proceedings of the 28th ACM International Conference on Information and Knowledge Management, Beijing, China, 3–7 November 2019; pp. 2585–2593.
40. Bhattacharya, M.; Barapatre, A. Query as Context for Item-to-Item Recommendation. *Comput. J.* **2021**, *64*, 1016–1027.
41. Ren, H.; Leskovec, J. Beta embeddings for multi-hop logical reasoning in knowledge graphs. In Proceedings of the Advances in Neural Information Processing System, Vancouver, Canada, 6–12 December 2020. Available online: https://papers.nips.cc/paper/2020/hash/e43739bba7cdb577e9e3e4e42447f5a5-Abstract.html (accessed on 13 November 2021).
42. Pirro, G. Explaining and suggesting relatedness in knowledge graphs. In *ISWC*; Springer: Cham, Switzerland, 2015; pp. 622–639.
43. Feddoul, L. Semantics-driven Keyword Search over Knowledge Graphs. In Proceedings of the DC@ISWC, Vienna, Austria, 3 November 2020; pp. 17–24.
44. Yan, H.; Wang, Y.; Xu, X. SimG: A Semantic Based Graph Similarity Search Engine. In Proceedings of the 27th International Conference of Advanced Cloud and Big Data, Suzhou, China, 21–22 September 2019; pp. 114–120. [CrossRef]
45. Hamilton, W.; Bajaj, P.; Zitnik, M.; Jurafsky, D.; Leskovec, J. Embedding logical queries on knowledge graphs. In *NeurIPS*; Curran Associates Inc.: Red Hook, NY, USA, 2018; pp. 2030–2041.
46. Ochieng, P. PAROT: Translating natural language to SPARQL. *Expert Syst. Appl. X* **2020**, *5*, 100024.

big data and
cognitive computing

MDPI

Article

The Predictive Power of a Twitter User's Profile on Cryptocurrency Popularity

Maria Trigka [1], Andreas Kanavos [2], Elias Dritsas [1,*], Gerasimos Vonitsanos [1] and Phivos Mylonas [3]

[1] Computer Engineering and Informatics Department, University of Patras, 26504 Patras, Greece; trigka@ceid.upatras.gr (M.T.); mvonitsanos@ceid.upatras.gr (G.V.)
[2] Department of Digital Media and Communication, Ionian University, 28100 Kefalonia, Greece; akanavos@ionio.gr
[3] Department of Informatics, Ionian University, 49100 Corfu, Greece; fmylonas@ionio.gr
* Correspondence: dritsase@ceid.upatras.gr

Abstract: Microblogging has become an extremely popular communication tool among Internet users worldwide. Millions of users daily share a huge amount of information related to various aspects of their lives, which makes the respective sites a very important source of data for analysis. Bitcoin (BTC) is a decentralized cryptographic currency and is equivalent to most recurrently known currencies in the way that it is influenced by socially developed conclusions, regardless of whether those conclusions are considered valid. This work aims to assess the importance of Twitter users' profiles in predicting a cryptocurrency's popularity. More specifically, our analysis focused on the user influence, captured by different Twitter features (such as the number of followers, retweets, lists) and tweet sentiment scores as the main components of measuring popularity. Moreover, the Spearman, Pearson, and Kendall Correlation Coefficients are applied as post-hoc procedures to support hypotheses about the correlation between a user influence and the aforementioned features. Tweets sentiment scoring (as positive or negative) was performed with the aid of Valence Aware Dictionary and Sentiment Reasoner (VADER) for a number of tweets fetched within a concrete time period. Finally, the Granger causality test was employed to evaluate the statistical significance of various features time series in popularity prediction to identify the most influential variable for predicting future values of the cryptocurrency popularity.

Keywords: blockchain; cryptocurrency; Kendall Correlation Coefficient; Pearson Correlation Coefficient; sentiment analysis; social media analytics; Spearman Correlation Coefficient; Twitter; user influence

Citation: Trigka, M.; Kanavos, A.; Dritsas, E.; Vonitsanos, G.; Mylonas, P. The Predictive Power of a Twitter User's Profile on Cryptocurrency Popularity. *Big Data Cogn. Comput.* **2022**, *6*, 59. https://doi.org/10.3390/bdcc6020059

Academic Editors: Konstantinos Kotis and Dimitris Spiliotopoulos

Received: 21 March 2022
Accepted: 18 May 2022
Published: 20 May 2022

1. Introduction

A social network is a social structure that consists of nodes (e.g., unique users, businesses, artistic profiles, etc.), which are connected to each other by various types of interdependence (e.g., kinship, friendship, sympathy, admiration, curiosity, financial relations). In recent years, however, the usefulness of these networks, as well as the extensions they have taken on in our lives, make any definition rather incomplete.

Twitter is a tool for microblogging [1] and a social networking platform that appeared in March 2006 and still remains among the most visited websites in the world. The power of Twitter is essentially the production of news in real-time, and it remains today one of the best indicators of what is happening in the world at any given time. This is really amazing, considering that the original idea behind its creation was a platform that allows a registered user to compile and publish a status of up to 280 characters.

Influencer marketing and consequently influencers, as well as the ability provided by the data of millions of Twitter users to create or predict trends, thus determining even the global fluctuations of stock prices, are the main aspects discussed in this study. Influencers in the world outside of social networks are persons who have the ability to influence the choices of others because of their knowledge, their professional reputation, or their personal relationship that they have managed to develop with a certain portion of the audience. This audience can be influenced by each influencer to a small and sometimes to a greater extent. The natural question that arises is whether a social network can be the appropriate digital platform in which this power of influence of an individual can be measured, calculated, and in some way be a product of study.

When trying to empirically measure the impact of a Twitter account or any other social network, the following question will be triggered: what is the content that primarily increases the loyalty or commitment of an existing audience? Engagement is essentially considered as the way of discriminating whether the content of an account manages to keep the interest of its audience, which would result in a potential increase in the number of followers.

The ubiquity of Internet access has triggered the emergence of currencies distinct from those used in the prevailing monetary system. The advent of cryptocurrencies based on a unique method called "mining" has brought about significant changes in the online financial activities of users. Various cryptocurrencies have appeared since 2008, when Bitcoin was first introduced [2,3]. Nowadays, cryptocurrencies are often used in online transactions, and their usage has increased every year since their introduction [4,5].

Cryptocurrencies are mainly characterized by fluctuations in their price and number of transactions [3,4]. For instance, the most famous cryptocurrency, Bitcoin, did not fluctuate significantly in price and number of transactions until the end of 2013, when it began to attract global attention, and marked a significant increase and fluctuation in price and number of transactions [4]. Bitcoin quickly gained interest as a possible replacement for standard monetary forms. Other cryptocurrencies, such as Ripple and Litecoin, have shown significantly unstable fluctuations since the end of December 2013 [6]. Such volatile fluctuations have served as an opportunity for some users to speculate while preventing most others from using cryptocurrencies [3,7,8]. In this way, the plethora of objects, opinions, and information about Bitcoin are predominant through the majority of social media sphere [9]. In addition, the Bitcoin currency is considered the modern principal cryptocurrency that could even replace other currencies [10].

Twitter constitutes a platform on which peoples' thoughts can be almost automatically translated into digital information. Nonetheless, one of the most important issues for the supporters of Bitcoin is not only the sharp fluctuation of its exchange rate but also the factors that influence these fluctuations. Sentiment analysis in Twitter has been extensively studied in numerous works that demonstrate the potential of this topic [11–14]. Based on these thoughts, in this article, we made a statistical causality test for investigating whether sentiment, followers, retweets, favorites, and lists time series are effective in forecasting the popularity of two cryptocurrencies. Finally, we conclude on the popularity of cryptocurrencies in users' list timelines.

This study presents a comparison of the popularity of four popular cryptocurrencies, i.e., Bitcoin, Ethereum, Litecoin, and Stellar, based on different features that can be identified in the posts of Twitter users. These characteristics are the number of followers, the ratio of retweets per tweet, the ratio of favorites per tweet, and the number of lists to which the user belongs. Furthermore, the dataset used in the paper consists of 12,000 posts collected for a time period of 12 days, from 6 April to 18 April 2020. More to the point, the timelines of the 500 most influential users were taken into consideration. As a next step, we applied the Spearman, Pearson, and Kendall Correlation Coefficients as post-hoc procedures to support hypotheses about the correlation between these four features. Finally, the Granger causality test was employed to evaluate the statistical significance of various features time series in popularity prediction as it identified the most influential variable to predict future values of cryptocurrency popularity.

The rest of this paper is structured as follows: Section 2 presents related works in the field of Blockchain and Cryptocurrency, as well as sentiment analysis in Cryptocurrencies. Section 3 analyzes the proposed architecture and the tools required for its implementation.

Next, Section 4 describes and analyzes the features of the used dataset and provides the experimental results, including correlation analysis and statistical tests. Finally, we summarize the paper and conclude with future work in Section 5.

2. Related Work

Numerous studies conducted empirical analyses regarding the economic considerations of cryptocurrencies, including market efficiency [15–17], price movements and their determinants [18,19], and price discovery [20,21]. Moreover, some other papers examine the existence of herding in the cryptocurrency market [22,23]. The analysis of the existence of herding in the cryptocurrency market is of paramount importance since the presence of this phenomenon would give rise to an inefficient market in which asset pricing models based on rational economic behavior cannot be properly applied. In this paper, we will delve into the effect of sentiment analysis and the user's influence on the popularity of cryptocurrency on Twitter.

2.1. Blockchain and Cryptocurrencies Technology

In the last decades, the huge technological advances have managed to reshape or even radically change most, if not all, business sectors. More specifically, in the field of economics, this technological explosion has managed not only to improve and facilitate marketing methods but also to ask questions about money and whether its very form can be transformed into an alternative genre, much more transparent, and for the most part highly compatible with the digital world.

Nowadays, due to the evolution of internet platforms and social media, cryptocurrency remains a challenging issue to investigate. Cryptocurrency is predicted to become the future currency that could disrupt the present paper currency around the world [24]. In addition, the opportunities in cryptocurrency, such as the high investment return, the low transaction cost, and the security of its technology were discussed. Authors in [25] surveyed several widely used cryptocurrency systems such as Auroracoin, Bitcoin, Blackcoin, Dash, Decred, Ethereum, Litecoin, Namecoin, Peercoin, Permacoin, and Ripple.

The key element in the operation of cryptocurrencies at the technological and structural level is the Blockchain technology. Blockchain could be described as a database form that accepts a large number of user registrations. These records are placed in a data sheet, also known as a block, and over time, these records grow, and the blocks that are created are connected to each other in the form of a chain. This feature makes the blockchain look like an account book, open to all users, which verifies its designation as the most decentralized trading system.

The impact of government pseudo-events on changes in public discourse on controversial technologies is examined in [26]. The authors focused on changes in the public discourse on Twitter about cryptocurrency and blockchain technology, according to the different government agencies' announcements regarding the regulation of domestic cryptocurrency transactions.

2.2. Sentiment Analysis in Cryptocurrencies

In recent years, there has been a growing interest in Sentiment Analysis exploiting data from social media, e.g., Twitter [27], and especially in discussion posts that review users' opinions and feelings on cryptocurrencies [28].

The work in [29] attempted to predict whether sentiment analysis in Twitter posts that are related to Bitcoin can be regarded as a predictive premise to show if the Bitcoin price will increase or decrease. Authors in [30] outline several machine learning pipelines with the aim of making Sentiment Analysis on Twitter Data and identifying the Bitcoin cryptocurrency market movement. They apply several supervised learning algorithms and achieve prediction on an hour as well as a daily basis with accuracy exceeding 90%. Similar work is presented in [31], where the way that prices of the cryptocurrencies mutually behave and are consistently related to the sentiment values identified through Twitter and

StockTwits messages is investigated. Authors examine whether a specific characteristic structure is considered within a market and enquire what the location is of the major cryptocurrencies within this structure.

In [32], an approach for the prediction of changes in the prices of Bitcoin and Ethereum that utilize Twitter data are proposed. The ultimate goal of this work is to employ sentiment analysis techniques to retrieve tweets in order to determine if the tweets are generally positive or negative in their opinions of cryptocurrencies. Authors in [33] investigated whether blockchain ventures can efficiently use Twitter signaling for increasing funding; natural language processing techniques to a 144, 492 tweets dataset related to 522 ventures for creating features in terms of regression models were applied.

Furthermore, authors in [34] estimate the relationship between Bitcoin price and sentiment extracted from social media, assuming lexicon-based sentiment analysis. In [35], researchers are concerned about the Bitcoin currency on the internet and on social media platforms to determine the importance and value of Bitcoin based on users' discussions and points of view through which a sentiment analysis of the users' tweets is carried out. The work in [36] utilizes the happiness in Twitter posts as a new feature for investors' analytics and studies its dependency on the returns of five popular cryptocurrencies.

Finally, in a more recent work in [37], the authors examine Twitter signals as a method for sentiment analysis in order to forecast the price alternations of the ZClassic cryptocurrency. The posts retrieved for a time interval of 3.5 weeks were classified with the use of a Gradient Boosting Regression Tree Model as positive, negative, or neutral.

2.3. User Profiling and Influence

User profiling has gained significant interest in the last several years, and several works have been occupied and thoroughly study this problem. In particular, the authors in [38] proposed a novel context-aware knowledge model schema as well as a method for the dynamic activation of user preferences with the aim of efficiently representing user interests in coherence with occurring user activities.

There have been numerous works that target influence and influencers on Twitter [39,40]. Within the same scope, but in a different domain, an influence method in GitHub that focuses on identifying and comparing influence metrics is reported in [41]. The number of followers depicts the popularity of a GitHub member, whereas the number that the developer's repositories were "forked" constitutes a measure of the value of the created content. User influence can be considered a measure that is related to the interest of the followers (using favorites, mentions, replies, and retweets) on the Twitter social network. The study in [42] focuses on analyzing the metrics of influence for all the users that took part in certain discussions and verifying the differences between them.

Moreover, user comments and replies found in online communities for predicting the number of transactions as well as the price of cryptocurrencies are utilized in [43]. These aspects showed their efficacy by affecting the number of transactions between users; this approach was examined and found to be efficient for buying and selling cryptocurrencies, as well as identifying aspects influencing user opinions.

There are six basic principles that govern any attempt to persuade a portion of the public to a new product on the market or even to adopt a new habit, namely, reciprocity, commitment and consistency, social proof, liking, authority as well as scarcity [44].

Focusing on whether or not one person is able to influence others highlights three specific actions that a Twitter user can take. The first step in actually expressing a user's interest is linking to accounts whose content is considered interesting. In addition, users often share with their followers information that they find interesting. This aspect is recognized by the retweet caption, i.e., @username to be included in the tweet. The third and last action is the ability for the user to reply to or comment on a specific post. These three activities undoubtedly reflect the three different forms of user influence [45]:

- Influence based on Followers: this number indicates the size of the user's "audience".

- Influence based on Retweets: this type of influence indicates the user's ability to produce content with timeless value or tweets that users can easily share.
- Influence based on Replies: this feature indicates the user's ability to initiate and participate in discussions within the network.

3. Tools and Environment

This section presents the preliminaries from the Twitter perspective, which will be utilized for the implementation of the proposed approach using Twitter API and Libraries [46]. Next, the framework for the two-dimensional evaluation of cryptocurrency popularity is presented. Finally, some background information on Spearman Correlation Coefficient is given as it will be used in the data analysis section.

3.1. Preliminaries

Twitter's Streaming API provides access to the global Twitter feed. The creation of a connection to the Twitter Streaming API is implemented with a long-lasting HTTP request without having to stop the data flow like in Rest API.

Regarding the implementation, a set of Python libraries were utilized, which proved to be particularly useful both in collecting, processing, and displaying data. Several preprocessing steps must be applied in order for the mining methodology of the collected data to be facilitated. The major modules of the proposed methodology are:

- Tweepy: It is a Python library that implements the fetching of the posts; it also permits, with the use of the Twitter interface, the management of the profile of a user, the data collection by considering specific search words, and finally the creation of a batch of posts over a particular time interval. Tweepy is therefore the communication bridge between Python and the Twitter API.
- Textblob: It is a Python library capable of processing data in text format as it provides a simple API for performing natural language processing (NLP) tasks, including sentiment analysis, and, specifically, in our paper, it will be used to calculate the popularity of cryptocurrencies.
- Pandas: It constitutes a Python library that effectively handles high-performance data and provides tools for the analysis of powerful structures. It also utilizes the fast and efficient structure of Dataframes with automatic initialization indexes and offers data alignment along with many options for managing potentially missing data.

3.2. Proposed Approach

In this subsection, two different and related points of interest are presented. On one hand, the users' influence on Twitter can be practically evaluated using Python and Twitter API, whereas, in the second step, we elaborate on methods using the Twitter API search, which simulates a specific metric. This metric is entitled status.reply_count and is considered an additional metric of user popularity.

We aim to create a list of users that can be considered the most influential regarding specific criteria related to financial interest on the Twitter social networking platform. The topic of discussion on which we focus our attention is a cryptocurrency, while the relevant words that users search for are Economy, Bitcoin, Finance, Forex, Ethereum, and others related to cryptocurrencies.

The implementation details are presented below:

1. Search for tweets based on popular hashtags of financial interest (e.g., #Bitcoin, #Finance and #Markets).
2. Collection of users who address the specific tweets in dataframes, named List_of_Users _#X, where X corresponds to the hashtag of our search.
3. Create a common dataframe named List_of_Users_Final, which consists of the union of all the concrete dataframes and contains all the users sorted by the number of their followers; this is the first influence rank.

4. Three more columns to the dataframe are added, where each user receives a ranking number according to three different popularity features. The first criterion is the ratio of retweets per tweet, which expresses both the user's ability to produce quality content and their ability to communicate their tweets to a larger audience. The second criterion constitutes the favorites per tweet ratio, which expresses the percentage of tweets that have a positive response from the user's followers. The third and last is the number of lists to which the user belongs and thus taking into consideration the fact that this metric strengthens the user links through the creation of new networks.
5. These four criteria (followers, retweets, favorites, lists) are simultaneously applied by combining data from all rankings and extracting a list of 30 users who are considered as the Twitter users with the greatest influence on the economy and cryptocurrencies.
6. Having received the id of each user, the next step is to search their timeline with api.user_timeline in order to extract a variable-sized list of users that meets all four of the above popularity criteria.

3.3. Sentiment Score Calculation

The sentiment score of each tweet is calculated using the VADER algorithm, which is a combined approach of lexicon and rule-based sentiment analytic software [47]. VADER is feasible to identify the polarity of text into three categories, which are positive, negative, and neutral. It uses factors like emojis, intensifiers, contraction, punctuation, and acronyms to calculate the scores.

Pre-processing is not essential for VADER as, unlike with some supervised methods of NLP, pre-processing necessities such as tokenisation, stemming, and lemmatisation are not required. The sentiment is determined by the use of plain text. Python provides a library entitled "vaderSentiment" and specifically the "polarityscores" function.

Furthermore, there is no need for pre-processing as VADER implements five major heuristics in terms of sentiment intensity. These include capitalisation, degree modifiers, punctuation, tri-grams analysis as well as the use of "but".

3.4. Correlation Coefficients

The correlation analysis of the sentiment score determined from the tweets with the cryptocurrency popularity plays an important role in prediction. This correlation can quantify the relationship strength associated with the derived sentiment score and popularity. Specifically, the change in opinion of users can later have an impact on the popularity. This marks the importance of the cross-correlation analysis.

The distinction is that cross-correlation introduces a lag, allowing one of the time-series to be shifted left or right to obtain a better correlation. Three statistical correlation methods, namely Spearman, Pearson, and Kendall, were used and compared in the analysis.

To support hypotheses about correlation or not between columns, the Spearman Correlation Coefficient was applied. This factor constitutes a numerical measure, or better indicator, of the size of the correlation between two sets of values. It ranges from -1.00 to $+1.00$ passing through 0.00. A positive sign indicates a positive correlation; this practically means that, when the values of one variable increase, the same happens with those of the second variable. On the contrary, the negative symbol indicates a negative correlation between the two variables; that is, when the values of one variable seem to increase, the values of the second variable decrease. The value 0.00 indicates complete randomness concerning the fluctuations of the two variables we are considering.

Spearman's ρ is the Pearson Correlation Coefficient applied to a set of values after separately sorting the values of both variables, from the smallest to the largest. Calculating the constant correlation of Spearman is a non-parametric process, while the constant evaluates the relationship between two numerical variables without speculating on the real relationship between these two variables. The Spearman constant is calculated from the following equation and expresses the correlation between two tables:

$$\rho = 1 - \frac{6\sum(x_i - y_i)^2}{N^3 - N} \tag{1}$$

where x_i and y_i are the ranks of the variables in a number of observations.

The Pearson Correlation constitutes one of the most widely used correlation approaches as it is for variables with a linear relationship and normal distribution of data. According to [48], Pearson's correlation coefficient (r) is defined as:

$$r = \frac{\sum_{i=1}^{N}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{N}(x_i - \bar{x})^2 \sum(y_i - \bar{y})^2}} \tag{2}$$

where x_i and y_i are the values of features x and y and $\bar{x}, \bar{y}$ denote the mean of x and y for the i-th record in N-ranking tables.

The Kendall Correlation constitutes a non-parametric statistical technique that measures the strength of dependency between two or more variables, similar to Spearman Rank Correlation. Following [49], this coefficient is defined as:

$$\tau = \frac{N_c - N_d}{\frac{N(N-1)}{2}} \tag{3}$$

where N is the sample size, $\frac{N(N-1)}{2}$ are the unique unordered pairs, and N_c, N_d are the number of concordant and discordant, respectively.

4. Results

In this section, we show the outcomes from the evaluation of the proposed approach. For this purpose, three experiments have been conducted. The first one concerns the analysis of the features from a Twitter perspective, the second investigates the association between the involved features, and the last one concerns the sentiment score and the statistical significance of the time-lag per feature for the cryptocurrency popularity prediction.

The first set of experiments produced top-k ranked user lists for four different features, namely followers, retweets, favorites, and lists. In order to estimate whether a correlation between these columns exists, the second set of experiments was applied regarding the aforementioned correlation coefficients. In the following, we present the values of the three above correlation statistical methods. The third experiment employs the Granger causality test to assess the importance of sentiment and each feature, separately, in the cryptocurrency popularity.

4.1. Features Analysis

Twitter's Streaming API, along with the Tweepy library, was used in order to fetch Twitter posts and information for the sentiment analysis. Tweepy constitutes, as previously mentioned, an effective way of retrieving concrete information through permitting information retrieval from Twitter and allowing filtering based on keywords, topics, or hashtags.

The hashtags were derived from the most representative words in the context of each corresponding cryptocurrency. However, this search may incorporate posts that are relevant to other cryptocurrencies as well, and so the selection must be focused in order to comprise particular words that are considered as synonyms to each cryptocurrency. For example, regarding #Bitcoin, the synonyms are #BTC and #Bitcoinprice. The posts we collected were published for a time period of 12 days (from 6 April to 18 April 2020), and in the following Table 1, the relevant hashtags are displayed.

Table 1. Example Hashtags.

Bitcoin	Ethereum	Litecoin	Stellar
BTC Bitcoinprice	ETH Ethereumprice	LTC Litecoinprice	XLM Stellarprice

In Table 2, the number of posts of the four most well-known cryptocurrencies, i.e., Bitcoin, Ethereum, Litecoin, and Stellar, are compared as the timelines of the 500 most influential users were taken into consideration. The results of this table are not associated with the actual prices of these cryptocurrencies but only with the profile of the users. The number of posts regarding Ethereum and Bitcoin is much larger than the others, followed by those of Litecoin and Stellar, respectively.

Table 2. Number of posts per cryptocurrency.

Bitcoin	4700
Ethereum	5200
Litecoin	1700
Stellar	400
Total Posts	12,000

Specifically, in Table 3, we notice that, especially when considering a small number of users, overlap can be identified. More to the point, user #Forbes is observed among the top users in two features, namely followers as well as lists. The same stands for other users that exist in the lists of two different features.

Table 3. Top-4 ranked users for different features.

Followers	Retweets	Favorites	Lists
Forbes	ShashiTharoor	AVFCOfficial	Forbes
detikcom	maggieNYT	maggieNYT	Milenio
ShashiTharoor	gtconway3d	gtconway3d	CNBC
Milenio	Rewards4Justice	Rewards4Justice	maggieNYT

4.2. Correlation Analysis

In the second set of experiments, we observe in Table 4 that the feature of lists has a strong degree of correlation with the feature of followers, that is, the highest value is equal to 0.795. Similarly, the feature of favorites is also strongly associated with the feature of retweets, with the value of 0.618 being the second highest degree of correlation. Finally, the feature of followers appears to have a weak correlation with the feature of retweets and the same stands for Lists with Favorites.

The relationship of the user influence is found to be positive using a Pearson statistical method. Respectively, Tables 5 and 6 assess the relatedness among the same features by employing Pearson's and Kendall's coefficients. Similar behavior is verified by both of these correlation methods, although the values of the latter are slightly different (with either lower or higher reduction).

Table 4. Spearman's Correlation between Influence Ranks.

	Followers	Retweets	Favorites	Lists
Followers	1.000	0.279	0.186	0.795
Retweets	0.279	1.000	0.618	0.242
Favorites	0.186	0.618	1.000	0.124
Lists	0.795	0.242	0.124	1.000

Table 5. Pearson's Correlation between Influence Ranks.

	Followers	Retweets	Favorites	Lists
Followers	1.000	0.258	0.163	0.743
Retweets	0.258	1.000	0.598	0.212
Favorites	0.163	0.598	1.000	0.104
Lists	0.743	0.212	0.104	1.000

Table 6. Kendall's Correlation between Influence Ranks.

	Followers	Retweets	Favorites	Lists
Followers	1.000	0.178	0.123	0.713
Retweets	0.178	1.000	0.508	0.142
Favorites	0.123	0.508	1.000	0.100
Lists	0.713	0.142	0.100	1.000

4.3. Sentiment Analysis Results

Furthermore, another experiment concerns the sentiment scores of posts per cryptocurrency, where we will focus only on Bitcoin and Ethereum in Figure 1. Sentiment analysis relies on a dictionary which has lexical features corresponding with emotion values, which constitute the sentiment scores. The sentiment score of a tweet can be acquired by summing up the sentiment score of each word in it.

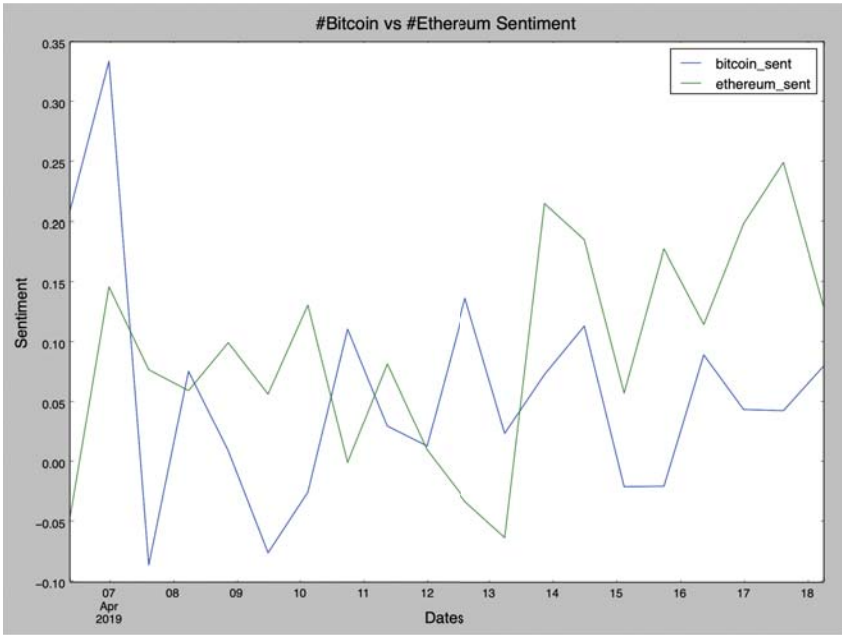


Figure 1. Sentiment scores of posts per cryptocurrency.

We tried to draw conclusions about their popularity not by counting the overall number of tweets but by measuring the sentiment of the tweets regarding these two cryptocurrencies. We can not identify any important notions regarding the price of each cryptocurrency; nevertheless, fluctuations of tweets sentiment by observing the timelines of the most influential user groups can be illustrated.

4.4. Features Statistical Significance on Cryptocurrency Popularity Prediction

To assess cryptocurrency popularity, we based our proposed methodology on (a) user influence and (b) sentiment scores. Sentiment analysis is often combined with a (Granger-) causality test and/or regression model(s) [28]. Initially, we evaluate the statistical significance (p-value) of time lag in both components of popularity using the Granger causality test. The null hypothesis ($H0$) mentions that sentiment/followers/retweets/favorites/lists time series do not (Granger) cause cryptocurrency popularity time series.To reject the null hypothesis, it can be shown that sentiment/followers/retweets/favorites/lists values provide statistically significant information about future values of popularity.

Cryptocurrency popularity is treated as a multivariate autoregressive model of order p. Let us consider the variables CP_t, S_t, F_t, R_t, FV_t, and L_t that represent the popularity, sentiment, followers, retweets, favorites, and lists time series data, respectively. In the last experiment, we evaluate five cases where each column of Tables 7–10 corresponds to one of the following cases for all four cryptocurrencies, respectively:

- Case 1: Forecast CP_{t+1} based on past values CP_t, S_t.
- Case 2: Forecast CP_{t+1} based on past values CP_t, F_t.
- Case 3: Forecast CP_{t+1} based on past values CP_t, R_t.
- Case 4: Forecast CP_{t+1} based on past values CP_t, FV_t.
- Case 5: Forecast CP_{t+1} based on past values CP_t, L_t.

Table 7. Statistical significance (p-values) of bivariate Granger causality correlation for Bitcoin popularity based on tweet sentiment, followers, retweets, favorites, and lists.

Time Lag	Sentiment	Followers	Retweets	Favorites	Lists
1 day	0.0468	0.0025	0.0110	0.0016	0.0075
2 day	0.0318	0.0018	0.0121	0.0178	0.0084
3 day	0.0420	0.0011	0.0207	0.0142	0.0096
4 day	0.0251	0.0253	0.0251	0.0163	0.0135
5 day	0.0365	0.0276	0.0281	0.0137	0.0174
6 day	0.0253	0.0356	0.0314	0.0144	0.0286
7 day	0.0271	0.0182	0.0481	0.0169	0.0275
8 day	0.0169	0.0443	0.0421	0.0288	0.0197
9 day	0.0214	0.0422	0.0372	0.0184	0.0106
10 day	0.0375	0.0325	0.0351	0.0136	0.0195
11 day	0.0349	0.0249	0.0308	0.0107	0.0183
12 day	0.0435	0.0338	0.0204	0.0148	0.0116

Table 8. Statistical significance (p-values) of bivariate Granger causality correlation for Ethereum popularity based on tweet sentiment, followers, retweets, favorites, and lists.

Time Lag	Sentiment	Followers	Retweets	Favorites	Lists
1 day	0.0442	0.0035	0.0210	0.0086	0.0225
2 day	0.0378	0.0028	0.0121	0.0122	0.0171
3 day	0.0412	0.0010	0.0107	0.0032	0.0492
4 day	0.0256	0.0183	0.0201	0.0063	0.0205
5 day	0.0183	0.0236	0.0261	0.0076	0.0017
6 day	0.0153	0.0276	0.0334	0.0114	0.0006
7 day	0.0161	0.0172	0.0421	0.0192	0.0009
8 day	0.0109	0.0434	0.0499	0.0228	0.0019
9 day	0.0114	0.0322	0.0472	0.0284	0.0056
10 day	0.0275	0.0365	0.0431	0.0216	0.0115
11 day	0.0249	0.0219	0.0368	0.0347	0.0133
12 day	0.0335	0.0108	0.0304	0.0318	0.0076

Table 9. Statistical significance (p-values) of bivariate Granger causality correlation for Litecoin popularity based on tweet sentiment, followers, retweets, favorites, and lists.

Time Lag	Sentiment	Followers	Retweets	Favorites	Lists
1 day	0.0355	0.0102	0.0278	0.0132	0.0256
2 day	0.0218	0.0078	0.0167	0.0255	0.0223
3 day	0.0389	0.0125	0.0087	0.0103	0.0385
4 day	0.0318	0.0223	0.0155	0.0097	0.0165
5 day	0.0278	0.0253	0.0335	0.0102	0.0123
6 day	0.0245	0.0355	0.0123	0.0086	0.0095
7 day	0.0289	0.0168	0.0324	0.0238	0.0045
8 day	0.0221	0.0387	0.0442	0.0128	0.0078
9 day	0.0145	0.0267	0.0492	0.0397	0.0032
10 day	0.0298	0.0354	0.0412	0.0129	0.0097
11 day	0.0175	0.0179	0.0218	0.0327	0.0163
12 day	0.0374	0.0332	0.0244	0.0123	0.0054

Table 10. Statistical significance (p-values) of bivariate Granger causality correlation for Stellar popularity based on tweet sentiment, followers, retweets, favorites, and lists.

Time Lag	Sentiment	Followers	Retweets	Favorites	Lists
1 day	0.0145	0.0188	0.0235	0.0055	0.0099
2 day	0.0338	0.0182	0.0216	0.0143	0.0054
3 day	0.0367	0.0128	0.0305	0.0211	0.0156
4 day	0.0212	0.0357	0.0171	0.0235	0.0259
5 day	0.0398	0.0212	0.0249	0.0167	0.0128
6 day	0.0318	0.0364	0.0358	0.0223	0.0241
7 day	0.0411	0.0272	0.0495	0.0291	0.0376
8 day	0.0219	0.0413	0.0476	0.0328	0.0377
9 day	0.0345	0.0486	0.0416	0.0217	0.0256
10 day	0.0415	0.0387	0.0398	0.0177	0.0214
11 day	0.0408	0.0315	0.0387	0.0247	0.0288
12 day	0.0449	0.0418	0.0344	0.0119	0.0155

We avoid combining the following pair of variables F_t, R_t, FV_t and L_t due to their dependency, as Spearman's correlation coefficient shows. The popularity model considers the lagged values of both CP_t and the time series of the rest features separately for various time lags. Specifically, this test aims to determine the significance of the association between time lag and each tweet sentiment (positive or negative). The same process is repeated independently for the time series that captures user influence, namely, followers, retweets, favorites, and lists in relation to the CP_t time series. This test was performed on a short time period of 12 days (from 6 April to 18 April 2020) to identify relations that are statistically significant (p < 0.05).

Observing Tables 7–10, for Bitcoin, Ethereum, Litecoin, and Stellar cryptocurrencies, we conclude the predictive power of all variables on the popularity variable, as the p-values are all well below the 0.05 level. Hence, we reject the null hypothesis, and, as a result, the current data are stationary. The lower the p-value ($p < 0.05$) of the variables, the higher their predictive power on cryptocurrencies' popularity is. Finally, the daily changes in Twitter metrics-variables could forecast a similar rise or fall in cryptocurrency popularity and its fluctuations in advance.

5. Conclusions and Future Work

In this paper, we focused on comparing the popularity of four popular cryptocurrencies based on two different results. Initially, the number of tweets for a concrete time period was measured, and, in the following, the classification of these tweets as positive or negative was implemented. Furthermore, the Granger causality analysis is applied to identify the proper time lag and the most influential variable to predict future values of the cryptocurrency popularity considering the past. In addition, given that previous correlation analysis only indicates the relationship between features (either positive or negative), it could be used in conjunction with convergent cross-mapping (CCM) [50] to determine the direction and magnitude of the causality (as illustrated in Figure 4 of Sugihara et al. [51]), also studying the problem under noisy conditions.

For future work, the analysis could be improved by employing a domain-specific lexicon, as the latter can improve the classifier performance and the prediction accuracy by identifying corresponding cryptocurrency, economy and financial terms; thus, a more representative sentiment can be yielded [28]. Moreover, Bitcoin price can be treated as a time-series problem where the price index can be forecasted with the use of machine learning techniques, like Recurrent Neural Network (RNN), Long Short Term Memory (LSTM), or ARIMA model [52–54]. The proposed method of predicting fluctuations in the price and trading volume of cryptocurrencies based on user comments and replies in online communities is likely to increase the understanding and availability of cryptocurrencies if a range of improvements and applications are implemented. Finally, different approaches to user comments and replies in online communities are expected to bring more significant results in diverse fields.

Author Contributions: M.T., A.K., E.D., G.V. and P.M. conceived of the idea, designed and performed the experiments, analyzed the results, drafted the initial manuscript, and revised the final manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Java, A.; Song, X.; Finin, T.; Tseng, B. Why We Twitter: Understanding Microblogging Usage and Communities. In Proceedings of the 9th WebKDD and 1st SNA-KDD Workshop on Web Mining and Social Network Analysis (WebKDD/SNA-KDD), San Jose, CA, USA, 12 August 2007; pp. 56–65.
2. Nakamoto, S. Bitcoin: A Peer-to-Peer Electronic Cash System. *Decentralized Bus. Rev.* **2008**, 21260. Available online: https://bitcoin.org/en/bitcoin-paper (accessed on 19 May 2022).
3. Reid, F.; Harrigan, M. An Analysis of Anonymity in the Bitcoin System. In Proceedings of the IEEE Third International Conference on Privacy, Security, Risk and Trust (PASSAT)/IEEE Third International Conference on Social Computing (SocialCom), Boston, MA, USA, 9–11 October 2011; pp. 1318–1326.
4. Böhme, R.; Christin, N.; Edelman, B.; Moore, T. Bitcoin: Economics, Technology, and Governance. *J. Econ. Perspect.* **2015**, *29*, 213–238. [CrossRef]
5. Grinberg, R. Bitcoin: An Innovative Alternative Digital Currency. *Hastings Sci. Technol. Law J.* **2012**, *4*, 159.
6. Ahamad, S.; Nair, M.; Varghese, B. A survey on crypto currencies. In Proceedings of the 4th International Conference on Advances in Computer Science (AETACS), Delhi, India, 13–14 December 2013; pp. 42–48.
7. Kondor, D.; Pósfai, M.; Csabai, I.; Vattay, G. Do the Rich Get Richer? An Empirical Analysis of the Bitcoin Transaction Network. *PLoS ONE* **2014**, *9*, e86197. [CrossRef] [PubMed]
8. Ron, D.; Shamir, A. Quantitative Analysis of the Full Bitcoin Transaction Graph. In Proceedings of the 17th International Conference on Financial Cryptography and Data Security, Okinawa, Japan, 1–5 April 2013; Volume 7859, pp. 6–24.
9. Franco, P. *Understanding Bitcoin: Cryptography, Engineering and Economics*; John Wiley & Sons: Hoboken, NJ, USA, 2014.
10. Bornholdt, S.; Sneppen, K. Do Bitcoins Make the World Go Round? On the Dynamics of Competing Crypto-currencies. *arXiv* **2014**, arXiv:abs/1403.6378.
11. Agarwal, A.; Xie, B.; Vovsha, I.; Rambow, O.; Passonneau, R. Sentiment Analysis of Twitter Data. In Proceedings of the Workshop on Language in Social Media (LSM), Portland, OR, USA, 23 June 2011; pp. 30–38.
12. Baltas, A.; Kanavos, A.; Tsakalidis, A. An Apache Spark Implementation for Sentiment Analysis on Twitter Data. In Proceedings of the International Workshop on Algorithmic Aspects of Cloud Computing (ALGOCLOUD), Aarhus, Denmark, 22 August 2016; pp. 15–25.

13. Kanavos, A.; Nodarakis, N.; Sioutas, S.; Tsakalidis, A.; Tsolis, D.; Tzimas, G. Large Scale Implementations for Twitter Sentiment Classification. *Algorithms* **2017**, *10*, 33. [CrossRef]
14. Kanavos, A.; Perikos, I.; Hatzilygeroudis, I.; Tsakalidis, A. Emotional Community Detection in Social Networks. *Comput. Electr. Eng.* **2018**, *65*, 449–460. [CrossRef]
15. Al-Yahyaee, K.H.; Mensi, W.; Yoon, S.M. Efficiency, Multifractality, and the Long-memory Property of the Bitcoin Market: A Comparative Analysis with Stock, Currency, and Gold Markets. *Financ. Res. Lett.* **2018**, *27*, 228–234. [CrossRef]
16. Bariviera, A.F. The Inefficiency of Bitcoin Revisited: A Dynamic Approach. *Econ. Lett.* **2017**, *161*, 1–4. [CrossRef]
17. Jiang, Y.; Nie, H.; Ruan, W. Time-varying Long-term Memory in Bitcoin Market. *Financ. Res. Lett.* **2018**, *25*, 280–284. [CrossRef]
18. Balcilar, M.; Bouri, E.; Gupta, R.; Roubaud, D. Can Volume Predict Bitcoin Returns and Volatility? A Quantiles-based Approach. *Econ. Model.* **2017**, *64*, 74–81. [CrossRef]
19. Cagli, E.C. Explosive Behavior in the Prices of Bitcoin and Altcoins. *Financ. Res. Lett.* **2019**, *29*, 398–403. [CrossRef]
20. Brandvold, M.; Molnár, P.; Vagstad, K.; Valstad, O.C.A. Price Discovery on Bitcoin Exchanges. *J. Int. Financ. Mark. Inst. Money* **2015**, *36*, 18–35. [CrossRef]
21. Ciaian, P.; Rajcaniova, M.; d'Artis, K. The Economics of BitCoin Price Formation. *Appl. Econ.* **2016**, *48*, 1799–1815. [CrossRef]
22. da Gama Silva, P.V.J.; Klotzle, M.C.; Pinto, A.C.F.; Gomes, L.L. Herding Behavior and Contagion in the Cryptocurrency Market. *J. Behav. Exp. Financ.* **2019**, *22*, 41–50. [CrossRef]
23. Vidal-Tomás, D.; Ibáñez, A.M.; Farinós, J.E. Herding in the Cryptocurrency Market: CSSD and CSAD Approaches. *Financ. Res. Lett.* **2019**, *30*, 181–186. [CrossRef]
24. Fauzi, M.A.; Paiman, N.; Othman, Z. Bitcoin and Cryptocurrency: Challenges, Opportunities and Future Works. *J. Asian Financ. Econ. Bus. (JAFEB)* **2020**, *7*, 695–704. [CrossRef]
25. Mukhopadhyay, U.; Skjellum, A.; Hambolu, O.; Oakley, J.; Yu, L.; Brooks, R.R. A Brief Survey of Cryptocurrency Systems. In Proceedings of the IEEE 14th Annual Conference on Privacy, Security and Trust (PST), Auckland, New Zealand, 12–14 December 2016; pp. 745–752.
26. Hong, Y. How the Discussion on a Contested Technology in Twitter Changes: Semantic Network Analysis of Tweets about Cryptocurrency and Blockchain Technology. In Proceedings of the 22nd Biennial Conference of the International Telecommunications Society (ITS), Seoul, Korea, 24–27 June 2018.
27. Dritsas, E.; Livieris, I.E.; Giotopoulos, K.; Theodorakopoulos, L. An apache spark implementation for graph-based hashtag sentiment classification on twitter. In Proceedings of the 22nd Pan-Hellenic Conference on Informatics, Athens, Greece, 29 November–1 December 2018; pp. 255–260.
28. Kraaijeveld, O.; Smedt, J.D. The predictive power of public Twitter sentiment for forecasting cryptocurrency prices. *J. Int. Financ. Mark. Inst. Money* **2020**, *65*, 101188. [CrossRef]
29. Stenqvist, E.; Lönnö, J. *Predicting Bitcoin Price Fluctuation with Twitter Sentiment Analysis*; KTH Royal Institute of Technology, School of Computer Science and Communication: Stockholm, Sweden, 2017.
30. Colianni, S.; Rosales, S.; Signorotti, M. *Algorithmic Trading of Cryptocurrency Based on Twitter Sentiment Analysis*; CS229 Project. 2015, pp. 1–5. Available online: https://www.semanticscholar.org/paper/Algorithmic-Trading-of-Cryptocurrency-Based-on-Colianni-Rosales/9b838a3177523b8179511283b9489caa0f69910d (accessed on 20 March 2022).
31. Aste, T. Cryptocurrency Market Structure: Connecting Emotions and Economics. *Digit. Financ.* **2019**, *1*, 5–21. [CrossRef]
32. Abraham, J.; Higdon, D.; Nelson, J.; Ibarra, J. Cryptocurrency Price Prediction Using Tweet Volumes and Sentiment Analysis. *SMU Data Sci. Rev.* **2018**, *1*, 1.
33. Albrecht, S.; Lutz, B.; Neumann, D. The Behavior of Blockchain Ventures on Twitter as a Determinant for Funding Success. *Electron. Mark.* **2020**, *30*, 241–257. [CrossRef]
34. Karalevicius, V.; Degrande, N.; Weerdt, J.D. Using Sentiment Analysis to Predict Interday Bitcoin Price Movements. *J. Risk Financ.* **2018**, *19*, 56–75. [CrossRef]
35. Alghobiri, M. Using Data Mining Algorithm for Sentiment Analysis of Users' Opinions about Bitcoin Cryptocurrency. *J. Theor. Appl. Inf. Technol.* **2019**, *97*, 2195–2205.
36. Naeem, M.A.; Mbarki, I.; Suleman, M.T.; Vo, X.V.; Shahzad, S.J.H. Does Twitter Happiness Sentiment Predict Cryptocurrency? *Int. Rev. Financ.* **2020**, *21*, 1529–1538. [CrossRef]
37. Li, T.R.; Chamrajnagar, A.S.; Fong, X.R.; Rizik, N.R.; Fu, F. Sentiment-Based Prediction of Alternative Cryptocurrency Price Fluctuations Using Gradient Boosting Tree Model. *Front. Phys.* **2019**, *7*, 98. [CrossRef]
38. Vallet, D.; Fernández, M.; Castells, P.; Mylonas, P.; Avrithis, Y. A Contextual Personalization Approach Based on Ontological Knowledge. In Proceedings of the 2nd International Workshop on Contexts and Ontologies: Theory, Practice and Applications (C&O-2006) Collocated with the 17th European Conference on Artificial Intelligence (ECAI-2006), Riva del Garda, Italy, 28 August 2006; Volume 210.
39. Drakopoulos, G.; Kanavos, A.; Mylonas, P.; Sioutas, S. Defining and evaluating Twitter influence metrics: A higher-order approach in Neo4j. *Soc. Netw. Anal. Min.* **2017**, *7*, 52:1–52:14. [CrossRef]
40. Kafeza, E.; Kanavos, A.; Makris, C.; Vikatos, P. T-PICE: Twitter Personality Based Influential Communities Extraction System. In Proceedings of the 2014 IEEE International Congress on Big Data, Anchorage, AK, USA, 27 June–2 July 2014; pp. 212–219.
41. Badashian, A.S.; Stroulia, E. Measuring User Influence in Github: The Million Follower Fallacy. In Proceedings of the 3rd International Workshop on CrowdSourcing in Software Engineering (CSI-SE@ICSE), Austin, TX, USA, 16 May 2016; pp. 15–21.

42. Kanavos, A.; Livieris, I.E. Fuzzy Information Diffusion in Twitter by Considering User's Influence. *Int. J. Artif. Intell. Tools* **2020**, *29*, 2040003:1–2040003:22. [CrossRef]
43. Kim, Y.B.; Kim, J.G.; Kim, W.; Im, J.H.; Kim, T.H.; Kang, S.J.; Kim, C.H. Predicting Fluctuations in Cryptocurrency Transactions Based on User Comments and Replies. *PLoS ONE* **2016**, *11*, e0161197. [CrossRef]
44. Cialdini, R.B. *Influence: Science and Practice*; Pearson Education: Boston, MA, USA, 2009; Volume 4.
45. Cha, M.; Haddadi, H.; Benevenuto, F.; Gummadi, P.K. Measuring User Influence in Twitter: The Million Follower Fallacy. In Proceedings of the 4th International Conference on Weblogs and Social Media (ICWSM), Washington, DC, USA, 23–26 May 2010; The AAAI Press: Palo Alto, CA, USA, 2010.
46. Dritsas, E.; Vonitsanos, G.; Livieris, I.E.; Kanavos, A.; Ilias, A.; Makris, C.; Tsakalidis, A.K. Pre-processing Framework for Twitter Sentiment Classification. In Proceedings of the 15th IFIP International Conference on Artificial Intelligence Applications and Innovations (AIAI), Hersonissos, Crete, Greece, 24–26 May 2019; Volume 560, pp. 138–149.
47. Hutto, C.J.; Gilbert, E. VADER: A Parsimonious Rule-Based Model for Sentiment Analysis of Social Media Text. In Proceedings of the 8th International Conference on Weblogs and Social Media (ICWSM), Ann Arbor, MI, USA, 1–4 June 2014; The AAAI Press: Palo Alto, CA, USA, 2014.
48. Yang, B.; Sun, Y.; Wang, S. A Novel Two-stage Approach for Cryptocurrency Analysis. *Int. Rev. Financ. Anal.* **2020**, *72*, 101567. [CrossRef]
49. Puth, M.T.; Neuhäuser, M.; Ruxton, G.D. Effective Use of Spearman's and Kendall's Correlation Coefficients for Association between Two Measured Traits. *Anim. Behav.* **2015**, *102*, 77–84. [CrossRef]
50. Tu, C.; Fan, Y.; Fan, J. Universal Cointegration and Its Applications. *iScience* **2019**, *19*, 986–995. [CrossRef] [PubMed]
51. Sugihara, G.; May, R.; Ye, H.; hao Hsieh, C.; Deyle, E.; Fogarty, M.; Munch, S. Detecting Causality in Complex Ecosystems. *Science* **2012**, *338*, 496–500. [CrossRef] [PubMed]
52. Alessandretti, L.; ElBahrawy, A.; Aiello, L.M.; Baronchelli, A. Anticipating Cryptocurrency Prices using Machine Learning. *Complexity* **2018**, *2018*, 8983590:1–8983590:16. [CrossRef]
53. Madan, I.; Saluja, S.; Zhao, A. *Automated Bitcoin Trading via Machine Learning Algorithms*; Stanford University: Stanford, CA, USA, 2015.
54. McNally, S.; Roche, J.; Caton, S. Predicting the Price of Bitcoin using Machine Learning. In Proceedings of the 26th Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP), Cambridge, UK, 21–23 March 2018; pp. 339–343.

big data and cognitive computing

Article

Social Media Analytics as a Tool for Cultural Spaces—The Case of Twitter Trending Topics

Vassilis Poulopoulos [1,*,†] and Manolis Wallace [1,†]

[1] Knowledge and Uncertainty Research Laboratory, University of the Peloponnese, 221 31 Tripolis, Greece; wallace@uop.gr

* Correspondence: vacilos@uop.gr

† These authors contributed equally to this work.

Abstract: We are entering an era in which online personalities and personas will grow faster and faster. People are tending to use the Internet, and social media especially, more frequently and for a wider variety of purposes. In parallel, a number of cultural spaces have already decided to invest in marketing and message spreading through the web and the media. Growing their audience, or locating the appropriate group of people to share their information, remains a tedious task within the chaotic environment of the Internet. The investment is mainly financial—usually large—and directed to advertisements. Still, there is much space for research and investment in analytics that can provide evidence considering the spreading of the word and finding groups of people interested in specific information or trending topics and influencers. In this paper, we present a part of a national project that aims to perform an analysis of Twitter's trending topics. The main scope of the analysis is to provide a basic ordering on the topics based on their "importance". Based on this, we clarify how cultural institutions can benefit from such an analysis in order to empower their online presence.

Keywords: social media analysis; cultural spaces; cultural informatics; trending topics

Citation: Poulopoulos, V.; Wallace, M. Social Media Analytics as a Tool for Cultural Spaces—The Case of Twitter Trending Topics. *Big Data Cogn. Comput.* **2022**, *6*, 63. https://doi.org/10.3390/bdcc6020063

Academic Editors: Konstantinos Kotis and Dimitris Spiliotopoulos

Received: 7 April 2022
Accepted: 26 May 2022
Published: 2 June 2022

1. Introduction

Nowadays, many people use online services for business or for leisure. This phenomenon has grown to the point of being a huge part of everyday life for many people around the globe, especially as a result of the COVID-19 pandemic's lockdowns. On these grounds, people have started to be concerned about their online profiles, having even created personas. The Internet and its services became a main stream of socialization at some points, as people were unable to physically meet in person. Social media platforms, which have already gained a lot of attention, became the main communication channels, and teleconferencing services replaced daily meetings.

Under these strange circumstances, many businesses changed (and keep changing) their business models, starting to empower their online presence; some of them pivoted to online only. A large number of people around the globe are working remotely, and more and more companies are announcing to their employees that they will continue with the "remote work" model for the coming months at least. However, a number of companies, including culture-related spaces, have a business model that is based on the presence of their audience (visitors). Cultural organizations are in the strange situation of their "customers" desiring the physical presence of culturally significant objects (say, in order to enjoy a piece of art), despite the fact that it is prohibited to reach or even approach the cultural objects.

A large number of culture-related organizations have already turned to the Internet as a space for message spreading and audience acquisition. They have also invested in cutting-edge technology in order to offer unique experiences to visitors. Technology has also helped to perform interdisciplinary research in the field of culture, giving birth to museum and cultural informatics. The EU has already made a very large investment in culture and cultural heritage, by providing funds for research projects, having as the apogee the announcement of 2018 as the year of cultural heritage. It is interesting that an important number of the funded projects are related to information technology, either related to cultural heritage preservation or museum informatics. The main axes of technology in culture are related to digitization, virtual reality, preservation, semantic analysis, crowdsourcing, networking and IoT (Internet of Things). This is proof that cutting edge technologies are widely used in modern museums, enabling maximization of the visitors' stimulation. Modern cultural spaces are trying to take advantage of innovation and new techniques and technologies. Still, a large number of them remain attached to classical research and methodologies; without calling this the wrong side, concerns about the world changing must be taken into consideration seriously.

The pandemic has slightly changed the point of view for a large number of businesses, including cultural spaces in general. They understood that there is a strong need to support alternative ways of spreading their messages and attracting people. Moreover, physical presence is now considered "risky" in several places, leading to solutions that include technology and "remote access". In parallel, during the lockdowns of the COVID-19 pandemic, people realized that culture is part of their everyday life. A number of cultural spaces in Greece started providing open access to culture-related content. The Greek National Opera announced a number of online performances during April 2020 (https://www.nationalopera.gr/en/news-features/item/3117-ministry-of-culture-and-sports-the-gno-free-broadcasts-continue-with-great-success (accessed on 30 May 2021)). The content was presented live, using platforms such as YouTube and Facebook, and was available for free access some hours after it finished its live broadcast. On other occasions, theatrical shows were broadcast online at a voluntary price. The aforementioned were organized within a small time frame from the moment the quarantine was announced, meaning that the technological infrastructure and readiness are at a very high level. The majority of online performances from famous cultural organizations reached sufficient levels of spectators (online users), but still, the information reached the people that were, and are, searching for plays regardless of the lockdown due to the pandemic. The possibility of reaching a larger audience, especially at a time at which there seems to be a paradigm shift regarding the engagement of people towards arts and culture, remains at low levels. The readiness level in alternative presentations is very high, but still, there are very low expectations regarding the audience that is reached and is convinced by the modern communication channel.

Efforts towards discovering the audience of a business in general have been the subject of research ever since the Internet became popular. Once people were using the Internet for socializing, a wide variety of businesses started turning to deep marketing analysis on the medium. The cultural organizations were late to adopt such technologies and techniques. However, even today, with the extensive usage of technology in our everyday lives, so much so that in a sense we are considered as online or offline in our lives, cultural spaces have not yet adopted the kinds of technology that will help them reach broader audiences and provide information about their content and messages.

In this paper, we present how social media analytics, as part of the Greek National Project "PaloAnalytics", can be helpful for cultural spaces. We focus on the analysis of Twitter's trending topics and relate its outcomes to cultural spaces in order to prove how they can benefit from the simple correlation of their content with the analysis that can be performed to social media. "PaloAnalyics" is a nationally funded project that intends to focus on the basic challenges that organizations face and is related to the implementation of a universal monitoring tool with links and interconnections between collected data. It is analyzed by a number of different researchers in various languages. The project as a whole can be a useful tool for large companies, but select parts of it, implemented within the scope of the project, can be beneficial for culture-related organizations. On these grounds, we selected a module that can be used as a stand-alone tool and presented how it can be used so as help organizations put the focus of their online efforts into specific actions and posts, in order to achieve higher penetration to the medium and better public awareness.

The rest of the paper is organized as follows. Section 2 presents research performed in the field of cultural informatics and social media analysis. The next section gives detailed evidence on the algorithmic approach of the proposed solution. Section 4 presents the modules that were implemented from a technical perspective and the flow of information within the system. Experimental results are presented in Section 5, and the paper concludes with Section 6 that presents the discussion and future work.

2. Related Work

From a research perspective, the combination of technology and culture was initiated from what is called "museum informatics". The research in this field started more than 40 years ago, with Bearman being a pioneer trying to empower the use of technology, at least in relation to archives and internal organizations ([1–3]). Initial efforts dealt with databases and cultural object registration and internal organization of cultural spaces and definition of prototypes for the classification of objects, mainly in museums ([4,5]). However, as technology evolved, more and more "technological advances" were found in cultural venues. Digitization and digital collections ([6–10]), visitor participation ([11–13]), ByoD ([11,14,15]), or AR and VR ([16–21]) as part of a museum's procedure are only some of the examples of early or later adoptions. Nowadays, the interdisciplinary research that is performed in culture and technology intends to bring together cultural spaces with the latest advances in technology. P.F. Marty has performed extensive research in the area, covering all the bases that lead to the aforementioned and will formulate the future of culture in relation to technology ([22–24]).

Talking of which—the future of culture in relation to technology—we should note that it seems that people tend to have a mixed type of life including a lot of digital alongside with the "analog" part. In this sense, extensive research is performed considering the role of museums and cultural spaces in the online world. More specifically, we examine how the problem of a museum's presence in the world of social media can be tackled.

Research on behaviors and interactions is common, usually in relation to social media. Research on Instagram accounts related to art was performed and it seems that "likes and comments are greatly influenced by interactions with confusion and curiosity being a big reason to engage" [25]. This reveals that "active participation" plays an important role in social media presence. It is, on the other hand, interesting to examine all the issues from several different angles when dealing with "data-driven" arts, as there is always the possibility of "opportunities or chimeras" [26]. In this work, the authors conclude that in an era that is filled with a plethora of data, following a data analytics approach would benefit ACOs (Arts and Culture Organizations). Research analysing the situation in relation to Greek museums proved that, despite the fact that a main stream of the museum message is based on images of the exhibits, only a few museums use Instagram. At the same time, a large number of users tag museums and cultural objects, without any kind of interaction from the cultural spaces [27]. Many concerns regarding the stance of the museums and cultural spaces were raised during the COVID-19 pandemic. Democratization of the procedures of the museums is also a novel approach in the world of the web [28]. In this research, the authors explore the unfulfilled potential for democratizing museums by exploring aspects of Instagram (Instagram—https://www.instagram.com/ (accessed on 30 May 2021). Their findings show that museums are using the medium in a way that is not attractive, as they keep a more "traditional promotional" stance or "an authoritative knowledge-telling attitude". The aforementioned mean that the museum has to become participatory and inclusive in alternative ways.

Inspiration among visitors is a factor that museums are really interested in, and research was performed on this issue with regards to data analysis on Twitter [29]. Within the aforementioned research, we implemented prototype tools to collect information from the medium, in order to find "expressions of inspiration in Tweets". From the findings, there is an indication that "social media may have some potential as a source of valuable information for museums, though this depends heavily upon how annotation exercises are conducted".

A literature review on the usage of social media by museums is presented in [30]. The review leads to the assumptions that the museum communication can benefit with the use of social media and that the educational role of the museum can be enhanced. Furthermore, it is stated that "beyond social media effectiveness, museums managers lag into dialogical communication". This proves that there is a strong need for tools and methods to be provided to museums and cultural spaces, in order to have a differentiated view of the data in social media. Digging more for specific use cases in museums, it seems to be complex for small organizations to have a decent presence in social media, but another interesting aspect of the usage of social media in museums can be revealed [31]. It seems that "evidence that the museum staff acts as 'social media champions' represents a qualitative indicator of an increase in the employees' commitment and loyalty to the organization". This reveals another part of the online presence of the museums in regards to the engagement of the employees. The interactive and participatory presence of museums in social media gives the opportunity to the employees to be a part of the communication strategy, which is not possible otherwise.

Social media analysis is not something new. A study back in 2012 tried to present the uses and evaluations of social media in American museums [32]. At that time (10 years ago), the results "indicate that American museums believe becoming involved with social media is important, but they are not using the sites at high levels of dialogic engagement". The latter reasoning remains a problem for cultural spaces till today. It takes huge effort to decide on which part of social media to "invest time" into so as to maximize the performance on the respective platforms. In this case, we are trying to help museums focus on specific topics that are already trending alongside the medium. Another case, the one of the Côa Valley Museum and Archaeological Park, is presented in [33]. From the analysis, it seems that as long as museums stand for "guardians of human memories", the author believes that "museums have an important role to play in providing widespread access to their collections, facilitating scientific research and fostering its use for educational, leisure or recreation". The case of ephemeral storytelling with social media was researched in [34]. This research indicates that the usage of the "stories" feature of specific social media are a means to engage a larger audience, and they conclude that "museums should adapt their policies and programs to current social media communication behaviors to remain relevant and be a part of what and how people share their lives". It is obvious that the culture-related spaces need to be aligned with contemporary trends. More recent research on the part that is related to nowadays tech-savvy audience reveals that "Tech-savvy tourists will enjoy and appreciate the overall digital experience, thus identifying themselves with and feeling part of the museum, becoming loyal, and willingly providing economic support" [35]. In fact, it is obvious from a number of research outcomes that at the end of the day, museums have sufficient online communication, but there seems to be zero interaction [36]. From a number of researches regarding what is expected from the museum in its online presence today, the results remain the same: the museum has to interact and participate.

The aforementioned research proves that the community that researches cultural informatics is aware of the power of social media and the stance of people towards technology. We live in an era where people tend to be more and more digital (or online). In this light, the use of social media seems to be a one-way road for the cultural spaces.

As a matter of fact, using social media in general and having a decent profile means three things:

- Providing detailed information;
- Performing information dissemination;
- Participating.

While the first two things necessary in order to maintain a decent profile are easy to understand, the third one remains a difficult issue. In the case of a cultural space using social media, providing detailed information means having a complete profile, with correct and precise information considering the communication, information about the organization, working hours, and booking tickets and online shopping. These features are required by the end-users when visiting a social media profile or page of a culture-related space. Considering information dissemination, it is expected that social media presence is frequently enriched with information about the actions of the organization. Usually, one can find information about exhibitions, objects, educational programs, visits or spacial dates. The very fast pace of the data in social media implies that this dissemination of information has to be frequent. Due to the fact that it is impossible to have very high frequency of information dissemination, another factor is proposed for the cultural spaces. Disseminating information in a cultural space means a long time of preparation, and as such, it is difficult to achieve disseminating information very frequently. In this work, we propose the factor of participation as a key factor in order to keep the cultural space up to date (as a term in social media) and consider its appearance as modern and synchronous, while in parallel keep "appearing" more in front of people on social media in order to spread the message they carry as information carriers. Our proposal is the utilization of systems that are able to perform alternative types of analytics on social media big data in order to help cultural spaces take advantage of the information created.

3. Algorithmic Approach

In this work, we present a novel method of social media analytics and, more specifically, the case of Twitter analysis within the scope of a Greek national project entitled PaloAnalytics. The idea that lies behind the analysis is that Twitter provides detailed information about its trending topics per area. In parallel, people tend to put their focus on trending topics, usually described as viral. Virality, as a keyword, is related to the Internet today and seems to be the life boat for both people and social media in the chaotic world of data! In order to take advantage of the fact that Twitter provides direct access (through an API) to information related to posts and users, we put the focus on the following procedure and facts. Cultural spaces, as already mentioned, have the problem of participation. In the chaotic world of data on the Internet, it is very difficult to decide where to put the efforts and time for participation. The idea is to find a means of trending topic classification in such a way so as to denote the trending topics that are mostly related to people that use them to participate in serious conversations. This seems to be a good start for a museum to initiate a conversation in order to achieve message spreading and audience acquisition and engagement. Firstly, trending topics related to a place are provided by a direct API call. The API is said to be caching the results for at least 5 min, which means that searching for trending topics very frequently is not possible. Nevertheless, a frequency of 3 times per hour is considered to be enough for our experiment in order to discover and analyze trending topics in time. Secondly, the users' behaviour towards trending topics is more or less as follows: most people tend to react to trending topics, using either the "like" or "retweet" features of the medium. As a matter of fact, a retweet is somewhat more powerful than a simple like. Furthermore, there are different user profiles on Twitter: people whose profile includes trending topics with a comment, which are usually liked and retweeted, and people whose profile includes mainly retweets of trending posts, produced by the first category of people. Others are just interactive with the aforementioned categories (like or list). Inevitably, a number of other behaviours in the medium exist, but as long as our source of information is trending topics, in this work we will not analyze them. Moreover, by checking the profiles of the aforementioned people based on the followers/following metric, we assume the following four alternatives:

- People with a large number of followers and small number of following;
- People with a small number of followers and large number of following;
- People with small number of followers and following;
- People with large numbers of followers and following.

As a matter of fact, people in the fourth category seem to be rare and did not come up as users of our experimental procedure. By focusing on people of the other three categories, we assume the following generic comments:

- People with a large number of followers seem to be generally more affecting and try to have an account (profile) that has more "original" posts, or at least they write a personal comment even if they are reproducing information.
- People with small numbers of followers are generally reproducing (retweeting in our case).
- Among people with similar post rates, people posting more comments (mentioning) usually have larger number of followers than people that usually do not post comments.
- People with large numbers of followers follow people with large numbers of followers as well, but it is very rare to follow people with a small number of followers.

Research is performed on the exploration of the different user types and user profiles on social media and how they may possibly affect other people ([37–42]). The authors have already presented an analysis on the personalities of social media users in [43,44], where detailed information about the profiles of influencing personalities is presented. In particular, the research work concludes with the assumption that a mixture of an Influencing and Dominant user profile (according to the DiSC personality test [45]), with a writing style that includes viral topics and excess sentiment, seems to be the most influential user type.

At the end of the day, despite the fact that one is presented with the information related to people that she or he follows, it is more than clear that a "Twitter wall" is filled up with information deriving from trending topics and users that seem to be "influencers". This is not to be a source of blame, as trending topics are topics that are flooding the network, so they will inevitably be present in several "walls". In parallel, people that are interacting more, either by posting more things or by being followed by more people, or by mentioning people in their posts, are more likely to come up in a wall.

Taking the aforementioned into consideration, we examine the trending topics of the medium so as to extract information on how they could affect the behaviour of cultural spaces in their online presence. As trending topics are topics that in the end will come up on one's dashboard, it is something to follow in order to recognize and uncover online places for action and message spreading. In fact, in cases where a cultural space needs a place to start its interaction, finding trending topics that can initiate serious conversations is the right place.

As the role of the museum nowadays is changing, the cultural spaces have to face the reality of the digital world accompanied with its rules and culture. As a matter of fact, it seems like the cultural spaces have to act immediately and decisively so that they will change the "flat" culture of the web. Social media is a place to interact and present the alternative view of culture. Our research is focused on recognizing trending topics deriving from people with high "influence" value in order to discover ways to interact.

The idea of our approach is the alteration of the analysis of the trending topics of Twitter, so as not to rank them only according to their volume (which seems to be how they are presented), but add some qualitative features related to a "score", which is assigned to users who post data about trending topics. The idea behind the scoring of trending topics is to help museums realize which are the topics that are more probable to be parts of serious conversations.

On this occasion, trending topics inside a "retweet" by a user with low number of followers should score less than trending topics inside an original "tweet" by a person with large numbers of followers. According to the aforementioned idea, we conclude with the procedure depicted in Algorithm 1.

Algorithm 1 Twitter trending topics analysis.

```
procedure TrendsFetch($woeid$)
    $trends \leftarrow fetchTrendsAPI$
    while $trends$ do
        $tweetssearchForTweets(trend)$
        for hasMoreTweets($tweets$) do
            $tweet \leftarrow$ TweetInformation
            save($tweet$)
            $user \leftarrow$ getUserInformation(tweet)
            save($user$)
            $userPower \leftarrow$ calculateUserPower(user, tweet)
            save($userPower$)
        end for
        $trendPower \leftarrow$ calculateTrendPower(userPower, trend)
        save($trendPower$)
    end while
end procedure
```

The idea is to assign a "power" to each trend, according to the power deriving from the posts that include the trending topic and the "power" of each user of the medium.

In order to measure the power of the users, we utilize metrics such as:

- Number of posts;
- Number of followers;
- Number of following;
- Time registered to the medium;
- Is the user verified or not;
- Retweets of posts that include trending topics;
- Original posts that include trending topics.

The following algorithm provides a power for the user.

$$userPower = verified * (a * (followers/statuses) + b * statusFrequency + c * (followers/following)) \tag{1}$$

$$userPostInfluence = trendingTopicPosts/numberOfPosts \tag{2}$$

$$userPostPower = retweetTrendingPosts/numberOfPosts \tag{3}$$

The first equation utilizes four different metrics. The first factor is related to the user verification. If the user is verified, then the corresponding parameter is set to 1.2, else it is set to 1. On this occasion, people with verified profiles have 20% more power than people without verified profiles. The second metric is the rate of followers based on the number of statuses (posts). This metric provides evidence on the rate of people following according to each tweet posted. The third metric is the status frequency, which is a metric providing information on how many tweets are posted by the user within the time the user is using the social media account. Finally, another factor is the ratio of followers to following. In general, this ratio provides evidence on the number of followers related to the number of users being followed by the user. The factor can have a large variety of controversial values; as such, it is calculated according to the following:

$$followers/following = \begin{cases} 0 & < 1 - \text{the user is following more than their followers} \\ 1 & 1 \leq x \leq 10 - \text{the user has at most 10 times more followers than following} \\ 2 & > 10 - \text{the user has more than 10 times more followers than following} \end{cases}$$

The parameters a and b are weights on how much each of the two aforementioned parameters should count in the final result. According to experimentation, it seems that a ratio of a/b of around 4 provides a balance in the results. Parameter c depicts the weight that should be used for the followers/following rate. In cases where we need to focus on this ratio, the value of c should be close to 30. A generic value should be at around 15.

Equations related to postInfluence and postPower are used in order to recognize each users' post penetration within the medium. These metrics are continuously updated leading to differentiations of the total user power.

As a second step, the qualitative parameters of userPower together with postInfluence and postPower are used when reevaluating the power of trending topics. Each topic has a number of posts in which it appears, and each of these posts has a number of "likes" and retweets. Both are indicators of the trending topic power. In parallel, each of the retweets is performed by a user, whose power is already evaluated and continuously recalculated. As such, the following equation is used in order to measure a postPower.

This procedure implies that from the trending topics retrieved from the medium, we search for original posts (no retweets or mentions fetched) and we re-evaluate the "power" of the posts according to several factors. Each time a trending topic is mentioned in a tweet, its "power" is re-evaluated following the algorithm that is presented.

$$postPower = (a * totalShares + (1 - a) * totalLikes) * userPower \tag{4}$$

In this way, each trending topic fetched is provided with a score, which is the sum of the power of the post. It is updated every 20 min, which is the time interval of the system fetching trending topics. The following section presents the complete system architecture in order to support the recorded data and experimental results on the system execution, and how the results of the system can lead to beneficial results for culture-related spaces.

4. Architecture

The system is based on modular architecture, so as to be able to create each system independently. Figure 1 presents the subsystems of the implemented solution.

4.1. Flow of Information

The designed system utilizes data fetched from Twitter. The fetched data, which include information about trending topics, users and tweets, are directly stored in a relational database, while in parallel, new data are calculated. New data include an assigned user power, the user Post Influence factor, the user Post Power variable and a calculated trending topic power. These metadata are stored in parallel with all the original information fetched from the medium. This information is stored both in the relational database and in a time-series database and a document-based database. After data are stored, they are analyzed in order to be combined and filtered, and they are then provided as grouped information through a Visualization tool (Grafana (Grafana: The open observability platform, https://grafana.com/ (accessed on 30 May 2021))) and a RESTful API (built with Laravel (Laravel: The PHP frameowork for web artisans, https://laravel.com/ (accessed on 30 May 2021))).

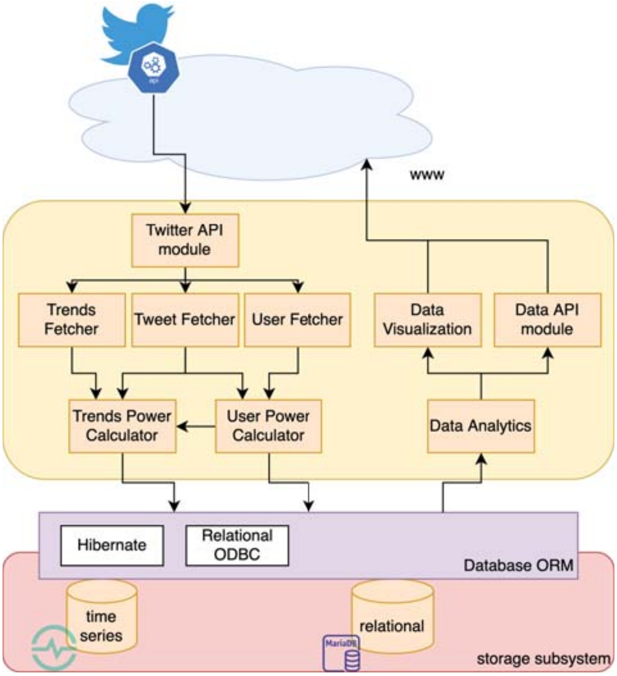


Figure 1. System Architecture.

4.2. Storage Subsystem

The database subsystem consists of three different database management systems and is supported by a database ORM. The storage consists of a time-series database for the storage of information to be visualized in real time, a relational database for permanent storage of all fetched and analyzed data, and a document-based database for storing information to be easily analyzed by taking into account semantic information. Detailed information on the standards of data storage is out of the scope of the present work.

4.3. Data Fetcher

The data fetcher is a system that is responsible for fetching the required information. Information is fetched by utilizing three Twitter API calls (Twitter API—https://developer.Twitter.com/en/docs/Twitter-api (accessed on 30 May 2021)). The first system is responsible for fetching the trending topics. It utilizes the Trending Topics API, which returns a list of "keywords" as the trending topics in a specific area. After the trending topics are received, each of the topics is used as the input to a Twitter Search API endpoint. This returns a number of posts that are relevant to the trending topic. From this procedure, any retweet or mentioned post is omitted, so that only original posts including the trending topics are saved. Finally, for each of the tweets, information about the user that posted it is extracted, followed by an amount of metadata including number of followers, number of posts, if she or he is verified, etc. All the data deriving from the data fetcher are saved to the relational database.

4.4. Power Calculators

The data that are fetched from Twitter are analyzed in order to evaluate a so-called "power". The purpose of this re-evaluation is to measure the strength of a trending topic among the users of the medium. The strength can be considered as the ability to spread in the community. This power is already visible by the volume of tweets containing the trend, but is mainly dependent on countable factors such as the number of likes or retweets. We put our focus on adding another parameter in order to create an algorithm for power evaluation, which is the power of each user. Information about the algorithms is discussed in Section 3. User and trending topic power evaluation leads to storing metadata in the database that accompany the medium's information.

4.5. Data Analytics

Information that is stored in the three different types of databases is analyzed in order to achieve the next step, that is, visualization and access to the structured data through a RESTful API. The analytics include procedures that combine information in trending lines. For example, as long as the fetching algorithm fetches trending topics every 20 min, the trending topics reoccur over time and have to be combined in order to form a "trending line" of information that evolves in time. Another type of analysis that is performed in order to feed mainly the RESTful API is searching for different behaviours in the trending topics. The analytics lead to three different types of trending topics. The first type is topics that appear, have a quite quick time with a high level of power and fade out within a finite period of time (usually within some hours, which is computed as around 12–16 for Greek Twitter). The second type are topics that appear in a very short period of time (usually an hour or less), achieve a low score, and disappear. Finally, the third type is trending topics that either remain present for a long period of time (more than 36 h), or appear in a periodical manner (e.g., every day, the same hour, with the same behaviour). The aforementioned findings can be found both in the visualization and in the RESTful API. For each type of topics, there is evidence on how a cultural space should act. Starting from what is easy to understand is that the second type of posts act like comets. Their appearance is short, fast and superficial, which means that one should not spend much time on these topics. The third topics, the periodical ones, are related to a recurring event, usually a TV show. These topics are not to be given much attention as well. On the other hand, given the fast pace at which the Internet moves, and more specifically the data on the Internet, attention must be paid to the trending topics that remain active for a medium period of time. These are topics that remain active for a period of a day and receive a lot of attention by several different types of users. The analysis performed provides information about such topics and can guide a cultural space to engage with them.

4.6. Data Presentation

The system concludes two types of data presentation in order to support the part of our experimentation that deals with the connection to cultural spaces. Both the data visualization module and the API are used in order to obtain a presentation of the analyzed information and support our findings, in order to enhance the presence of cultural spaces on the web, and specifically social media. Visualization is achieved with the help of Grafana, in which a dashboard including different kinds of real-time information is set up, and the API, which is provided with the help of a Laravel installation that holds the RESful API endpoints. Both of them are presented in Section 5.

5. Experimental Results

The experimentation procedure is separated into two different parts. During the first part, we examine the use of the system and the results from the algorithms' application to the data. We present the results from Twitter trending topics as they are fetched from the corresponding API (without information) and how the system adds metadata for extra analysis. During this procedure, we examine how we utilize the extra information in order to offer data visualization and endpoints from the RESTful API. The second procedure utilizes results from the visualization procedure in order to show how the information presented can be used by a culture-related space in order to provide a benefit during their online presence.

5.1. Trending Topics on Steroids, Ordered

The first part of the system collects information about trending topics, tweets and users in order to enhance the information related to the trending topics. When data are fetched from the medium concerning the trending topics, they include a lot of accompanying information according to Table 1.

Table 1. Example of original Data for Trending Topics (place ID = Greece).

Name	Time	First Appearance	Volume
#covid19gr	16 October 2020 18:10:05	12 April 2020 09:10:05	null
elon musk	5 April 2022 01:30:04	22 October 2020 19:30:03	233,504
ukraine	3 April 2022 20:10:04	25 February 2022 06:10:02	1,411,402

In general, information from the original endpoint includes the name of the trending topic, a URL to search for it, and the impact it appears to have within the medium, if the latter is available. In fact, impact is available for a small number of trending topics, especially in the place of Greece where the system was mainly tested. As such, this information is not taken under consideration in our procedures.

Respectively, after retrieving the tweets related to each trending topic, information about the users are retrieved as presented in Table 2. The information presented are the ones that concern our algorithmic procedure, while original data include more information.

Table 2. Sample of original Data for Users.

Name	Verified	Followers	Following (Friends)	Favorites	Statuses	Created At
Black H_	false	977	2136	7	5	7 July 2019 12:50:34
Matthildi M_	false	28,048	8549	1536	495	25 May 2011 19:55:00
Nick G_	false	13,582	9	100	261	30 September 2013 21:08:23

The table does not present users with small numbers of followers, as these users are by default given a very low userPower according to the described algorithm.

The data that we retrieve are enriched with more information, that is, the userPower, which is calculated according to the algorithm presented in Section 3 and the number of trending topics that the user has posted, and the user post power and the user influence power. The parameters are recalculated every time a user's post is returned as a result in the search-for-posts procedure.

According to our algorithm, the aforementioned sample profiles receive a userPower score. The first profile presents a user with a very low number of posts (statuses), a large number of followers (in comparison to the statuses), and a very high number of following. It seems like a typical profile that has a large number of followers both from the friends (following) and from another factor that is "hidden". This is the type of user that comments a lot, which is the reason that the user has quite a large number of followers. In our procedure, we omit any statuses that are related to conversations and are not original posts.

The second profile has a significantly large power due to the large number of followers. The large number of followers is achieved both by the large number of friends (following) and by the favorites, which reveal very high interaction with others.

The third user is the one with the highest score. It seems that this user has achieved a very large number of followers without having to follow a large number of users, or having to favorite, but it seems that they post on a not so regular basis.

According to our algorithm, Nick G is given a power score of 100.34, Mathildi M a power score of 67.23 and Black H a power score of 4.33. The means that for every trending topic that includes a post from one of the aforementioned users, the trending topic power score will be increased by the score of the user. We should note that this score is counted only once for the specific moment in time that the system will fetch the specific post, related to the trending topic. For each post that is re-fetched as a search result at a different moment in time in the future, the post power will be recalculated. This means that we can possibly measure the instant and accumulated power score of a trending topic in time.

Table 3 presents lists of trending topics fetched at a specific time and how they are ordered after they were given a power score.

Table 3. Trending topics list received on October 2020.

List 1	List 1 Ordered
amka	Prothipourgos (91.81)
KETHEA	KETHEA (78.43)
Rodo	#Tourism (68.88)
Prothipourgos	Rodo (35.43)
#XFactorGR	amka (34.28)
#EuroLeague	#EuroLeague (22.90)
#Tourism	#XFactorGR (20.02)

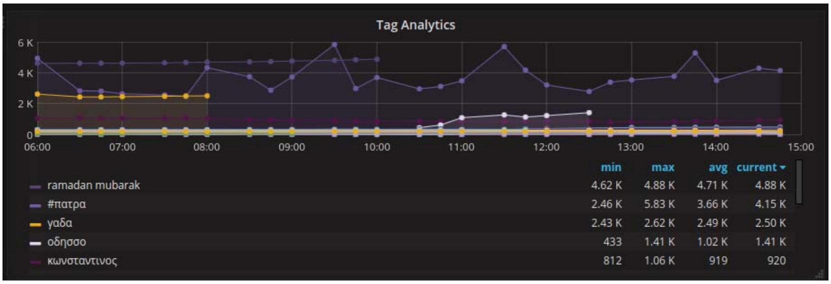
The words are fetched from Greek Twitter and they are originally in Greek language. It is important to note that tourism as a keyword is given a higher score, while xFactor is given a lower score. This simplistic example provides us with information on how a museum can benefit from the trending topics sorting procedure. When it seems that tourism becomes a trending topic, it is given a high score so that a museum can find the right time and space to start interactions in the medium. After all, culture is interconnected with tourism.

The system evaluates the input and decides on the trending topic score. It is able to perform a cold start as it does not have to calculate any value based on its own historical data, but only on historical data provided by the medium. According to qualitative analysis on the trending topics from several different system cycles, it seems that the system is able to calculate a score for the trending topics, so that the topics that can be of interest to a culture-related space seem to be enhanced and score higher. This means that the system is able to create an ordered list of the trending topics in such a way to promote these keywords that are usually present in more formal environments and serious conversations.

5.2. Information Visualization

For the visualization of information, the Grafana tool is utilized. Information visualization is an easy way to present large amounts of information to people, so that they can possibly obtain desired information without having to "dig" through big data. The environment is connected to a time-series database that holds all the information produced by the procedures that are already mentioned. The data that a time-series database needs are the values of the parameters to be displayed, followed by a timestamp. In our study, we store information about the trending topics and their power, as calculated by the internal procedures, together with the total number of posts (volume), favorites (likes) and retweets. The analytics mechanism performs a grouping of the trending topics to be easily depicted in graphs. Figure 2 presents a sample dashboard of a timeline presenting the evolution of trending topics in time.

By selecting a specific term from the legend, it is possible to obtain visualized information about the specific term. Figure 3 presents the presentation of the evolution in time of a specific term.



	min	max	avg	current ▾
ramadan mubarak	4.62 K	4.88 K	4.71 K	4.88 K
#πατρα	2.46 K	5.83 K	3.66 K	4.15 K
γαδα	2.43 K	2.62 K	2.49 K	2.50 K
οδησσο	433	1.41 K	1.02 K	1.41 K
κωνσταντινος	812	1.06 K	919	920

Figure 2. Sample Visualization of trending topics.

Tag Analytics

Figure 3. Single term evolution in time.

Grafana gives the possibility to build dashboards, where the time frame and interval of the analysis can be modified, in order to have different views of the information. Apart from depicting information about all the trending topics in real time, it is possible to create dashboards for each of the different trending topics that appear on the screen. The dashboard includes other parameters that are recorded in the time-series database to have a spherical approach of the collected data. Figure 4 presents a comparison of the trend power metric with the retweets and favorites over time. The representation is proof of the usage of the proposed implemented mechanism. More specifically, despite the fact that the trend keeps having higher and higher numbers of retweets and favorites, the score reaches a peak, and then it drops significantly. Furthermore, while the trend is spread across the medium, the people that write original posts related to it do not have high user power, meaning that the trend is not spread anymore by people that tend to be influencers, losing in this way its power in the medium.

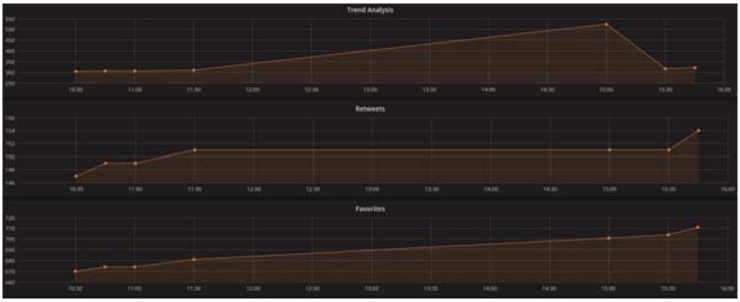


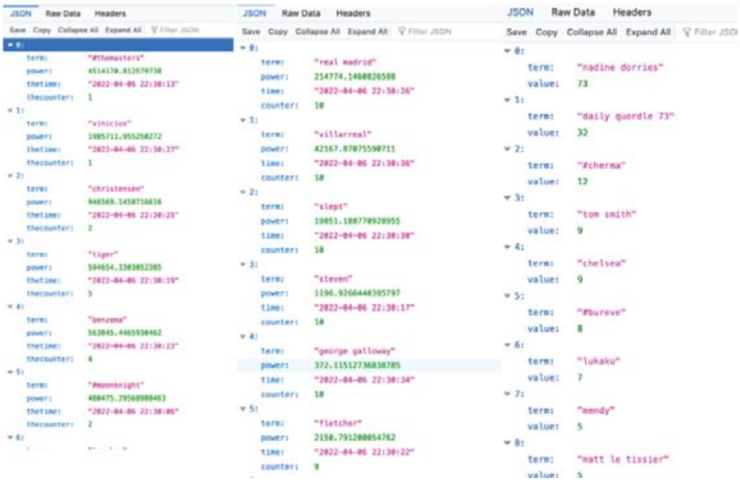
Figure 4. Trend power compared to retweets and favorites over time.

According to the collected information, the data that occur within a time of a day, concerning Greek Twitter, include 50–90 trending topics and 9000–15,000 collected posts. Inevitably, this information is impossible to be parsed manually, and the system is able to provide a simple means of data visualization in order to locate trending topics that may be of interest to a cultural space. In parallel, a testing procedure that was conducted including data from UK Twitter proves that the aforementioned numbers are increased to reach 180–200 trending topics daily with more than 50,000 tweets collected. The numbers prove that the proposed approach can provide a solution to the problem of the vast amount of data and their daily analysis.

In parallel, a second module is able to provide specific information about the behaviour of the trending topics. This is the RESTful API module that can provide information that can be of extreme usefulness. These three are the basic parts of the API:

- Present during the last three hours;
- Persisting trends (for long time);
- Fastest growing trends.

The information from the API is provided in JSON (JSON—JavaScript Object Notation, https://www.json.org/ (accessed on 30 May 2021)) format as presented in Figure 5.



```
▾ 0:
    term:        "#themasters"
    power:       4514170.812579738
    thetime:     "2022-04-06 22:30:13"
    thecounter:  1
▾ 1:
    term:        "vinicius"
    power:       1985711.955250272
    thetime:     "2022-04-06 22:30:27"
    thecounter:  1
▾ 2:
    term:        "christensen"
    power:       948569.1458716618
    thetime:     "2022-04-06 22:30:25"
    thecounter:  2
▾ 3:
    term:        "tiger"
    power:       594654.3303052385
    thetime:     "2022-04-06 22:30:19"
    thecounter:  5
▾ 4:
    term:        "benzema"
    power:       563845.4465930462
    thetime:     "2022-04-06 22:30:23"
    thecounter:  4
▾ 5:
    term:        "#moonknight"
    power:       480475.2956098463
    thetime:     "2022-04-06 22:30:06"
    thecounter:  2
▾ 6:
```

```
▾ 0:
    term:     "real madrid"
    power:    214774.1460826598
    time:     "2022-04-06 22:30:26"
    counter:  10
▾ 1:
    term:     "villarreal"
    power:    42167.87075590711
    time:     "2022-04-06 22:30:36"
    counter:  10
▾ 2:
    term:     "slept"
    power:    19851.188770920955
    time:     "2022-04-06 22:30:30"
    counter:  10
▾ 3:
    term:     "steven"
    power:    1196.9266440395797
    time:     "2022-04-06 22:30:17"
    counter:  10
▾ 4:
    term:     "george galloway"
    power:    372.11512736838785
    time:     "2022-04-06 22:30:34"
    counter:  10
▾ 5:
    term:     "fletcher"
    power:    2150.791200054762
    time:     "2022-04-06 22:30:22"
    counter:  9
```

```
▾ 0:
    term:   "nadine dorries"
    value:  73
▾ 1:
    term:   "daily quordle 73"
    value:  32
▾ 2:
    term:   "#cherma"
    value:  12
▾ 3:
    term:   "tom smith"
    value:  9
▾ 4:
    term:   "chelsea"
    value:  9
▾ 5:
    term:   "#bureve"
    value:  8
▾ 6:
    term:   "lukaku"
    value:  7
▾ 7:
    term:   "mendy"
    value:  5
▾ 8:
    term:   "matt le tissier"
    value:  5
```

Figure 5. Sample results from JSON data execution.

The information provided by using this type of data export can be easily become an input to any system that supports creating a dashboard with JSON input. Despite the fact that information is not quite clear for a person that is not accustomed to raw data, a number of dashboards are based on JSON data. Figure 6 presents the visualization of data through a simple service (jsontoChart (JSONtoChart)—https://jsontochart.com/ (accessed on 30 May 2021)).

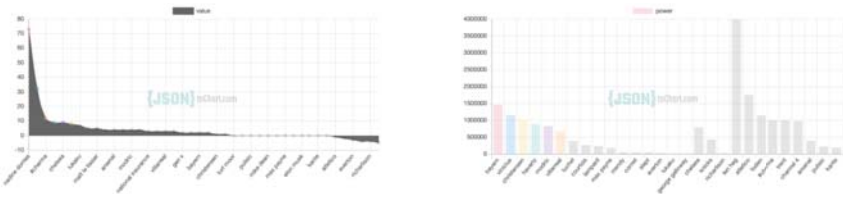


Figure 6. JSON Data to Charts.

5.3. Benefits of Usage

Despite the fact that a number of steps and cutting-edge technologies are used to support cultural spaces under the umbrella of "cultural informatics", there still seems to be a number of steps that need to be taken considering the web presence. The part that seems to be the weakest considering the cultural spaces has to do with the online participation. Participation, in contrast to a generic online strategy, cannot be pre-defined as it is a dynamic procedure that changes from medium to medium and from user to user.

By recognizing the online trends, the cultural spaces are able to locate a place to start their online participation and interaction.

Our approach is to establish trends that will lead to serious conversations, and as such, this is the reason we research user and trend scores. In this way, we try to empower the part of a cultural space strategy that has to do with participation in social media as we narrow down the entry paths for serious conversations. Of course, trending topics that concern a museum or a cultural space may not occur on an hourly or even daily basis. However, our research proved that analysis on trending topics from a medium and a classification to promote trends that lead to serious conversations can be a "place to start" in the chaotic world of the Internet.

6. Conclusions and Future Work

This paper presented the research procedures that were performed as part of a research task of the Greek National Project entitled PaloAnalytics. The research work focuses on the trending topics of Twitter and intends to enhance them with qualitative data. More precisely the scope of the research is to measure the impact of each user and project this information onto the trending topics provided by the medium itself. We proposed an algorithm that takes into account users' information to create an impact score, while in parallel we tried to use this score in order to create an ordering of the trending topics. While the original scope of the research project was limited to trending topic classification, we proceeded to a further step. We tried to interconnect the solution with a problem that exists in museums and cultural spaces. The problem is called participation and is related to the absence of an online presence of the museums when it comes to online conversations. By trying to alter parts of the implemented algorithm in order to classify higher trending topics that derive from specific types of users (with high influence), we claimed that we can possibly help museums and cultural spaces locate a place to initiate a serious conversation.

Technology is nowadays a part of museums and cultural spaces. Modern advances in technology have attracted a lot of spaces and people are expecting museums to have web presence, use high-end technology, digitally communicate their message or their educational activities, and generally have a larger radius of reach. On the other hand, some technological "necessities", such as social media analysis for usage in culture-related spaces, are a concern that affects both people in a museum and researchers related to cultural informatics. This is because it is a matter of high importance, as more and more people tend toward a mixed type of life, while in parallel it seems that the culture of the web tends to be very flat. From the literature review, it seems that several problems have been solved, but participation remains a major issue. Participation cannot be foreseen or predicted. It has to be a dynamic procedure that should adapt both to people and the reality of digital life, which is constantly changing.

In this work, we presented a mechanism that is able to analyze a medium (Twitter) in order to enrich it with information that could be useful for organizations such as museums. This is because the analysis performed can lead to revealing spaces on the Internet where cultural spaces can interact and participate. The results from the system execution show that the qualitative analysis of the medium related to trending topics, in real time, without having to extract very large amounts of data from it, can lead to significantly high-quality results for organizations seeking conversations to intervene and fulfill their "participation" obligation.

The next research step is the actual application of the system within an organization (preferably cultural) in order to apply the results of the system execution in a real environment. However, still, while it is quite straightforward to measure the impact of the information dissemination (reach and reactions), on the contrary, it still remains an issue to measure the impact of the participation procedure.

Author Contributions: Conceptualization, V.P. and M.W.; Methodology, V.P.; Software, V.P.; Validation, V.P. and M.W.; Formal Analysis, V.P. and M.W.; Investigation, M.W.; Resources, V.P. and M.W.; Data Curation, V.P.; Writing—Original Draft Preparation, V.P. and M.W.; Writing—Review and

Editing, V.P.; Visualization, V.P.; Supervision, M.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research has been co-financed by the European Union and Greek national funds through the Operational Program Competitiveness, Entrepreneurship and Innovation, under the call RESEARCH—CREATE—INNOVATE (project code: T1EDK-03470).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Bearman, D.A.; Lytle, R.H. *The Power of the Principle of Provenance*; Archives and Museum Informatics: Totonto, ON, Canada, 1985; Volume 21, pp. 14–27.
2. Bearman, D. *Collecting Software: A New Challenge for Archives & Museums*; Archives and Museum Informatics: Toronto, ON, Canada, 1987. Available online: https://www.archimuse.com/publishing/col_soft/col_soft.Ch7.pdf (accessed on 6 April 2022).
3. Bearman, D. *Functional Requirements for Collections Management Systems*; Archives and Museum Informatics: Toronto, ON, Canada, 1987.
4. Bierbaum, E.G. Records and access: Museum registration and library cataloging. *Cat. Classif. Q.* **1988**, *9*, 97–111. [CrossRef]
5. Bitner, R. *Nomenclature for Museum Cataloguing; A System for Classifying Man-Made Objects*; American Association for State and Local History: Nashville, TN, USA, 1980.
6. Appel, S. Copyright, digitization of images, and art museums: Cyberspace and other new frontiers. *UCLA Ent. L. Rev.* **1998**, *6*, 149. [CrossRef]
7. Bertacchini, E.; Morando, F. The future of museums in the digital age: New models for access to and use of digital collections. *Int. J. Arts Manag.* **2013**, *15*, 60–72.
8. Hirtle, P.B.; Hudson, E.; Kenyon, A.T. *Copyright and Cultural Institutions: Guidelines for Digitization for US Libraries, Archives, and Museums*; Forthcoming, U of Melbourne Legal Studies Research Paper No. 434; Cornell University Library Press: Ithaca, NY, USA, 2009.
9. Hylland, O.M. Even better than the real thing? Digital copies and digital museums in a digital cultural policy. *Cult. Unbound* **2017**, *9*, 62–84. [CrossRef]
10. Terras, M.M. The Rise of Digitization. In *Digitisation Perspectives. Educational Futures Rethinking Theory and Practice*; Rikowski, R., Ed.; SensePublishers: Rotterdam, The Netherlands, 2011; Volume 46.
11. Ciolfi, L.; Bannon, L.J.; Fernström, M. Including visitor contributions in cultural heritage installations: Designing for participation. *Mus. Manag. Curatorship* **2008**, *23*, 353–365. [CrossRef]
12. Rey, F.B.; Casado-Neira, D. Participation and technology: Perception and public expectations about the use of ICTs in museums. *Procedia Technol.* **2013**, *9*, 697–704. [CrossRef]
13. Winter, M. Visitor perspectives on commenting in museums. *Mus. Manag. Curatorship* **2018**, *33*, 484–505. [CrossRef]
14. Petrelli, D.; O'Brien, S. Phone vs. Tangible in Museums: A Comparative Study. In Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems, Montreal, QC, Canada, 21–26 April 2018.
15. Sayre, S. Bring It On: Insuring the Success of BYOD Programming in the Museum Environment. In Proceedings of the Museums and the Web 2015, Chicago, IL, USA, 8–11 April 2015; Proctor, N., Cherry, R., Eds.; Museums and the Web LLC: Silver Spring, MD, USA, 2015.
16. Antoniou, A.; Dejonai, M.I.; Lepouras, G. 'Museum escape': A game to increase museum visibility. In Proceedings of the International Conference on Games and Learning Alliance, Laval, France, 9–10 December 2019; Springer: Cham, Switzerland, 2019; pp. 342–350.
17. Clini, P.; Quattrini, R.; Bonvini, P.; Nespeca, R.; Angeloni, R.; Mammoli, R.; Dragoni, A.F.; Morbidoni, C.; Sernani, P.; Mengoni, M.; et al. Digit (al) isation in Museums: Civitas Project–AR, VR, Multisensorial and Multiuser Experiences at the Urbino's Ducal Palace. In *Virtual and Augmented Reality in Education, Art, and Museums*; IGI Global: Hershey, PA, USA, 2020; pp. 194–228.
18. Lee, H.; Jung, T.H.; tom Dieck, M.C.; Chung, N. Experiencing immersive virtual reality in museums. *Inf. Manag.* **2020**, *57*, 103229. [CrossRef]
19. Shehade, M.; Stylianou-Lambert, T. Virtual reality in museums: Exploring the experiences of museum professionals. *Appl. Sci.* **2020**, *10*, 4031. [CrossRef]
20. Vassilakis, C.; Kotis, K.; Spiliotopoulos, D.; Margaris, D.; Kasapakis, V.; Anagnostopoulos, C.N.; Santipantakis, G.; Vouros, G.A.; Kotsilieris, T.; Petukhova, V.; et al. A semantic mixed reality framework for shared cultural experiences ecosystems. *Big Data Cogn. Comput.* **2020**, *4*, 6. [CrossRef]
21. Vital, R.; Sylaiou, S. Digital survey: How it can change the way we perceive and understand heritage sites. *Digit. Appl. Archaeol. Cult. Herit.* **2022**, *24*, e00212. [CrossRef]

22. Marty, P.F.; Jones, K.B. (Eds.) *Museum Informatics: People, Information, and Technology in Museums*; Taylor & Francis: Tokyo, Japan, 2008; Volume 2.
23. Marty, P.F. Museum informatics and collaborative technologies: The emerging socio-technological dimension of information science in museum environments. *J. Am. Soc. Inf. Sci.* **1999**, *50*, 1083–1091. [CrossRef]
24. Marty, P.F. Unintended consequences: Unlimited access, invisible work and the future of the information profession in cultural heritage organizations. *Bull. Am. Soc. Inf. Sci. Technol.* **2012**, *38*, 27–31. [CrossRef]
25. Kang, X.; Chen, W.; Kang, J. Art in the age of social media: Interaction behavior analysis of Instagram art accounts. *Informatics* **2019**, *6*, 52. [CrossRef]
26. Nuccio, M.; Bertacchini, E. Data-driven arts and cultural organizations: Opportunity or chimera? *Eur. Plan. Stud.* **2021**, 1–18. [CrossRef]
27. Amanatidis, D.; Mylona, I.; Mamalis, S.; Kamenidou, I.E. Social media for cultural communication: A critical investigation of museums' Instagram practices. *J. Tour. Herit. Serv. Mark. (JTHSM)* **2020**, *6*, 38–44.
28. Bosello, G.; Haak, M.V. #Arttothepeople? An exploration of Instagram's unfulfilled potential for democratising museums. *Mus. Manag. Curatorship* **2022**, 1–18. [CrossRef]
29. Gerrard, D.; Sykora, M.; Jackson, T. Social media analytics in museums: Extracting expressions of inspiration. *Mus. Manag. Curatorship* **2017**, *32*, 232–250. [CrossRef]
30. Vassiliadis, C.; Belenioti, Z.C. Museums & cultural heritage via social media: An integrated literature review. *Tourismos* **2017**, *12*, 97–132.
31. Lazzeretti, L.; Sartori, A.; Innocenti, N. Museums and social media: The case of the Museum of Natural History of Florence. *Int. Rev. Public Nonprofit Mark.* **2015**, *12*, 267–283. [CrossRef]
32. Fletcher, A.; Lee, M.J. Current social media uses and evaluations in American museums. *Mus. Manag. Curatorship* **2012**, *27*, 505–521. [CrossRef]
33. Fernandes, A.B. "But will there be visitors?" Public outreach efforts using social media and online presence at the Côa Valley Museum and Archaeological Park (Portugal). *Internet Archaeol.* **2018**, *47*. [CrossRef]
34. Villaespesa, E.; Wowkowych, S. Ephemeral storytelling with social media: Snapchat and Instagram stories at the Brooklyn Museum. *Soc. Media+ Soc.* **2020**, *6*, 2056305119898776. [CrossRef]
35. Zollo, L.; Rialti, R.; Marrucci, A.; Ciappei, C. How do museums foster loyalty in tech-savvy visitors? The role of social media and digital experience. *Curr. Issues Tour.* **2021**, 1–18. [CrossRef]
36. Contri, M. Museums and their audience: Towards dialogic communication through social media? *Int. J. Digit. Cult. Electron. Tour.* **2020**, *3*, 22–35. [CrossRef]
37. Utami, E.; Hartanto, A.D.; Raharjo, S. Systematic Literature Review of Profiling Analysis Personality from Social Media. *J. Phys. Conf. Ser.* **2021**, *1823*, 012115.
38. Apriyanto, S.; Nurhayaty, A. Born In Social Media Culture: Personality Features Impact In Communication Context. In Proceedings of the 2nd ICoLLiT (International Conference on Language, Literature and Teaching), Padang, Indonesia, 22–23 August 2019.
39. Campbell, W.; Baseman, E.; Greenfield, K. Content+ context networks for user classification in Twitter. In Proceedings of the Neural Information Processing Systems (NIPS), Stateline, NV, USA, 5–10 December 2013.
40. De Choudhury, M.; Diakopoulos, N.; Naaman, M. Unfolding the event landscape on Twitter: Classification and exploration of user categories. In Proceedings of the ACM 2012 Conference on Computer Supported Cooperative Work, Seattle, WA, USA, 11–15 February 2012; pp. 241–244.
41. Pennacchiotti, M.; Popescu, A.M. A machine learning approach to Twitter user classification. In Proceedings of the International AAAI Conference on Web and Social Media, Barcelona, Spain, 17–21 July 2011; Volume 5.
42. Uddin, M.M.; Imran, M.; Sajjad, H. Understanding types of users on Twitter. *arXiv* **2014**, arXiv:1406.1335.
43. Poulopoulos, V.; Vassilakis, C.; Antoniou, A.; Lepouras, G.; Theodoropoulos, A.; Wallace, M. The Personality of the Influencers, the Characteristics of Qualitative Discussions and Their Analysis for Recommendations to Cultural Institutions. *Heritage* **2018**, *1*, 239–253. [CrossRef]
44. Poulopoulos, V.; Vassilakis, C.; Antoniou, A.; Lepouras, G.; Wallace, M. Personality Analysis of Social Media Influencers as a Tool for Cultural Institutions. In Proceedings of the Euro-Mediterranean Conference, Nicosia, Cyprus, 29 October–3 November 2018; Springer: Cham, Switzerland, 2018; pp. 236–247.
45. Scullard, M.; Baum, D. *Everything DiSC Manual*; Wiley: Hoboken, NJ, USA, 2015.

big data and
cognitive computing

Article

ID2SBVR: A Method for Extracting Business Vocabulary and Rules from an Informal Document

Irene Tangkawarow [1], Riyanarto Sarno [2,*] and Daniel Siahaan [2]

[1] Informatics Department, Faculty of Engineering, Universitas Negeri Manado, Minahasa 95618, Indonesia
[2] Informatics Department, Faculty of Intelligent Electrical and Informatics Technology, Institut Teknologi Sepuluh Nopember, Surabaya 60111, Indonesia
* Correspondence: riyanarto@if.its.ac.id; Tel.: +62-431-321847

Abstract: Semantics of Business Vocabulary and Rules (SBVR) is a standard that is applied in describing business knowledge in the form of controlled natural language. Business process designers develop SBVR from formal documents and later translate it into business process models. In many immature companies, these documents are often unavailable and could hinder resource efficiency efforts. This study introduced a novel approach called informal document to SBVR (ID2SBVR). This approach is used to extract operational rules of SBVR from informal documents. ID2SBVR mines fact type candidates using word patterns or extracting triplets (actor, action, and object) from sentences. A candidate fact type can be a complex, compound, or complex-compound sentence. ID2SBVR extracts fact types from candidate fact types and transforms them into a set of SBVR operational rules. The experimental results show that our approach can be used to generate the operational rules of SBVR from informal documents with an accuracy of 0.91. Moreover, ID2SBVR can also be used to extract fact types with an accuracy of 0.96. The unstructured data is successfully converted into semi-structured data for use in pre-processing. ID2SBVR allows the designer to automatically generate business process models from informal documents.

Keywords: SBVR; resource efficiency; fact type; operational rules; informal document to SBVR; natural language

Citation: Tangkawarow, I.; Sarno, R.; Siahaan, D. ID2SBVR: A Method for Extracting Business Vocabulary and Rules from an Informal Document. *Big Data Cogn. Comput.* **2022**, *6*, 119. https://doi.org/10.3390/bdcc6040119

Academic Editors: Konstantinos Kotis and Dimitris Spiliotopoulos

Received: 23 August 2022
Accepted: 22 September 2022
Published: 19 October 2022

1. Introduction

Business knowledge is an essential aspect of the early stages of systems development and evaluation in information systems engineering. The features of model-driven development and transformation are essential [1]. Determining business vocabulary and business rules is laborious because it requires much time and many resources. A more straightforward method to define business vocabulary (BV) and business rules (BR) is to conduct interviews and then extract them automatically using a natural language processing approach. The document resulting from the interview is an informal document.

The analysts who build a process model by gathering information require different techniques, such as document review and interviews [2]. Determining semantic business vocabulary and rules is challenging because of problems that result from the interview process. The processes described are not sequential, there are missing processes, and the interview document contains statements that are not relevant to the subject matter (noise). This documentation is not always well-structured and can be challenging to solve.

Informal documents usually refer to information that lacks a data model and that computer programs cannot easily use [3]. According to Praveen and Chandra [4] and Baig, Shuib, and Yadegaridehkordi [5], unstructured documents include files such as word processing documents, spreadsheets, PDFs, social media and message system content, graphical content, videos, and multimedia. The unstructured document does not have structured data information and precise data types and rules to apply to its stored data.

The difference between formal and informal documents is that formal documents are written following specific standards. In contrast, informal documents are more casual, conversational, and do not have a writing standard [6]. Examples of formal documents are documents containing standard operating procedures (SOP), laws and regulations, official script procedures, and policy documents. Examples of informal documents are news documents, documented interview results, memos, personal letters, and software requirements specifications (SRS).

The fact type defines the relationship between different concepts in BR and business process models: the noun concept indicates the name of the actor and the action verb indicates the process [7]. Informal documents that pass through the preprocessing stage are the basis for determining Semantics of Business Vocabulary and Business Rules (SBVR). SBVR is a standard to describe business knowledge in the form of controlled or structured natural language. Research to determine the transformation rules from SBVR to Business Process Modeling Notation (BPMN) in terms of structural rules and operational rules has been carried out. The research illustrated the transformation of data input using data already in the form of SBVR [8,9]. The enhanced Natural Language Processing (NLP) SBVR extraction provides recognition of entities, noun and verb phrases, and multiple associations [10]. They presented NLP-enhanced and pattern-based algorithms for SBVR automatic extraction from UML case diagrams. Previous research on NLP-enhanced algorithms was extended with a model-to-model (M2M) transformation approach [11]. According to Mishra and Sureka [12], there are inconsistencies between BPMN and SBVR. They generated Extensible Markup Language (XML) from a BPMN diagram, extracted triplets (actor, action, and object) using grammatical relations, searched node-induced sub-graphs, and applied algorithms to detect instances of semantic inconsistency. These indicate that recent research developments in natural language aim to deliver automatic model transformation.

In this paper, we present a novel approach to perform automatic translation of the informal document into fact type and operational rules of SBVR, called ID2SBVR. This method bridges the gap between an informal document, such as an open-ended interview, and a process model. We contribute to the model-driven information system development domain by automatic extraction of SBVR related to operational rules (behavioral rules) from informal documents. Specifically, the ID2SBVR searches for the sequence words, extracts the triplet, searches for the actor in a sentence, extracts the fact type, splits the fact type into compound, complex, or compound-complex sentences, and generates the operational rule of SBVR.

2. Related Works

The NLP research that has focused on business process modeling and SBVR has had various proposed methodologies. Several works that concern NLP, SBVR, and BPMN can be separated into six groups: works that discuss business process improvement and business process re-engineering to optimize the process and increase efficiency [13–16]; works that discuss SBVR transformation related to Software Requirements Specification (SRS) into XML [17–19]; works that discuss generating Unified Modeling Language (UML) class models from SRD using NLP [20]; works that discuss transformation from SBVR to BPMN where SBVR structured English (SE) specification is consistent and complete [12,21–24]; works that discuss producing SBVR from UML (use case diagram) [10,11]; and works that discuss generating natural language from business processes [2,25–27]. Further explanation regarding the grouping of related works is discussed below.

The Business Process Management (BPM) is an approach for advancing workflow in order to align processes with customer needs in an organization [13]. BPM covers both business improvement and business process re-engineering [14]. Business Process (BP) focuses on re-engineering of processes and constant process improvement to achieve optimized procedures and increase efficiency and effectiveness [15].

Aiello et al. [17] investigated a mapping methodology and SBVR transformation grammar to produce rules that are ready to process in a rule engine. The main objective of their research is to overcome some weaknesses in the software development process that can result in inconsistencies between the identification of domain requirements and the functionality of the software implemented. Arshad, Bajwa, and Kazmi [18] provided an approach for translating SBVR specifications of software requirements into an XML schema. The translation mapped verb concept, noun concept, characteristic, and quantification. Akhtar et al. [19] generated a knowledge graph based on Resource Description Framework SBVR (RDFS) from SBVR. They used SBVR rules and created a triplet (actor, action, and object), then generated the RDF and the RDFS [28].

Mohanan and Samuel [20] generated UML class models instantly from software requirement specifications (SRS) using a modern approach. Their approach used OpenNLP for lexical analysis and generated required POS tags from the requirement specification. In their further research, they developed a prototype tool that can generate accurate models in a shorter time [29]. It reduces the cost and budget for both the designers and the users.

BP modeling has a long-standing tradition in several domains. This discipline persists in the constant improvement of process and issue solving [21]. They examined the basic principle and the disparity between the specifications of BV and BR modeling and BP modeling. Another research transformed BR in SBVR into BPMN to assist the business expert in the requirement validation phase [22]. The focus was on the model transformation where the SBVR Structured English (SE) specification is consistent and complete. Kluza and Honkisz [24] presented an interoperability solution for transforming a subset of the SBVR rules into the BPMN and Decision Model and Notation (DMN) models. They combined process and decision models with translation algorithms to translate the SBVR vocabulary and structural and operational rules. Bazhenova, et al., [30] succeeded to identify a group of patterns that grab potential data representations in BPMN processes and it can be used to conduct the derivation of de-cision models related to current process models. Purificação and da Silva [31] succeeded in validating SBVR business rules that deliver content to assist users writing SBVR rules. This method supplied the functionality to update parts of the defined grammar with runtime and to locate and extract verb concepts that can be validated from the BR. Mishra and Sureka [12] investigated automatic techniques to detect inconsistencies between BPMN and SBVR. The research transformed rules to graphics and applied subgraph-isomorphism to detect instances of inconsistencies between BPMN and SBVR models.

Danenas et al. [10] succeeded in producing the SBVR from UML (use case diagrams) by automatic extraction. This research enhanced recognition of entities, entire nouns and verb phrases, improved various associations extraction capabilities, and produced better quality extraction results than their previous solution [11]. Their main contributions were pre- and post-processing algorithms and extraction algorithms using a custom-trained POS tagger.

Rodrigues, Azevedo, and Revoredo [25] investigated a language-independent framework for automatically generating natural language texts from business process models. They found empirical support that, in terms of knowledge representation, the textual work instructions can be considered equivalent to process models represented in BPMN. The research investigating the natural language structure showed that mapping rules and correlations between words representing the grammatical classes indicate a process element through keywords and/or verb tenses [2]. Furthermore, a semi-automatic approach successfully identified process elements from the natural language with a process description [26,27]. There were 32 mapping rules to recognize business process text elements using natural language processing techniques. This was discovered through an empirical study of texts comprising explanations of a process [2].

This current study presents two principal novel outcomes in terms of natural language processing and translating informal documents into SBVR. First, ID2SBVR generates the operational rules of SBVR from fact type, and, second, it can extract fact types from informal documents.

3. Materials and Methods

This section describe the research objectives of this study and the method of how ID2SBVR extracts operational rules of SBVR from informal documents. The method is explained in further detail in the following subsections.

3.1. Research Objectives

The main research objectives are: (i) to develop an implementation from concept to a fully functioning method to translate informal documents to SBVR; (ii) to analyze the correctness and accuracy of automatic fact type extraction by ID2SBVR; and (iii) to analyze the method's correctness and accuracy in generating operational rules of SBVR.

3.2. Mining Fact Type Candidate

An SBVR consists of business vocabulary (BV) and business rules (BR). BV is a vocabulary under business jurisdiction, and a BR is a rule under business jurisdiction [32]. A business vocabulary is composed of a noun concept, a verb concept, and an object type. In this work, we use a subset of the BV concept:

- The general concept is a noun concept. It is classified by its typical properties, e.g., noun person, noun place, noun thing;
- The verb concept can be auxiliary verbs, action verbs, or both.

BR signifies specific contexts of business logic; every BR is based on at least one fact type (FT). A combination of a verb concept and a noun concept is a fact type. The fact type determines the relationship between different BR concepts in BP models. The noun concept represents the actor, and the action verb concept represents a process. RuleSpeak in Object Management Group Annex-H [33] is an existing, well-documented BR notation developed by Business Rule Solutions (BRS). RuleSpeak is a set of rules for conveying business rules in a brief, business-friendly style [34]. It is not a language or syntax like structured English but rather a set of best practices for speakers of English.

BR in SBVR specifies two kinds of rules, structural and operational. Structural rules use such modal operators as necessary or possible/impossible. Operating rules use such modal operators as obligatory, permitted/forbidden [20,25]. The organizational settings are defined with rules of definition or structural rules in SBVR. For example, "**It is necessary that** each customer *has* one customer ID". The behavior of the noun person is defined with behavior rules or operation rules in SBVR. For example, '**It is obligatory that** librarian *sorts* books **after** librarian *receives* books'. Notation standard of SBVR as the controlled natural language used in this research [32]:

term of a noun concept that is part of used or defined vocabulary.
name for individual concepts and numerical values
verb for a fact type that is usually a verb or preposition, or both
keyword that accompanies designations or expressions; for example, obligatory, each, at most, at least, etc.

The SBVR notation that the ID2SBVR generates supports both SBVR structured English and RuleSpeak. Keyword modals available in BPMN include only the 'must' (or 'It is obligatory that') modal keyword because there is no way to present other modal operation in BPMN [20,27]. We executed part-of-speech (POS) and dependency parsing of all the example sentences in this paper using the Natural Language Processing (NLP) online software executor (https://corenlp.run; accessed on 12 April 2021) by the NLP Group of Stanford University (https://nlp.stanford.edu; accessed on 30 March 2021).

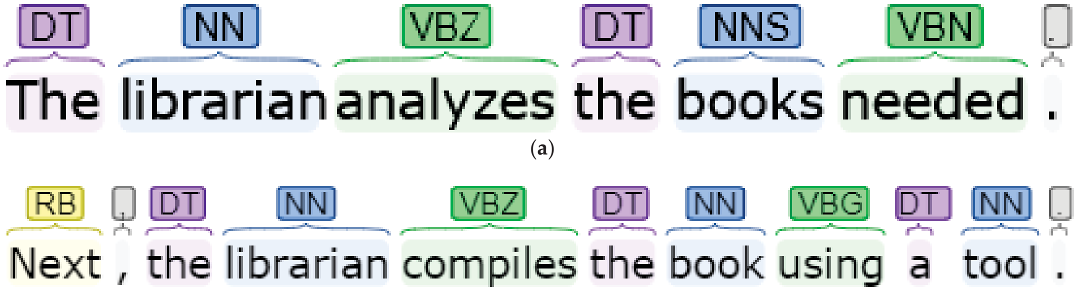
We consider only operational rules because they focus directly on the proper conduct of business activities, which in turn can be transformed into BPMN. To generate operational rule of SBVR, we perform several steps, and in the early stages, we search for a fact type candidate. The required fact type is one that has a noun, an active verb, and an object, which together form a triplet [12]. Sentence 1 and sentence 2 below illustrate fact type:

Sentence 1: 'The librarian analyzes the books needed'.

Sentence 2: ‘Next, the librarian compiles the book using a tool’.

Sentence 1 and sentence 2 are examples of a simple sentence taken from an interview document. Figure 1a,b are the result from NLP that illustrated the POS from sentence 1 and sentence 2. The color and text in POS showed the label of POS tagging, where purple indicates DT label as determiner, blue indicates NN label as singular noun, blue indicates NNS label as plural noun, green indicates VBZ label as verb for third person singular in the present simple, green indicates VBN label as verb for past participle, green indicates VBG label as verb for gerund or present participle, and yellow indicates RB label as adverb. From the POS from both sentences, the fact type candidate is the arrangement of NN or NNS, VBZ, and objects consisting of arrays of NN or NNS, DT, VBN, and VBG. In the fact type rule, all DT and RB in the beginning of the sentence are not used. The fact type of sentence 1 and sentence 2 are:

‘Librarian analyzes the books needed.’
‘Librarian compiles the book using a tool.’


DT
NN
VBZ
DT
NNS
VBN
.
The librarian analyzes the books needed .
RB
,
DT
NN
VBZ
DT
NN
VBG
DT
NN
.
Next , the librarian compiles the book using a tool .

(**a**)

(**b**)

Figure 1. Part-of-speech of (**a**) sentence 1 and (**b**) sentence 2.

The BR structure as an operational rule of SBVR should be:

’It is obligatory that (fact type sentence 2) **after** (fact type sentence 1)’.

The operational rule of SBVR of sentence 1 and sentence 2 is:

’It is obligatory that librarian compiles the book using a tool **after** librarian analyzes the books needed’.

Based on the example above, the fact type consists of a noun (NN), a verb (VB/VBZ), and a noun as an object. In Figure 2 showing this research framework, the input document uses the interview document written in English. In the preprocessing step, the interview data is separated based on the interviewer’s questions and interviewee’s responses. The ID2SBVR searches the sentences for sequence words and the order between them. The ID2SBVR must indicate all the sequence words. Furthermore, the sentences that are not indicated have a sequence word parser to detect dependency and grammatical relations. The ID2SBVR searches the triplet (subject, active verb, and object) to fulfill the required fact type as operational rules. After that, the ID2SBVR determines the actor of each sentence with a noun. Then, the ID2SBVR extracts the fact type from fact type candidates. Next, ID2SBVR extracts the process name. Finally, the ID2SBVR generates the SBVR of operational rules.

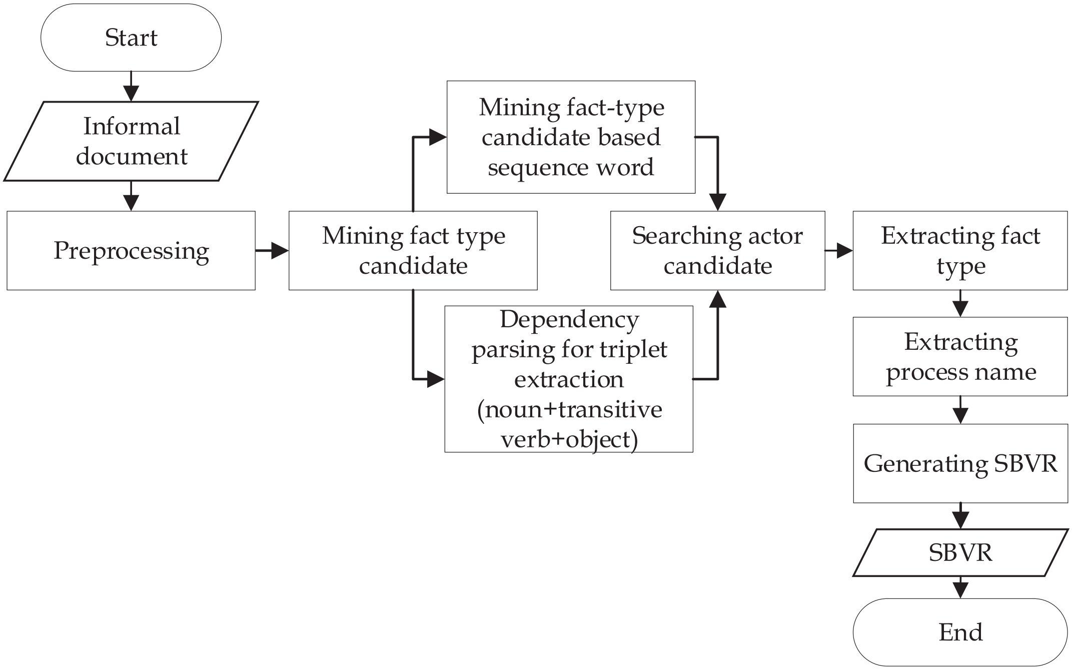

Start
Informal document
Preprocessing
Mining fact type candidate
Mining fact-type candidate based sequence word
Dependency parsing for triplet extraction (noun+transitive verb+object)
Searching actor candidate
Extracting fact type
Extracting process name
Generating SBVR
SBVR
End

Figure 2. Research framework.

3.2.1. Mining the Fact Type Candidate Indicated by Sequence Words

Research on the classification of textual functions is primarily based on Hyland's model [35] of linguistic expressions in academic texts and semantic categories of linking adverbs [36]. There are five main functional semantic categories in Wang [37]: enumeration, code glosses, structuring signals, transition signals, and causal-conditional signals. Our research focused on enumeration especially for sequential words. Enumeration lists a series of steps, procedures, ideas, or subparts of the text, e.g., 'first', 'then', and 'finally', called sequence words.

At this stage, case folding is performed to change all letters in the document to lower case [38]. Any characters other than letters are removed, then tokenization is performed. Tokenization is the procedure of separating a text into words, phrases, or other meaningful parts, called tokens [38].

The ID2SBVR checks each sentence containing the sequence word. Sentences that have sequence words automatically become candidate fact types. The sequence words in question are: 'begins', 'starts', 'after', 'then', 'next', 'after that', 'when', 'finally', etc. The sentence with the sequence word in the interview is followed by an active sentence with a transitive verb. Furthermore, for sentences that do not have a sequence word, the next sentence is checked. If the following sentence has middle and ending sequence words, then the sentence is a fact type candidate. Middle and ending sequence words include 'after', 'then', 'next', 'after that', 'when', 'finally', etc. Algorithm 1 solves the problem of searching the fact type candidate based on sequence words.

3.2.2. Mining the Fact Type Candidate Indicated by Dependency Parsing

In this second phase of the solution, we used Python programming language with an open source library called the Natural Language Toolkit (NLTK) [39]. We consider only those rules which are action-oriented. The NLTK parse tree information represents the grammatical relation between words in the English structure. The result of dependency parsing requires triplet extraction. Sentences with transitive verbs are needed to determine operational rules in BR.

The required nouns for this process are those related to the actor of the fact type. The next stage will be very dependent on the grammatical relation of the words in each sentence. The example of a simple sentence as a fact type candidate taken from an interview response in an informal document is ‘The librarian analyzes the books needed’.

Algorithm 1. Fact type candidate based on sequence words

```
Data: input = answer
Output: fact type candidate
sequence word 1 = [‘begins’, ‘starts’, ‘firstly’, ‘secondly’, ‘first’, ‘after’, ‘then’, ’next’, ’after’, ’that’,
‘when’, ‘finally’, ‘furthermore’, ‘at the end’]
sequence word 2 = [‘after’, ’then’, ’next’, ’after that’
data_clean = []
or answer in data_interview:
    change answer into lowercase
    change answer into token
        if answer is not in sequence word 1:
            if(index i is not = answer length—1):
                insert next sentence into data temporary
                change data temporary into token
            if answer is not in sequence word 2
                insert answer into fact type candidate
        end
    end
    end
else:
    insert answer into fact type candidate
end
```

The dependency parsing of the simple sentence in Figure 3 shows the structure of a simple sentence indicating fact type candidates. Figure 3 shows the structure of the sentence:

Noun (NN) as noun subject (nsubj):‘librarian’
verb, 3rd person singular present simple (VBZ) as a transitive verb: ‘analyzes’.
object consists of noun plural (NNS), verb past participle (VBN): ‘the books needed’.

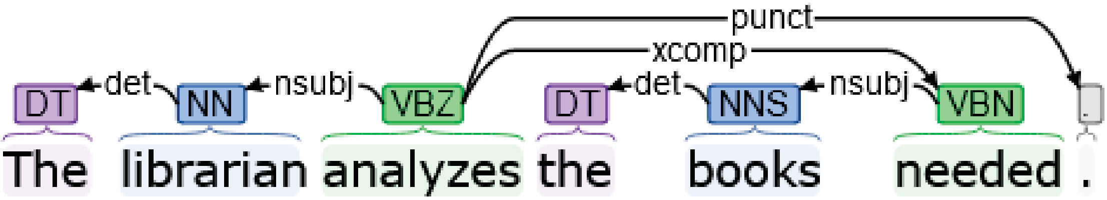


Figure 3. Dependency parsing of simple sentence.

The sentence structure consists of a noun (NN), a verb (VBZ), and an object.

The ID2SBVR determines the fact type candidate based on triplet extraction using Algorithm 2.

Algorithm 2. Fact type candidate based on triplet extraction

```
Data: input = answer
Output: Fact type candidate
for answer in enumerate (data_interview)
    if answer not in fact type candidate
    check noun subject in answer
    if noun subject exist in answer
        for index w in answer
        change answer to token
        if answer in tokens
        insert token with noun subject into data temporary
            insert token with verb into data temporary
            if token with verb = 'of': #check temporary verb with of
            for iteration as many as nlp
            check initialization= True
            if answer exist verb and answer is not exist 'of'
            check initialization= False
            if check = False:
                take index before the sentence
                insert sentence after index before the sentence
        end
        end
        end
        else:
        insert temporary subject
        insert temporary verb
        for iteration as many as nlp
        check initialization = True
        if answer verb followed by determiner (the) and object
        check initialization = False
        end
        else if answer verb followed by object
        check initialization = False
        if check initialization = False:
        take index before the sentence
        insert sentence after index before the sentence
        end
        end
    end
    end
end
end
```

3.3. Searching Actor Candidate

In this phase, we focus on searching for the actor candidate in the process. Actor candidate is a main actor of a fact type. The ID2SBVR searches the dependency parsing result that categorizes as nsubj (noun subject). We use the synset in WordNet NLTK corpus reader. Synset is a synonym set of words that share an ordinary meaning [39]. We use synset in searching all the noun subjects (nsubj) in the fact type candidate. Unfortunately, synset indicates each word as a noun, not a noun phrase. There are actors with a noun phrase, e.g., 'head of the library', 'the head of the employee subsection', and 'member candidate'. So, the ID2SBVR indicates the actor phrase using Algorithm 3.

Algorithm 3. Actor candidate

```
Data: data_interview
Output: actor candidate
for answer in enumerate (data_interview)
process answer to nlp
check result nlp process exist noun object
    if result nlp process exist noun object
    change answer to token
    for index w iteration as many answer exits noun subject
        if index w exist token
        insert token with noun subject into data temporary
            insert token with verb into data temporary
            if token with verb = 'of'
            for index i, t iteration as many nlp result
            check = True
            if answer exist verb and not an object
            if answer exist punct
            insert subject with answer
        end
        else:
            if the next word is followed by pucnt:
            insert answer to data temporary
        end
        else:
            insert answer with space to data temporary
            end
            end
        end
        if answer exist noun subject
        if previous word exist determiner
        insert previous word to data temporary
        else:
        insert subject into temporary subject
        if answer exist verb and answer is not exist 'of'
        check = False
        if check = False:
        if data temporary subject is not noun subject:
        show subject data temporary
        reset temporary
        break
        end
        end
        end
    end
    end
end
```

3.4. Extracting Fact Type

In this phase, we focus on extracting the complex fact type into a fact type. A sentence as a fact type has a noun, an active verb, and an object (see Section 3.1). Mishra and Sureka [12] named it a "triplet" (noun, verb, object). A sentence with more than one fact type categorizes as a compound sentence, a complex sentence, or a compound-complex sentence.

A compound sentence has a coordinating conjunction (CC) that joins two independent clauses, e.g., 'for', 'and', 'nor', 'but', 'or', 'yet', and 'so' [40]. Except for very short sentences, a comma (,) appears right before the coordinating conjunction. The example of a compound sentence in Figure 4 shows the CC 'and' join two independent clauses.

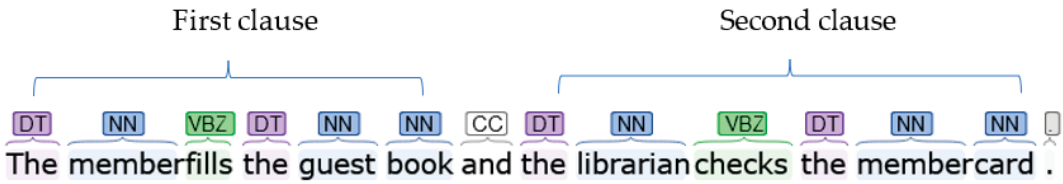

First clause
Second clause
DT NN VBZ DT NN NN CC DT NN VBZ DT NN NN
The memberfills the guest book and the librarian checks the membercard .

Figure 4. A compound sentence.

Based on the fact type structure, the compound sentence in Figure 4 should split into two fact types as two independent clauses. The first fact type is the clause before the CC, and the second fact type is the clause after the CC. The other step in splitting the compound sentence is identifying the fact type with no noun as a noun subject.

The second clause after the CC starts with the verb 'writes.' The noun subject should be 'librarian' as the first clause. The ID2SBVR adjusts it with the noun subject of the first fact type. Algorithm 4 shows the procedure for splitting the compound sentence into simple sentences as fact type. There are seven coordinating conjunctions listed. The fact types of Figure 4 are:

Fact type 1: 'Member fills the guest book'.
Fact type 2: 'Librarian checks the member card'.

Algorithm 4. Fact type from compound sentence

```
Data: Fact type candidate
Output: Fact type
sentence=compound sentence
split data (r 'and | for | nor | but | or | yet | so', sentence)
show data
```

A complex sentence has one independent clause and one or two dependent clauses [40]. It always has a subordinating conjunction ('because', 'since', 'after', 'although', 'when') or a pronoun, such as 'who', 'which', and 'that'.

Based on the fact type structure, a complex sentence in Figure 5 should split into two fact types. The first fact type is the clause before the subordinating conjunction 'before', and the second fact type is the clause after the subordinating conjunction. If there is no noun subject in the double fact type (after the subordinating conjunction or comma ','), the ID2SBVR adjusts it with the first fact type's noun subject. Algorithm 5 shows the algorithm for splitting the complex sentence into a simple sentence as fact type. All the subordinating conjunctions are listed to make the splitting easy.

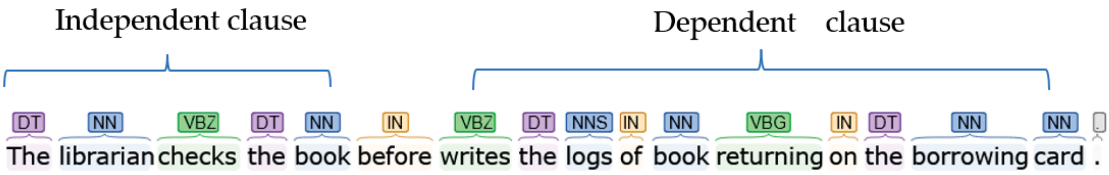

Independent clause
Dependent clause
DT NN VBZ DT NN IN VBZ DT NNS IN NN VBG IN DT NN NN
The librarian checks the book before writes the logs of book returning on the borrowing card .

Figure 5. The example of a complex sentence.

The fact types of Figure 5 are:

Fact type 1: 'Librarian checks the book'.
Fact type 2: 'Librarian writes the logs of returning book on the borrowing card'.

A compound-complex sentence is a sentence with a combination of a compound sentence and a complex sentence [40]. A compound-complex has three or more clauses and at minimum has two independent clauses and one dependent clause.

Based on the fact type structure, a compound-complex sentence in Figure 6 should split into 3 fact types. The first fact type is the clause before the subordinating conjunction 'then', the second fact type is the clause after the subordinating conjunction, and the third fact type is the clause after the CC 'and'. Algorithm 6 shows the process to split a compound-complex sentence. All the coordinating conjunction words (CC) and subordinating conjunction words are listed in the Algorithm 6.

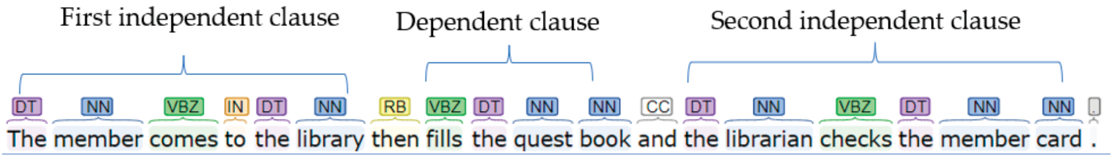

First independent clause
Dependent clause
Second independent clause
DT NN VBZ IN DT NN RB VBZ DT NN NN CC DT NN VBZ DT NN NN .
The member comes to the library then fills the quest book and the librarian checks the member card .

Figure 6. The example of compound-complex sentence.

The fact types of Figure 6 are:

Fact type 1: 'Member comes to the library'.
Fact type 2: 'Member fills the guest book'.
Fact type 3: 'Librarian checks the member card'.

Algorithm 5. Fact type from compound-complex sentence

```
Data: Fact type candidate
Output: Fact type
sentence = 'compound-complex sentence'
Split data (r 'and | for | nor | but | or | yet | so | after | once | until | although | then | provided
that | when | as | rather than | whenever | because | since | where | before | so that | whereas |
even if | than | wherever | even though | that | whether | if | though | while | in order that |
unless | why', sentence)
Show data
```

3.5. Extracting Fact Type

At this stage, the ID2SBVR extracts the process name and separates each fact type based on the name of the extracted process. The process name comes from the list of questions in the data. The process name is important for the transformation of SBVR into BPMN. BMPN will use it as the pool name. Algorithm 7 extracts the process name from the interview questions in the data.

Algorithm 6. Extracting process name

```
Data: Input = Question
Output: Process name
    If previous question is not question data:
    Show new line
    show question
    process question to nlp process
        for index t iteration as many result of nlp process
            if question exist compound type
                if previous word of question exist amod type
                insert result of nlp process with amod type to
```

Algorithm 6. *Cont.*

```
                    temporary data
                for iteration compound type until the last word
                if result of nlp exist punct type
                insert result of nlp process to temporary data
                show temporary data
                end
            end
        end
end
end
```

3.6. Generating SBVR

In this phase, we focus on the RuleSpeak SBVR for operational rules generated from the fact type. We are concerned with the operational rules in SBVR because our future research is to transform the operational rules into BPMN. The ID2SBVR uses the keywords or RuleSpeak SBVR with:

'It is obligatory that <fact type2> after <fact type1>'.

Algorithm 7 generates a fact type into an operational rule in SBVR. The ID2SBVR must consider the order of the fact type.

Algorithm 7. Fact type operational rule in SBVR

```
Data: Fact type candidate
Output: SBVR
#prosessbvr
topic = 0
sbvr S = result of sbvr process interview data
sbvr C = result of sbvr process check sentence
for index i iteration as many as result of sbvr process
    if index i = 0 or sbvr is not exist previous sbvr
        show topic
        end
        if index i is not exist complete sentences and sbvr S exist next
        sbvr S and sbvr C is 0 and next sbvr C is 0
        print ('It is obligatory that', next complete
        sentence, 'after', complete sentence)
    end
end
```

Based on the extracting fact type in the previous phase, we generate the fact type operational rule in RuleSpeak SBVR. The example of the SBVR follows.

'It is obligatory that member candidates fill and complete the form **after** librarian submits it to the circulation or reference sub—librarian section afterwards'.

'It is obligatory that member candidates enclose tuition fee slip and two pass member candidates photos **after** member candidates fill and complete the form'.

3.6.1. Conjunction

Conjunction is a binary logical operation that formulates that the meaning of each of its logical operands is true. The example of a conjunction appears below. Response 2 shows the compound sentence as a conjunction. In response 2 with conjunction, we separate the noun subject, verb, and object.

Response 1: 'Firstly, the librarian determines the exhibition themes'.
Response 2 with conjunction: 'The librarian selects material, librarian determines design, librarian prepares support event, and librarian prepares promotion concept'.

Response 3: 'Then, the librarian does the exhibition together with the team'.

The ID2SBVR generates the fact type based on the response above. Fact type 1 generates from response 1, fact type 2 generates from response 2 with conjunction, and fact type 3 generates from response 3.

Fact type 1: 'librarian determines the exhibition themes'.
Fact type 2 with conjunction: 'librarian selects materials, librarian determines design, librarian supports event, and librarian prepares promotion concept'.
Fact type 3: 'librarian does the exhibition together with the team'.

The ID2SBVR generates the SBVR 1 and SBVR 2 based on the fact type 1, fact type 2 with conjunction, and fact type 3.

The composition of SBVR 1:

<**It is obligatory that**> fact type 2 with conjunction separated with 'and' <**after**> fact type 1.

The composition of SBVR 2:

<**It is obligatory that**> fact type 3 <**after**> fact type 2 with conjunction separated with 'and'.

SBVR 1:

'It is obligatory that librarian selects materials **and** librarian determines design **and** librarian supports event **and** librarian prepares promotion concept **after** librarian determines the exhibition themes.'

SBVR 2:

'It is obligatory that librarian does the exhibition together with the team **after** librarian selects materials **and** librarian determines design **and** librarian supports event **and** librarian prepares promotion concept.'

3.6.2. Exclusive Disjunction

Exclusive disjunction in SBVR is a binary logical operation that indicates that the meaning of one logical operand is true and the meaning of the other logical operand is false [32]. Fact type with exclusive disjunction:

fact type: 'member active registered'
fact type: 'member allowed to enter the library **else** not allowed to enter'.

SBVR:

'It is obligatory that member allowed to enter the library **after** member active registered **else** not allowed to enter'.

3.6.3. Inclusive Disjunction

Inclusive disjunction is a binary logical operation that indicates that the meaning of at least one of its logical operands is true [32].

Response 1 with disjunction:

'The librarian categorizes scientific paper as thesis, librarian categorizes scientific paper as dissertation, or librarian categorizes other scientific paper'.

Response 2:

'Then, librarian puts the scientific paper in the cabinet'.

The ID2SBVR generates fact type 1 from response 1 and fact type 2 from response 2.
Fact type 1 with inclusive disjunction:

'librarian categorizes scientific paper as thesis'
'librarian categorizes scientific paper as dissertation'
'librarian categorizes other scientific paper'

Fact type 2:

'librarian puts the scientific paper in the cabinet.'

SBVR:

'It is obligatory that librarian puts the scientific paper in the cabinet **after** librarian categorizes scientific paper as thesis **or** librarian categorizes scientific paper as dissertation **or** librarian categorizes other scientific paper'.

4. Results and Discussion

This chapter describes the scenario of the experiment, a case study in a university library, and the results and discussion of each stage in this research.

4.1. Scenario

The experiment was performed to answer the basic question, "What is the correctness and accuracy of automatic extracting of fact type and generating SBVR from interview response as an informal document when compared to the benchmark result provided by manual extraction?" An evaluation was carried out to measure the correctness and accuracy of the information retrieval (IR) produced by the ID2SBVR.

The confusion matrix is used to measure the classification method's performance, in this case, precision, recall or sensitivity, specificity and accuracy. In basic terms, the confusion matrix contains information that compares the system classification results carried out with the expected results [41]. We defined the confusion matrix rule, i.e., the data extracted true as true positive (TP); data extracted false as false positive (FP); deviation between data extracted false and benchmark as false negative (FN); and deviation between data extracted true and benchmark as true negative (TN).

The IR evaluation metrics commonly used in text classification are precision, recall, fscore, and accuracy [42]. We evaluated the ID2SBVR developed to extract operational rules in terms of precision, recall, specificity, and accuracy. Precision is the ratio of correctly identified fact type to the total number of fact type extracted; the recall is a ratio of correctly determined fact type to several suitable fact types; accuracy is the accurate prediction ratio (positive and negative) to the overall data. We need to measure the precision and recall rather than accuracy because, with accuracy, the results do not necessarily match the necessary data [43]. The research of Skersys, Danenas, and Butleris [11] measures the accuracy of automatic and wizard-assisted semi-automatic extraction of SBVR from UML.

The proposed approach for extracting the fact type and generating SBVR were evaluated using one scenario. The scenario of this experiment involves the list of procedures in the library. The evaluation phase consists of the procurement process, inventory process, processing section, member registration, book borrowing process, book returning process, reference sub-section process, librarian promotion process, and the process to review the promotion document. There are six steps of the evaluation to extract the fact type and generate SBVR, i.e., (1) build a ground truth from fact types extracted by domain expert; (2) build a ground truth from SBVR extracted by domain expert; (3) measure the performance of ID2SBVR in extracting fact types using precision, recall, specificity and accuracy; (4) measure the accuracy of ID2SBVR in generating SBVR.

To build the ground truth in steps (1) and (2), we involved domain experts in the field of business process modeling. Domain experts manually executed the complete set of all processes in the library. The domain expert determines all the fact types of the semi-structured interview document. After that, the domain experts compile fact types into SBVR. The result of fact types and SBVR from domain experts can directly be used to evaluate the ID2SBVR method.

4.2. Description of University Library Case Study

This experiment uses a case study consisting of informal data as the interview response written in standard English. The interviews took place in two university libraries. The first university library has 24,354 book titles with 71,368 copies. This library has 12 librarians and 14 staff. The second university library has 145,252 book titles with 210,605 copies in its database. Of these, in its physical collection, this library has 89,672 book titles with 138,391 copies. The working staff consists of 27 librarians and 24 administrative staff and officials.

In the pre-processing phase, we divided all the questions and responses into columns. We also split each response per sentence into rows. Every question has more than one response. Each row response represents one sentence. The first dataset contains 1247 words, forming 110 sentences containing 61 sequence words. The details of the dataset are shown in Table 1 and the complete dataset has been published in online repository as dataset_university (https://doi.org/10.6084/m9.figshare.15123879.v1). The other dataset, dataset_university (https://doi.org/10.6084/m9.figshare.15123972.v2), contains 2585 words across 288 sentences containing 195 sequence words. The summary details of the dataset are shown in Table 2.

Table 1. Dataset_university1.

Question	**Process**	**Number of**				**Sentence**		
ID	**Name**	**Sentence**	**Word Response**	**Verb**	**Sequence Word**	**Compound**	**Complex**	**Compound-Complex**
q1	-	2	24	3	-	-	-	-
q2	procurement section	12	146	27	6	3	1	-
q3	processing section	6	74	6	5	1	2	-
...		...	...	...	...	...	...	...
q16	inventarisation	6	61	5	4	-	-	1
Total		110	1247	171	61	11	10	7

Table 2. Dataset_university2.

Question	**Process**	**Number of**				**Sentence**		
ID	**Name**	**Sentence**	**Word Response**	**Verb**	**Sequence Word**	**Compound**	**Complex**	**Compound-Complex**
r1	-	4	63	6	-	5	-	-
r2	LPBP Lelang	8	60	8	7	-	-	-
r3	LPENGOLBP book	7	51	7	6	-	-	-
...		...	...	...	...	...	...	...
r33	Training, seminars, and workshops held in library	18	264	17	6	3	1	-
Total		288	2585	337	195	17	13	16

There are no specific rules in the interview for determining the sequence of responses. However, the SBVR operational rule requires a sequence of responses indicated as a procedure. The ID2SBVR needs to identify the sequence words of each response to represent the sequence of the fact type. Furthermore, not every response has a sequence word, but it may have a candidate fact type. The ID2SBVR indicates the transitive verb in those responses with the extracted triplet (see Section 3.2.1). The sequence word identifies fact types and the order between fact types, not the sequence in SBVR.

4.3. Extracting Fact Type

In this experiment, the accuracy of the ID2SBVR-extracted fact type shows an average of 0.98. Table 3 presents the final calculation of the precision, recall, and accuracy collected from automatic extracting of fact type using dataset_university1. In the experimental results, the overall average values obtained are 0.98 in terms of precision, recall or sensitivity, specificity, and accuracy.

Table 3. Experimental results of extracting fact type in dataset_university1.

Question ID	Precision	Recall	Specificity	Accuracy
q1	-	-		-
q2	1.00	1.00	1.00	1.00
q3	0.83	0.83	0.83	0.83
q4	-	-		-
q5	1.00	1.00	1.00	1.00
q6	0.88	0.88	0.88	0.88
q7	1.00	1.00	1.00	1.00
q8	1.00	1.00	1.00	1.00
q9	1.00	1.00	1.00	1.00
q10	1.00	1.00	1.00	1.00
q11	1.00	1.00	1.00	1.00
q12	1.00	1.00	1.00	1.00
q13	1.00	1.00	1.00	1.00
q14	1.00	1.00	1.00	1.00
q15	1.00	1.00	1.00	1.00
q16	1.00	1.00	1.00	1.00
Average	0.98	0.98	0.98	0.98

The maximum accuracy value is 1.00 because the fact type is contracted according to the fact type benchmark. The q3 has minimum accuracy value of 0.83 because in q3 there are two fact types detected incorrectly from a total of seven fact types. An error in q3 occurs where the detected fact type combines two fact types with the conjunction 'or'.

fact type: librarian submits into the circulation or librarian submits into reference sub section afterwards.

The results of the fact type extraction should occur by splitting the sentence:

fact type: librarian submits into the circulation.
fact type: librarian submits into reference sub section afterwards.

Furthermore, the ID2SBVR accuracy in extracted fact type using dataset_university2 shows an average value of 0.93. Table 4 presents the precision, recall, and accuracy of ID2SBVR using dataset_university2. In the experimental results, the overall average values obtained are precision 0.98, recall 0.91, specificity 0.98, and accuracy 0.95.

Table 4. Experimental results of extracting fact type in dataset_university2.

Question ID	Precision	Recall	Specificity	Accuracy
r1	-	-	-	-
r2	1.00	1.00	1.00	1.00
r3	1.00	1.00	1.00	1.00
r4	1.00	1.00	1.00	1.00
r5	1.00	1.00	1.00	1.00
r6	1.00	1.00	1.00	1.00
r7	1.00	1.00	1.00	1.00
r8	1.00	1.00	1.00	1.00
r9	1.00	1.00	1.00	1.00
r10	1.00	1.00	1.00	1.00
r11	1.00	1.00	1.00	1.00
r12	0.86	0.75	0.88	0.81
r13	1.00	0.80	1.00	0.90
r14	1.00	0.86	1.00	0.93
r15	1.00	1.00	1.00	1.00
r16	1.00	0.88	1.00	0.94

Table 4. *Cont.*

Question ID	Precision	Recall	Specificity	Accuracy
r17	1.00	1.00	1.00	1.00
r18	1.00	1.00	1.00	1.00
r19	0.80	0.67	0.83	0.75
r20	1.00	1.00	1.00	1.00
r21	1.00	0.89	1.00	0.94
r22	1.00	1.00	1.00	1.00
r23	1.00	1.00	1.00	1.00
r24	1.00	0.81	1.00	0.91
r25	1.00	0.83	1.00	0.92
r26	1.00	1.00	1.00	1.00
r27	0.87	0.76	0.88	0.82
r28	0.83	0.63	0.88	0.75
r29	1.00	1.00	1.00	1.00
r30	0.95	0.83	0.96	0.89
r31	1.00	0.77	1.00	0.88
r32	1.00	0.67	1.00	0.83
r33	1.00	1.00	1.00	1.00
Average	0.96	0.89	0.98	0.93

The maximum accuracy values occur in r2–r11, r15, r18, r20, r22, r23, r26, r29, and r33 because the fact type is calculated according to the fact type benchmark. The r19 and r28 data points display a minimum accuracy value of 0.75. In fact, in r19, two fact types were detected incorrectly and two fact types were missing. The incorrectly detected fact types indicate that there is no subject in the sentence and the missing fact types indicate that the sentence's verb is not recognized by ID2SBVR. The two missing sentences are:

'First, member hands over the book and receipt'.
'Next, staff files borrowing receipts to its shelf'.

The results of the fact type extraction should be:

'member hands over the book and receipt'.
'staff files borrowing receipts to its shelf'.

The other minimum accuracy value occurs in r28 because there are two missing fact types. As previously, the missing fact types indicate that the verb and subject of the sentence are not recognized by ID2SBVR. The two sentences that are missing are:

'then, student scans the id card barcode'.
'if student chooses to save the file, then the file will be saved on the storage device, else prints the file'.

The results of the fact type extraction should be:

fact type: 'student scans the id card barcode'.
fact type: 'student chooses to save the file'.
fact type: 'the file will be saved on the storage device'.
fact type: 'student prints the file'.

In comparison with the study by Lopez et al. [2], their prototype for the extraction of business process elements in natural language text showed precision, recall, and accuracy values of 0.92, 0.84, and 0.88, respectively, based on a set of 56 texts. The advantage of ID2SBVR is that natural language is first extracted to SBVR which becomes standard English. This facilitates the transformation of the SBVR into BPMN. The research by Arshad et al. [18] used an approach that transformed SBVR to XML; this approach displayed an average recall value of 0.89 while the average precision value was 0.96.

Based on the results of performance calculations shows in Figure 7, from both datasets, analysis of the university1 dataset yielded higher precision, recall, and accuracy values than university2, while both datasets generated the same specificity value. The number of different data points in the dataset does not significantly affect the performance of this method. However, grammatical errors will increase false positives (FP) and false negatives (FN) which markedly affects the accuracy of this method.

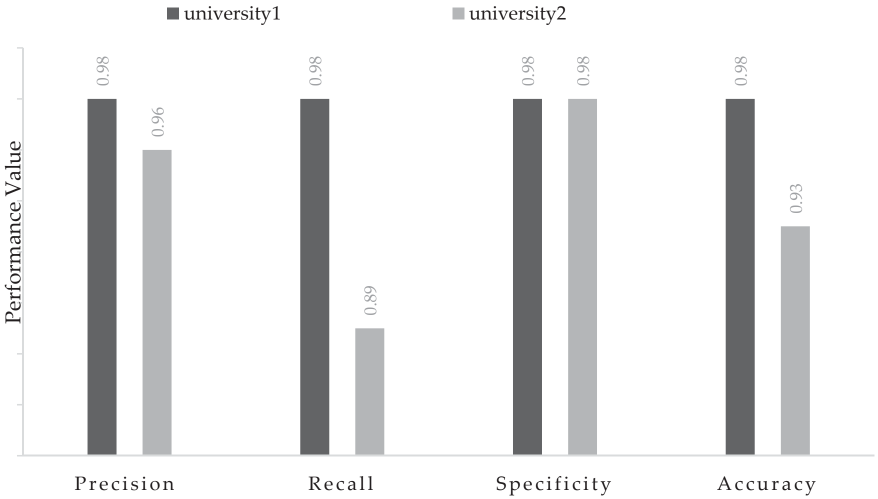


Figure 7. Graphical representations of precision, recall, specificity, and accuracy of ID2SBVR-extracted fact type.

4.4. Generating SBVR

The total operational rules generated by ID2SBVR was 103 operational rules for dataset_university1 and 209 operational rules for dataset_university2. Accuracy in generating SBVR is calculated for each process by comparing the number of SBVR detected correctly with the total SBVR generated. In dataset_university1, 99 SBVR were was correctly detected (96.2% of the total SBVR), while there were four SBVR with errors (3.8% of the total SBVR). The average accuracy value for generating SBVR using ID2SBVR was 0.94, as shown in Table 5. For the processed q3 and q6, there were two SBVR errors generated from a total of five SBVR. The errors in q3 and q6 were both caused by incorrect fact types. In terms of the variance of the ID2SBVR accuracy data, in dataset_university1 the value was 0.02 with a standard deviation [44] of 0.15.

Table 5. Experimental results of generating SBVR in dataset_university1.

Question ID	SBVR	Error Due to Wrong Fact Type	Error Due to Wrong Sequencing	Accuracy
q1	-	-	-	-
q2	11	-	-	1.00
q3	5	2	-	0.60
q4	-	-	-	-
q5	3	-	-	1.00
q6	5	2	-	0.60
q7	4	-	-	1.00
q8	4	-	-	1.00
q9	6	-	-	1.00
q10	8	-	-	1.00
q11	5	-	-	1.00
q12	8	-	-	1.00
q13	10	-	-	1.00
q14	13	-	-	1.00

Table 5. *Cont.*

Question ID	SBVR	Error Due to Wrong Fact Type	Error Due to Wrong Sequencing	Accuracy
q15	15	-	-	1.00
q16	11	-	-	1.00
			Average	0.94
			Variance	0.02
			Standard Deviation	0.15

In the dataset_university2, SBVR correctly detected 186 SBVR or 89% of the total SBVR, with an 11% error rate. The errors were caused by wrong fact types or missing fact types causing wrong sequencing. The average accuracy of ID2SBVR-generated SBVR was 0.88, as shown in Table 6. The statistical analysis of the ID2SBVR accuracy values in the university2 dataset shows very similar results to the previous dataset, namely, a variance value of 0.03 and a standard deviation [44] of 0.18.

Table 6. Experimental results of generating SBVR in dataset_university2.

Question ID	SBVR	Error Due to Wrong Fact Type	Error Due to Wrong Sequencing	Accuracy
r1	-	-	-	-
r2	7	-	-	1.00
r3	6	-	-	1.00
r4	6	-	-	1.00
r5	6	-	-	1.00
r6	3	-	-	1.00
r7	9	-	-	1.00
r8	3	-	-	1.00
r9	4	-	-	1.00
r10	6	-	-	1.00
r11	5	-	-	1.00
r12	5	-	2	0.60
r13	3	-	1	0.67
r14	5	-	1	0.80
r15	6	-	-	1.00
r16	5	-	-	1.00
r17	5	-	-	1.00
r18	3	-	-	1.00
r19	5	2	1	0.40
r20	18	-	-	1.00
r21	7	-	1	0.86
r22	8	-	-	1.00
r23	5	-	-	1.00
r24	12	-	3	0.75
r25	5	1	1	0.60
r26	7	-	-	1.00
r27	11	-	2	0.82
r28	4	1	1	0.50
r29	7	-	-	1.00
r30	13	2	2	0.69
r31	7	-	1	0.86
r32	3	-	1	0.67
r33	10	-	-	1.00
			Average	0.88
			Variance	0.03
			Standard deviation	0.18

The SBVR extraction from UML (M2M) result in Skersys, Danenas, and Butleris (2018) has an accuracy value for the original model of 0.70 and for the refactored model of 0.97. The accuracy value of M2M is higher because the SBVR is extracted from UML with an existing and clear structure. It differs from ID2SBVR where the initial data was in the form of natural language with an irregular language structure.

To measure the variance of data distribution related to ID2SBVR accuracy in generating SBVR, the standard deviation of the existing accuracy is determined. When the standard deviation [44] value is low, the extent of the variability in the accuracy values falls within a close range in all processes. Therefore, the ID2SBVR-generated SBVR results are close to each other and do not display any marked deviations. This was the case in both datasets, where the standard deviations of accuracy values of ID2SBVR were 0.15 and 0.18, respectively.

Figure 8 shows the accuracy of dataset_university1 and dataset_university2. ID2SBVR-generated SBVR data are strongly influenced by the fact type extraction results; therefore, differences in accuracy may also occur. Dataset_university2 has an accuracy value 0.06 lower than dataset_university1 because fact type errors and missing fact types occur more commonly than in dataset_university1.

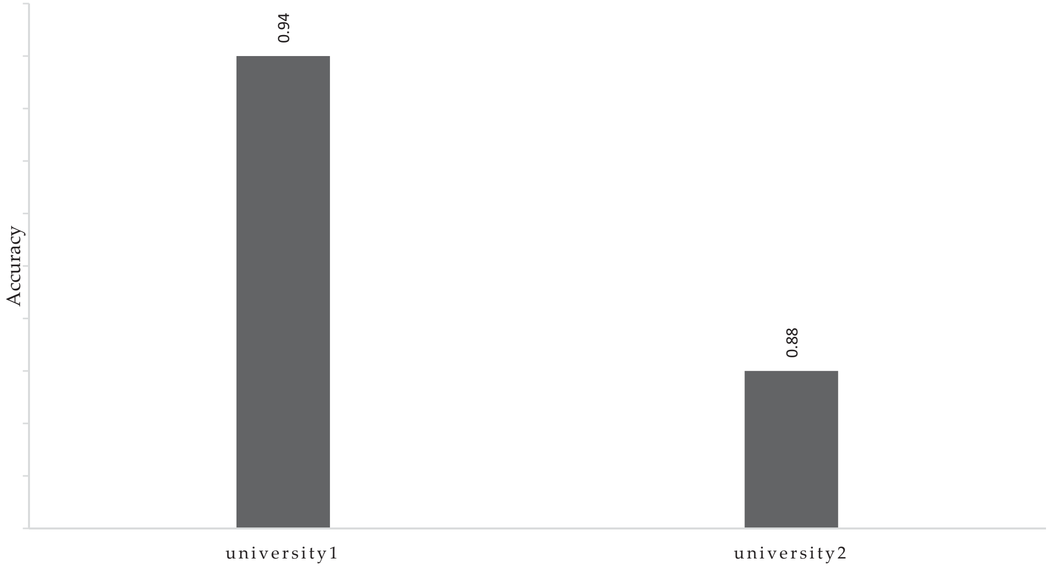


Figure 8. Graphical representations of the accuracy of ID2SBVR generating SBVR.

5. Threat to Validity

In this study, the threat to validity lies in two phenomena, namely (1) processes that are not mentioned in the interview and (2) processes referred to in interviews that are not sequentially described by the respondent. The validity of the ID2SBVR method has not been tested for the two phenomena above, because the dataset does not contain these phenomena. By design, the ID2SBVR method can be used in different business domains because this method is theoretically independent of domain.

The ID2SBVR method does not depend on the corpus of a particular domain. Although the test results of the ID2SBVR method show very good results in the PPT domain, it has not been tested in other domains. In addition, the dataset that is built is assumed to have a standard grammar. For this reason, in the dataset acquisition process, there is a correction process for meaningless words and for repeated sentences. The complexity of the resulting business processes is not limited. It has fulfilled all existing gateways in BPMN.

6. Conclusions

The ID2SBVR presents a new approach for extracting fact types from informal documents. The ID2SBVR allows a business process designer to translate natural language from an interview document into operational rules in SBVR, which in turn can be transformed into BPMN. The novelty of ID2SBVR is that it uses informal documents as a substitute for the formal documents that usually are required by BPMN. The informal documents are the result of an open-ended interview. The data are formed from irregularly structured natural language.

The ID2SBVR succeeds in SBVR operational rule extraction from informal documents on the basis of sentence extraction relevant to SBVR and its sequence. The unstructured data is successfully converted into semi-structured data for use in the pre-processing. The ID2SBVR method translates informal documents that are unstructured into structured ones with a high accuracy value of 0.91. The standard deviation of the ID2SBVR accuracy value in each process is 0.17. The ID2SBVR accuracy value does not show any large data deviations. The ID2SBVR method succeeded in extracting the types of facts including compound, complex, and complex-compound sentences, with an average value of 0.91 for precision and recall, and an almost perfect accuracy of 0.97.

This study has both theoretical and practical implications. Theoretically, this study complements linguistic studies relating to business vocabulary and business rules [9,10,16,25,39] and information retrieval [36,45,46]. Fact types can be extracted based on sequence words or from POS tags. The actor of a fact type with noun phrase can also be extracted. Thus, our research contributes to the extracted user story aspect of what and aspect of who. This new method establishes SBVR using datasets obtained from interview documents. This method succeeds in mining fact type candidates and generating SBVR from informal documents with almost perfect accuracy. The extraction results carried out by ID2SBVR will be used for transformation to a process model using the XML Process Definition Language (XPDL).

The current research has various practical implications, namely: (i) making it easier to understand organizational processes because they are presented in BPMN; (ii) enabling employee assessment of business processes, as BPMN makes them easier to understand; (iii) recommendations for business process improvements within organizations—initial BPMN documentation can be a basic consideration for submitting the re-engineering process; (iv) automatic transformation from SBVR to BPMN; and (v) the method can be used as a basis for comparative analysis between formal BP and actual implementation.

Our future work will focus on extending the method to detect whether there are missing processes, non-sequential processes, or conflicts because of different information obtained from more than one process. The next step would be to perform transformation from SBVR to BPMN by extending the process with another gateway in the BPMN.

Author Contributions: Conceptualization, I.T., R.S. and D.S.; methodology, I.T., R.S. and D.S.; validation, I.T.; formal analysis, I.T., R.S. and D.S; investigation, I.T.; data curation, I.T.; writing—original draft preparation, I.T.; writing—review and editing, R.S. and D.S; supervision, R.S. and D.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data available in a publicly accessible repository: the data presented in this study are openly available in FigShare at https://doi.org/10.6084/m9.figshare.15123879.v1 (accessed on 27 January 2022) and https://doi.org/10.6084/m9.figshare.15123972.v2 (accessed on 27 January 2022).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Skersys, T.; Danenas, P.; Butleris, R. Model-Based M2M Transformations Based on Drag-and-Drop Actions: Approach and Implementation. *J. Syst. Softw.* **2016**, *122*, 327–341. [CrossRef]
2. Lopez, H.A.; Marquard, M.; Muttenthaler, L.; Stromsted, R. Assisted Declarative Process Creation from Natural Language Descriptions. In Proceedings of the IEEE International Enterprise Distributed Object Computing Workshop, EDOCW, Paris, France, 28–31 October 2019; pp. 96–99. [CrossRef]

3. Deng, J.; Deng, X. Research on the Application of Data Mining Method Based on Decision Tree in CRM. *J. Adv. Oxid. Technol.* **2018**, *21*, 20–28. [CrossRef]
4. Praveen, S.; Chandra, U. Influence of Structured, Semi-Structured, Unstructured Data on Various Data Models. *Int. J. Isc. Eng. Res.* **2017**, *8*, 67–69.
5. Baig, M.I.; Shuib, L.; Yadegaridehkordi, E. Big Data Adoption: State of the Art and Research Challenges. *Inf. Process. Manag.* **2019**, *56*, 102095. [CrossRef]
6. McConnell, C.R.; Fallon, F.L. *Human Resource Management in Health Care*; Jones & Bartlett Publishers: Burlington, MA, USA, 2013.
7. Tangkawarow, I.; Sarno, R.; Siahaan, D. Modeling Business Rule Parallelism by Introducing Inclusive and Complex Gateways in Semantics of Business Vocabulary and Rules. *Int. J. Intell. Eng. Syst.* **2021**, *14*, 281–295. [CrossRef]
8. Mickeviciute, E.; Butleris, R.; Gudas, S.; Karciauskas, E. Transforming BPMN 2.0 Business Process Model into SBVR Business Vocabulary and Rules. *Inf. Technol. Control* **2017**, *46*, 360–371. [CrossRef]
9. Mickeviciute, E.; Skersys, T.; Nemuraite, L.; Butleris, R. SBVR Business Vocabulary and Rules Extraction from BPMN Business Process Models. In Proceedings of the 8th IADIS International Conference Information Systems 2015, IS 2015, Funchal, Portugal, 4–16 March 2015; pp. 211–215.
10. Danenas, P.; Skersys, T.; Butleris, R. Natural Language Processing-Enhanced Extraction of SBVR Business Vocabularies and Business Rules from UML Use Case Diagrams. *Data Knowl. Eng.* **2020**, *128*, 101822. [CrossRef]
11. Skersys, T.; Danenas, P.; Butleris, R. Extracting SBVR Business Vocabularies and Business Rules from UML Use Case Diagrams. *J. Syst. Softw.* **2018**, *141*, 111–130. [CrossRef]
12. Mishra, A.; Sureka, A. A Graph Processing Based Approach for Automatic Detection of Semantic Inconsistency between BPMN Process Model and SBVR Rules. In *Mining Intelligence and Knowledge Exploration*; Lecture Notes in Computer Science (including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics); Springer: Berlin/Heidelberg, Germany, 2015; Volume 9468, pp. 115–129. [CrossRef]
13. Dumas, M.; La Rosa, M.; Mendling, J.; Reijers, H.A. *Fundamentals of Business Process Management*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 8–10. [CrossRef]
14. Leon, R.D.; Rodríguez-Rodríguez, R.; Gómez-Gasquet, P.; Mula, J. Business Process Improvement and the Knowledge Flows That Cross a Private Online Social Network: An Insurance Supply Chain Case. *Inf. Process. Manag.* **2020**, *57*, 102237. [CrossRef]
15. Kluza, K.; Nalepa, G.J. Formal Model of Business Processes Integrated with Business Rules. *Inf. Syst. Front.* **2019**, *21*, 1167–1185. [CrossRef]
16. Kluza, K.; Nalepa, G.J. A Method for Generation and Design of Business Processes with Business Rules. *Inf. Softw. Technol.* **2017**, *91*, 123–141. [CrossRef]
17. Aiello, G.; Di Bernardo, R.; Maggio, M.; Di Bona, D.; Re, G.L. Inferring Business Rules from Natural Language Expressions. In Proceedings of the IEEE 7th International Conference on Service-Oriented Computing and Applications, SOCA 2014, Matsue, Japan, 17–19 November 2014; pp. 131–136. [CrossRef]
18. Arshad, S.; Bajwa, I.S.; Kazmi, R. Generating SBVR-XML Representation of a Controlled Natural Language. In *Communications in Computer and Information Science*; Springer: Singapore, 2019; Volume 932, pp. 379–390. [CrossRef]
19. Akhtar, B.; Mehmood, A.A.; Mehmood, A.A.; Noor, W. Generating RDFS Based Knowledge Graph from SBVR. In *Intelligent Technologies and Applications. INTAP 2018. Communications in Computer and Information Science*; Springer: Singapore, 2019; Volume 932, pp. 618–629. [CrossRef]
20. Mohanan, M.; Samuel, P. Natural Language Processing Approach for UML Class Model Generation from Software Requirement Specifications via SBVR. *Int. J. Artif. Intell. Tools* **2018**, *27*, 6. [CrossRef]
21. Skersys, T.; Kapocius, K.; Butleris, R.; Danikauskas, T. Extracting Business Vocabularies from Business Process Models: SBVR and BPMN Standards-Based Approach. *Comput. Sci. Inf. Syst.* **2014**, *11*, 1515–1536. [CrossRef]
22. Tantan, O.; Akoka, J. Automated Transformation of Business Rules into Business Processes From SBVR to BPMN. In Proceedings of the International Conference on Software Engineering and Knowledge Engineering, SEKE, Vancouver, BC, Canada, 1–3 July 2014; Volume 25, pp. 684–687.
23. Kluza, K.; Kutt, K.; Wozniak, M. SBVRwiki (Tool Presentation). *CEUR Workshop Proc.* **2014**, *1289*, 703–713. [CrossRef]
24. Kluza, K.; Honkisz, K. From SBVR to BPMN and DMN Models. Proposal of Translation from Rules to Process and Decision Models. In *Artificial Intelligence and Soft Computing: 15th International Conference, ICAISC 2016*; Lecture Notes in Computer Science; Springer: Zakopane, Poland, 2016; Volume 9693, pp. 453–462. [CrossRef]
25. Rodrigues, R.D.A.; Azevedo, L.G.; Revoredo, K.C. BPM2Text: A Language Independent Framework for Business Process Models to Natural Language Text. *ISys-Braz. J. Inf. Syst.* **2016**, *10*, 38–56. [CrossRef]
26. Ferreira, R.C.B.; Thom, L.H.; Fantinato, M. A Semi-Automatic Approach to Identify Business Process Elements in Natural Language Texts. In Proceedings of the ICEIS 2017: 19th International Conference on Enterprise Information Systems, Porto, Portugal, 26–29 April 2017; Volume 3, pp. 250–261. [CrossRef]
27. Delicado, L.; Sànchez-Ferreres, J.; Carmona, J.; Padró, L. NLP4BPM—Natural Language Processing Tools for Business Process Management. *CEUR Workshop Proc.* **2017**, *1920*, 1–5.
28. Iqbal, U.; Bajwa, I.S. Generating UML Activity Diagram from SBVR Rules. In *2016 6th International Conference on Innovative Computing Technology, INTECH 2016*; IEEE Xplore: Dublin, Ireland, 2017; pp. 216–219. [CrossRef]
29. Mohanan, M.; Bajwa, I.S. Requirements to Class Model via SBVR. *Int. J. Open Source Softw. Process.* **2019**, *10*, 70–87. [CrossRef]

30. Bazhenova, E.; Zerbato, F.; Oliboni, B.; Weske, M. From BPMN Process Models to DMN Decision Models. *Inf. Syst.* **2019**, *83*, 69–88. [CrossRef]
31. da Purificação, C.E.P.; da Silva, P.C. A Controlled Natural Language Editor for Semantic of Business Vocabulary and Rules. In *Advances in Intelligent Systems and Computing*; Springer: Berlin/Heidelberg, Germany, 2018; Volume 738, pp. 499–508. [CrossRef]
32. Object Management Group; OMG; Object Management Group. Semantics of Business Vocabulary and Business Rules Version 1.0. OMG Document Number: Formal/19-10-02. 2019. Available online: https://www.omg.org/spec/SBVR/1.5/PDF (accessed on 20 January 2021).
33. Object Management Group. Semantics of Business Vocabulary and Business Rules V.1.5 Annex H- The RuleSpeak Business Rule Notation. OMG Document Number: Formal/19-10-07. 2019. Available online: https://www.omg.org/spec/SBVR/1.5/About-SBVR/ (accessed on 23 January 2021).
34. Business Rule Solutions, L.; RuleSpeak. Business Rule Solutions, LLC. Available online: http://www.rulespeak.com/en/ (accessed on 12 March 2021).
35. Aluthman, E.S. A Cross-Disciplinary Investigation of Textual Metadiscourse Markers in Academic Writing. *Int. J. Linguist.* **2018**, *10*, 19–38. [CrossRef]
36. Altınel, B.; Ganiz, M.C. Semantic Text Classification: A Survey of Past and Recent Advances. *Inf. Process. Manag.* **2018**, *54*, 1129–1153. [CrossRef]
37. Wang, Y. A Functional Analysis of Text-Oriented Formulaic Expressions in Written Academic Discourse: Multiword Sequences vs. Single Words. *Engl. Specif. Purp.* **2019**, *54*, 50–61. [CrossRef]
38. Uysal, A.K.; Gunal, S. The Impact of Preprocessing on Text Classification. *Inf. Process. Manag.* **2014**, *50*, 104–112. [CrossRef]
39. Steven, B.; Loper, E.; Klein, E. *Natural Language Processing with Python**; O'Reilly Media Inc.: Sebastopol, CA, USA, 2009.
40. Demirezen, M. Determining the Intonation Contours of Compound-Complex Sentences Uttered by Turkish Prospective Teachers of English. *Procedia-Soc. Behav. Sci.* **2015**, *186*, 274–282. [CrossRef]
41. Stapor, K.; Ksieniewicz, P.; García, S.; Woźniak, M. How to Design the Fair Experimental Classifier Evaluation. *Appl. Soft Comput.* **2021**, *104*, 107219. [CrossRef]
42. Pereira, R.B.; Plastino, A.; Zadrozny, B.; Merschmann, L.H.C. Correlation Analysis of Performance Measures for Multi-Label Classification. *Inf. Process. Manag.* **2018**, *54*, 359–369. [CrossRef]
43. Darma, I.W.A.S.; Suciati, N.; Siahaan, D. Neural Style Transfer and Geometric Transformations for Data Augmentation on Balinese Carving Recognition Using MobileNet. *Int. J. Intell. Eng. Syst.* **2020**, *13*, 349–363. [CrossRef]
44. Mauro, N.; Ardissono, L.; Petrone, G. User and Item-Aware Estimation of Review Helpfulness. *Inf. Process. Manag.* **2021**, *58*, 102434. [CrossRef]
45. Joshi, B.; Macwan, N.; Mistry, T.; Mahida, D. Text Mining and Natural Language Processing in Web Data Mining. In Proceedings of the 2nd International Conference on Current Research Trends in Engineering and Technology, Gujarat, India, 9 April 2018; Volume 4.
46. Rashid, J.; Shah, S.M.A.; Irtaza, A. Fuzzy Topic Modeling Approach for Text Mining over Short Text. *Inf. Process. Manag.* **2019**, *56*, 102060. [CrossRef]

big data and cognitive computing

MDPI

Article

Ontology-Based Personalized Job Recommendation Framework for Migrants and Refugees

Dimos Ntioudis [1,*], Panagiota Masa [1], Anastasios Karakostas [2], Georgios Meditskos [1,3], Stefanos Vrochidis [1] and Ioannis Kompatsiaris [1]

[1] Centre for Research & Technology Hellas, Information Technologies Institute, 6th Km Charilaou—Thermi, 57001 Thessaloniki, Greece
[2] Draxis Environmental, 54655 Thessaloniki, Greece
[3] School of Informatics, Aristotle University of Thessaloniki, 54124 Thessaloniki, Greece
* Correspondence: ntdimos@iti.gr

Abstract: Participation in the labor market is seen as the most important factor favoring long-term integration of migrants and refugees into society. This paper describes the job recommendation framework of the Integration of Migrants MatchER SErvice (IMMERSE). The proposed framework acts as a matching tool that enables the contexts of individual migrants and refugees, including their expectations, languages, educational background, previous job experience and skills, to be captured in the ontology and facilitate their matching with the job opportunities available in their host country. Profile information and job listings are processed in real time in the back-end, and matches are revealed in the front-end. Moreover, the matching tool considers the activity of the users on the platform to provide recommendations based on the similarity among existing jobs that they already showed interest in and new jobs posted on the platform. Finally, the framework takes into account the location of the users to rank the results and only shows the most relevant location-based recommendations.

Keywords: recommendation systems; job matching; ontologies; reasoning; migrants; refugees

Citation: Ntioudis, D.; Masa, P.; Karakostas, A.; Meditskos, G.; Vrochidis, S.; Kompatsiaris, I. Ontology-Based Personalized Job Recommendation Framework for Migrants and Refugees. *Big Data Cogn. Comput.* **2022**, *6*, 120. https://doi.org/10.3390/bdcc6040120

Academic Editors: Konstantinos Kotis and Dimitris Spiliotopoulos

Received: 29 July 2022
Accepted: 8 October 2022
Published: 19 October 2022

1. Introduction

The issue of whether migrants and refugees are able to contribute to the economic growth and success of their hosts has become an increasingly hot topic. While it is widely accepted that labour mobility contributes positively to economies and ultimately welfare standards and reduces unemployment, it is not a given that migrants are afforded the same opportunity to find employment, become self-employed, finance their own living standards and contribute positively to their hosts' economies [1]. Participation in the labour market is seen as the most important factor favouring long-term integration into society. In service of this goal, Information and Communication Technology (ICT) tools play a vital role in the integration and social inclusion of migrants and refugees [2–5]. They can be used to support migrants in several activities and to overcome the difficulties they might encounter when they come in the destination country, ranging from learning the language of the new country, acquiring job-related skills, accessing education and job opportunities, assimilating with the wider community, and so on.

In this paper, we present the job recommendation framework of the IMMERSE platform. IMMERSE is a web-based platform that was designed and developed in the context of a European project called ICT Enabled Public Services for Migration (MIICT) [6]. The proposed framework acts as a matching tool that enables the contexts of individual migrants and refugees, including their resume and preferences to facilitate their matching with the job opportunities available in their host country. An overview of the architecture and the core components of IMMERSE was presented in [7]. The main objective of IMMERSE is to encourage migrants to overcome significant difficulties they face when they arrive in the host country. This is achieved by offering a variety of services and information that can help migrants integrate with the wider population. For example, migrants can use an IMMERSE platform for finding employment, learning a new language, finding courses to improve their skills, obtaining volunteering assistance and finding social activities. In addition, a preliminary version of the ontology-based personalized job recommendation framework was presented in [8].

Here, we significantly extend our previous approach by: (a) updating the ontology to also capture information about the skills of a user, the job preferences of the user as well as the interaction of the user with existing posts, and (b) extending the matchmaking service with new rules that consider the aforementioned additions in the ontology. The main contributions of this paper can be summarised as follows: (a) the final version of the knowledge representation model of IMMERSE that is capable of semantically representing the Curriculum Vitae (CV) of a job seeker details and the details of a job posting and (b) an ontology-based matchmaking service that provides relevant job recommendations considering the full CV of a job seeker and the details of the available jobs.

The rest of the paper is organized as follows: In Section 2, we describe related work. In Section 3, we present a brief overview of the IMMERSE architecture and the main technical components. Section 4 describes the IMMERSE Ontology and the semantic representation of basic notions. In Section 5, the Semantic Reasoning is presented. Section 6 demonstrates the matchmaking mechanism through a simulation example. Section 7 presents the evaluation results. Finally, Section 8 concludes the paper.

2. Background and Related Work

Ontologies are important in the Semantic Web because they provide a mechanism to deal with heterogeneous representations of web resources. They are a formal way of representing knowledge within a topic of interest, consisting of a range of ideas, their attributes, and the relationships between them. Ontologies can be used to specify various classes, properties, and relationships, as well as rules, axioms, and restrictions. In addition, to derive implicit knowledge hidden into metadata and discover inferences based on asserted information in the ontology, reasoning systems are used. Reasoners are systems that can handle and apply the semantics and can be used to make logical inferences. In particular, ***rule-based*** reasoning is used to create new information in a controlled way, based on very specific and clearly defined rules. The following subsections provide a brief overview of existing ontologies and standards that will form the basis of the IMMERSE ontology and reasoning.

2.1. Knowledge Representation

The use of ontologies requires an ontology language that is well-designed, well-defined, and web-compatible, as well as supporting reasoning tools. The Resource Description Framework (RDF) is a simple knowledge representation language that aims to standardize metadata and descriptions of Web-based resources definition and use [9]. The basic building block in RDF is a set of ***subject-predicate-object*** triples. Moreover, the Web Ontology Language (OWL) is a set of knowledge representation languages that is commonly used to create ontologies and was created to display rich and more complex information about things, groups of things, and relationships between them [10]. OWL has become the official World Wide Web Consortium (W3C) recommendation [11] for authoring and sharing ontologies and extends existing Web standards for representing knowledge such as the RDF, XML, etc.

2.2. User Profile Ontologies

Friend of a Friend (FOAF) [12] is an ontology that captures the fundamental characteristics of a person, e.g., name, gender, date of birth, location and aims to describe people as well as the relationships that connect people, places, and other objects. Similarly, a set of ontologies called the Core Vocabularies [13] encapsulate the core properties of an entity, such as a person or a government agency. The Core Vocabularies were created in response to the requirement for semantic interoperability in the context of European public services, as well as the lack of universally agreed data models and other semantic interoperability-related disputes. These vocabularies are context-neutral data models that are simple, reusable, and expandable.

2.3. Ontologies Relevant to E-Recruitment, Work Experience and Jobs

Job Description ontology [14] is used to describe the semantic structure for defining a job position and to provide a common ground for sharing knowledge. It specifies details about the job title, requirements, responsibilities, education, post date, last-apply date, organization name, etc. Description of a Career (DOAC) [15] is a vocabulary for describing a worker's professional skills. In addition, Ontology for Skill and Competency Management [16] facilitates the management of available human resources' skills and competencies. This ontology assists managers in knowing the competencies of available human resources and matching them with the existing requirements. ResumeRDF [17] is a Semantic Web ontology for expressing information contained in a personal resume or Curriculum Vitae (CV). This includes a greater number of properties and covers details such as work and academic experience, skills, certifications, courses attended and other relevant information. Finally, the Human Resources Management Ontology [18] describes the details of a job posting and the CV of a job seeker by acting as a common "language" in the form of a set of controlled vocabularies. This ontology is composed of thirteen modular ontologies: Competence, Compensation, Driving License, Economic Activity, Education, Geography, Job Offer, Job Seeker, Labour Regulatory, Language, Occupation, Skill and Time.

2.4. Rule Languages

Simple Protocol and RDF Query Language (SPARQL) [19] is a declarative language for extracting and updating information from RDF graphs that is recommended by the W3C community. It is an expressive language that allows the creation of complex relations between different entities and contains capabilities for querying required and optional graph patterns along with their conjunctions and disjunctions. Apart from that, SPARQL can be used to describe rules for generating new RDF graphs by combining data from existing graphs into new ones and by using the CONSTRUCT graph pattern. CONSTRUCT determines which RDF triples should be added to the new graph based on a set of matching criteria applied to existing graphs (i.e., WHERE clause). The CONSTRUCT query form returns a single RDF graph defined by a graph template. The output is an RDF graph generated by substituting the variables in the graph template for each query solution in the solution sequence, then setting union to join the triples into a single RDF graph.

2.5. Recommendation Systems for Job Recruitment

Recommendation systems are frequently utilized to help users find products or services that best fit their individual preferences. Thus, a personalized recommendation system is a solution to the problem of information overload and widely applied in many domains. For this reason, many studies have been conducted and are presented in this section. Firstly, an approach to the Semantic Matchmaking for Job Recruitment is presented in [20]. The aim of the Semantic Matchmaking for Job Recruitment approach is to present a hybrid ontology-based strategy for effectively matching job seekers with job advertisements. For each job posting, they distinguish between nice-to-have skills and must-have skills and also define the level of proficiency for each competence. Similarly, the matchmaking framework proposed in this work considers both nice-to-have requirements and must-have requirements as well as different educational levels and language competency levels to appropriately rank candidates that equally match an open position and improve the final recommendations.

Another solution in the same direction is a recommendation approach in [21] that is recommending jobs to candidates based on their preference profiles which are in turn based on previous preference ratings. Our method also takes into account the preferences of the migrants, i.e., posts that users have marked as favorites or applied to in the past, to favor posts that are more similar during the final ranking. Two stages are carried out to determine whether two jobs are similar: (a) first, we assess the requirements, including the required language, education, and working conditions, and (b) we use text-based similarity algorithms to compare the titles and descriptions of two given jobs. A final similarity score is then computed by combining the results of the two phases.

In addition, Ref. [22] presents Proactive, which is a comprehensive job recommendation system that assists job seekers in a variety of ways in finding appropriate opening jobs. The system depends on a series of ontologies for collecting and storing job related metadata such as the job's category, required educational level, experience and position type (i.e., full time, part time, etc.). This set of metadata fields is also extended with geographic information. This allows for calculating distance-based similarity between users' geographical preferences and job locations. In order to provide jobs that fit a user's profile and are nearby them, our matchmaker additionally uses geographical information from both job posts and users to produce a weighted score that takes their distance into account.

Moreover, in [23], they present an intelligent recruitment platform that uses an ontology-based skill matching algorithm and a complex evaluation to help recruitment sites find the best candidates for specific job openings. Users can post and receive information about job openings by visiting websites. With this graphical ontology model, they calculate the semantic similarity between resumes and employer's requirement for skills and, in order to construct the shortest path's weights summation (semantic similarity) between the appointed pair of skill nodes, they use the classical Dijkstra's algorithm. As part of our strategy, we have created a lengthy list of skills that job seekers may use to enhance their resumes and that employers can utilize when posting job adverts. The matchmaker then takes into account the skills listed on both sides and determines a matching score that will be used to determine the final recommendations.

Furthermore, in [24], they applied a machine learning framework and statistical forecasting models to provide job recommendations. Their approach uses user data along with forecasting and ranking models in order to provide efficient matching between potential candidates and job postings. The authors of [25] presented an approach that aims to provide job recommendations via profile matching methods such as semantic matching, tree-based knowledge matching and query matching. The degree of profile similarity is determined by integrating these methods in accordance with representations of the attributes of both users and jobs. The authors of [26] presented a self-learning recommendation engine that auto-fills missing information in a user's resume by using semantic reasoning. By doing this, it helps to boost a job seeker's chances, having an incomplete resume, of getting more matched job opportunities. Moreover, in [27], they propose a feature selection method using actual data to argue on the significance of the attributes needed for job matching. After the attributes are identified, the proposed system is instructed to use a clustering method to compare the job seekers' profiles to the job postings made by potential employers. Finally, in [28], a system that integrates neural networks and fuzzy logic is presented. The system is trained based on a large set of historical data of unemployed people that were either rejected or approved at several jobs in the past. The process of matching an unemployed person with an offered job is then performed using inference techniques.

3. IMMERSE Architecture

This section provides a high level overview of the architectural design of the IMMERSE system. Figure 1 describes the logical design of the IMMERSE platform, which shows how the different components are organized. The platform is comprised of a number of components, each of which has a specific function to perform. These are the following: (a) the ***Data Management System*** (DMS) component, which is in charge of efficiently storing heterogeneous data, (b) the ***User Interface*** (UI), which serves as the system's primary entry point, (c) the ***Authentication component*** (AUTH), which is used to secure the numerous operations carried out within the IMMERSE system, thereby enhancing data protection, (d) the ***Knowledge Base Service***, (KBS) which primarily accesses the IMMERSE ontology, and (e) a ***Message Bus*** (MSB), which serves as the primary channel for all intercomponent communication.

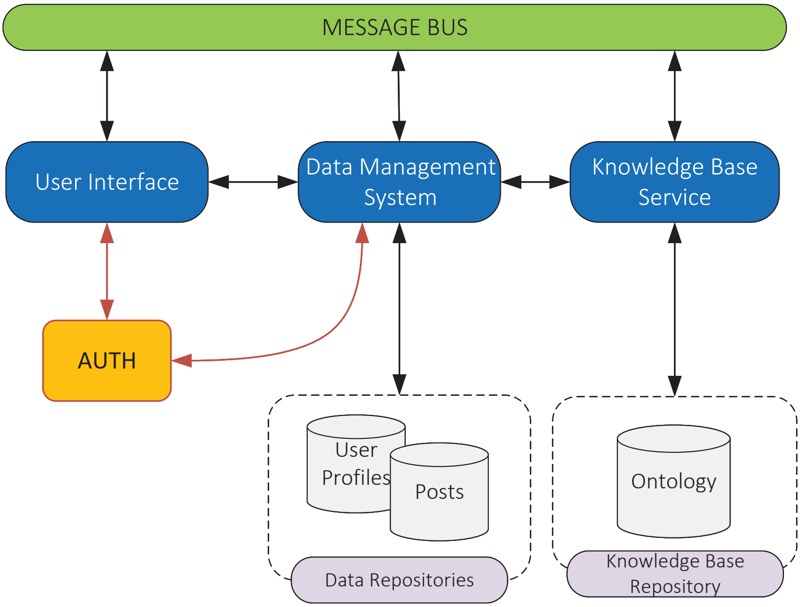


Figure 1. The IMMERSE architecture.

Semantic Framework

The ***Knowledge Base Service*** is a central component of the IMMERSE architecture. It is implemented in Java and acts as the main interface to the IMMERSE ***Knowledge Base*** (KB) also known as ontology, which is described in more detail in Section 4. KBS features subscribe capabilities to the MSB, through which it communicates with DMS for the exchange of information. Its purpose is to update DMS with reasoning results once they become available by its reasoning module.

More specifically, as shown in Figure 2, KBS encapsulates two modules: (a) the ***Knowledge Base Population*** component (KBP), which is responsible for integrating raw data that are collected from DMS, and semantically represent them to the IMMERSE ontology, and (b) the ***Semantic Reasoning*** component (SR) that consists of techniques, rules and algorithms that are performed on top of the IMMERSE domain ontology and aim at deriving new facts and relationships by combining existing knowledge.

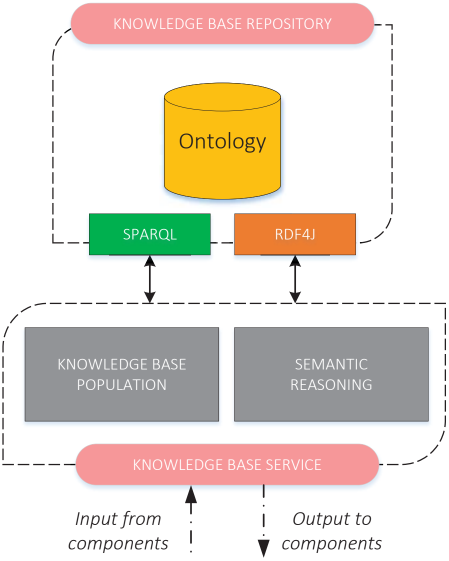


Figure 2. The semantic framework.

4. Ontology

The domain ontology is a lightweight knowledge representation model that is capable of semantically representing:

- *Basic user profile information*, such as location and contact details;
- *Extended user profile information*, including current and previous work experience, spoken languages, educational background, skills and work expectations;
- *Information about service providers*, including their professional profile, contact details as well as information about which services they offer;
- *Information about the actual listings* offered on the platform by its authorized service providers (i.e., job posts);
- *Information about the users' activity* on the platform including their applications and favourite posts.

Moreover, the ontology contains the analysis results that are generated based on the reasoning performed on top of the previous data. Such results contain:

- *Recommendation results* for registered migrants considering their profile and in-app activity as well as the post descriptions and requirements;
- *Similarity relations* between posts of the same category considering their title, description and the populated post attributes.

4.1. Representing User Profiles

The representation of the profile of a migrant user in the ontology is illustrated in Figure 3. More specifically, the ***MigrantProfile*** class refers to information relevant to the migrant's ***resume***. Five classes have been used to represent the migrant's resume (a) ***Language***, (b) ***Education***, (c) ***Work Experience***, (d) ***Skill*** and (e) ***Expectation***, each associated with a set of data properties as shown in the figure.

In addition, each migrant user can have several applications and several saved posts. This information is represented using the object properties ***hasAppliedPost*** and ***hasSavedPost*** respectively that point to an object of type ***Post*** in the ontology. For example, a migrant user can apply to or save posts of type ***Job***.

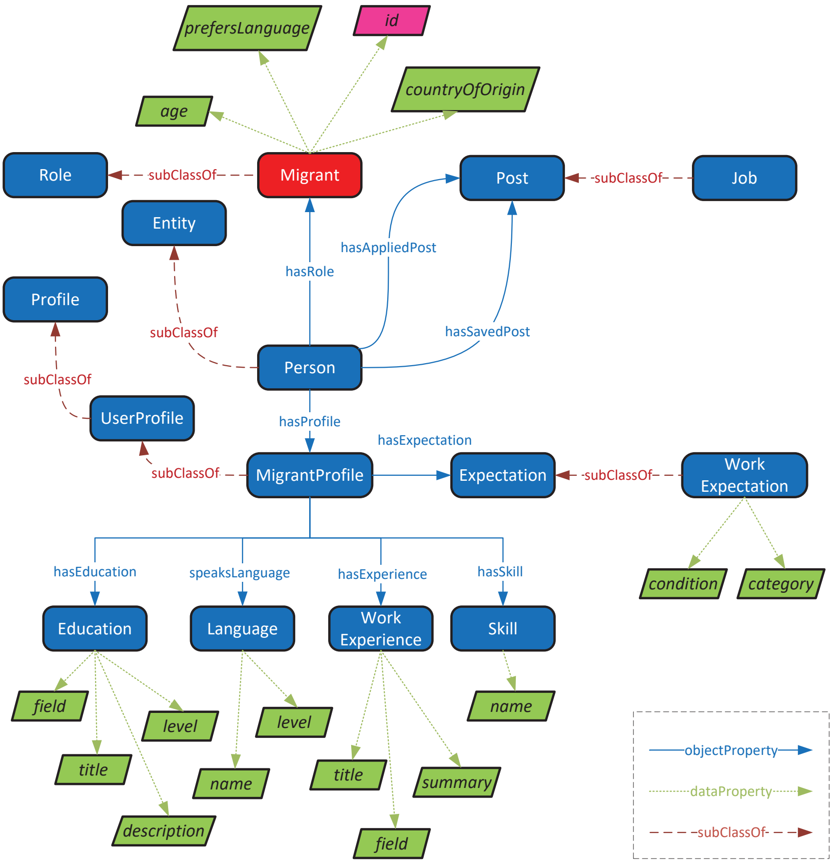


Figure 3. Migrant user representation.

4.2. Representing Job Listings

The class ***Job*** is used to represent a job that is posted in the platform. It is a subclass of the class ***Post*** from which it inherits two data properties, its ***title*** and ***description,*** while it has four other data properties dedicated to the class ***Job*** as depicted in Figure 4. Each job has a ***Category*** where several instances of this type are instantiated in the ontology. Finally, each ***Job*** is linked with other similar jobs through the property ***hasSimilarity*** that points to a class of type ***Similarity***. For each similar job, the ontology contains a property called ***score*** and an object property called ***pointsTo*** that points to a similar job in the domain ontology.

4.3. Representing Job Recommendations

Migrant users can receive recommendations based on the information that they have provided in their profiles. The class ***Job Recommendation*** in Figure 5 is a subclass of the class ***Recommendation***. Recommendations are generated and a score is calculated based on the user's (a) education (property ***education score***), (b) work experience (property ***work exp. score***), (c) spoken languages (property ***language score***), (d) expectations from using the platform, defined by users while creating their profile, such as finding a full time job under a specific category (property ***expectation score***), (e) distance from the job's location (property ***distance score***) and finally (f) a score based on the similarity of the given post to other posts that the user has either applied to or saved for later (property ***similarity score***).

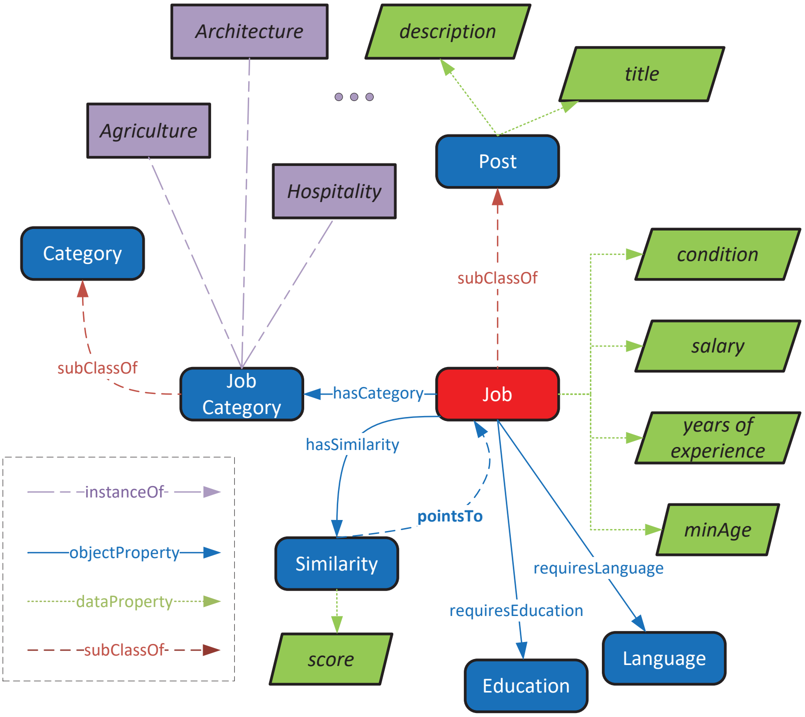


Figure 4. Job representation.

4.4. Ontology Validation

OntoMetrics [29], an online framework that validates ontologies based on established metrics, was used to assess the structure of the IMMERSE ontology. The results of OntoMetrics' analysis are presented in Table 1. Simple metrics, such as the count of classes, axioms, and objects, are included in ***Base Metrics***; these metrics show the number of ontology elements. ***Schema metrics***, on the other hand, are concerned with the ontology's design; metrics in this category indicate the ontology's richness, width, depth, and inheritance.

Starting with the base metrics, the total count of classes and properties indicates that the proposed ontology is a rather lightweight model. In addition, Description Logic (DL) expressivity refers to the Description Logic's variant the ontology belongs to. Since there are many varieties of DLs, there is an informal naming convention, roughly describing the operators allowed [30]. $SI^{(D)}$ expressivity of the IMMERSE ontology indicates a simple ontology, where S means that the ontology contains properties that have a transitive role, I means that the ontology contains inverse properties, and (D) refers to an ontology that uses datatype properties, data values or data types. Regarding schema metrics, the measurements in the table are adopted from [31,32]. More details on each metric are given below:

- ***Attribute richness*** is defined as the average number of attributes per class and can indicate both the quality of ontology design and the amount of information pertaining to instance data. The more attributes that are defined the more knowledge the ontology conveys;

- ***Inheritance richness*** is defined as the average number of subclasses per class and is a good indicator of how well knowledge is grouped into different categories and subcategories in the ontology;
- ***Relationship richness*** refers to the ratio of the number of non-inheritance relationships (i.e., object properties, equivalent classes, disjoint classes) divided by the total number of inheritance (i.e., subclass relations) and non-inheritance relationships defined in the ontology. This metric reflects the diversity of the types of relations in the ontology;
- Finally, ***axiom/class*** ratio, ***class/relation*** ratio, and ***inverse*** relations ratio describe the ratio between axioms–classes, classes–relations, and inverse relations–relations, respectively, and are indications of the ontology's transparency and understandability.

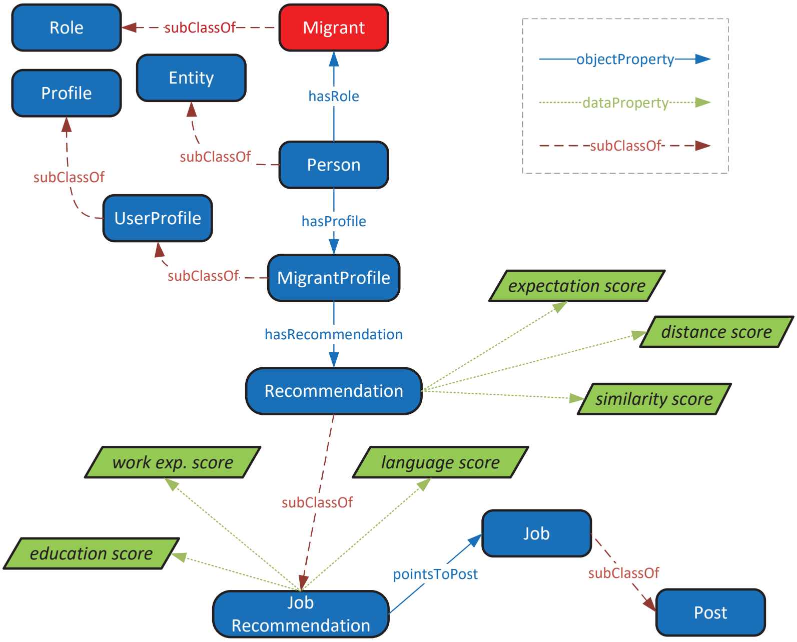


Figure 5. Recommendation representation.

Table 1. Ontology metrics for the IMMERSE ontology are generated by OntoMetrics.

Category	Metric	Value
Basic	Class Count	49
Basic	Object property count	25
Basic	Data property count	84
Basic	Description Logic expressivity	$SI^{(D)}$
Schema	Attribute richness	1.714
Schema	Inheritance richness	0.775
Schema	Relationship richness	0.479
Schema	Axiom/class ratio	9.040
Schema	Inverse relations ratio	0.071
Schema	Class/relation ratio	0.671

5. Semantic Reasoning

Semantic reasoning for IMMERSE is triggered by the Knowledge Base Service whenever new data become available in the system (e.g., when a new user profile is created, a new job is posted or any of the existing profiles and posts are updated). The main purpose of the semantic reasoning framework is to combine, integrate and semantically interpret the knowledge captured in the ontology and eventually extends the semantics of the system using rules and algorithms that are based on the available content, which further updates the knowledge captured in the semantic repository.

The core elements of the reasoning framework are depicted in Figure 6. To host the IMMERSE domain ontology, we have selected the GraphDB graph database [33]. GraphDB is a semantic graph database, compliant with the World Wide Web Consortium (W3C) standards. It is a highly efficient and robust RDF triplestore that provides native OWL 2 RL reasoning services as well as SPARQL-based query interfaces. One important aspect of GraphDB is that it implements the RDF4J framework [34], which is an open-source Java framework, widely used by the semantic community, for working with RDF data, including parsing, storing, inferencing, and querying.

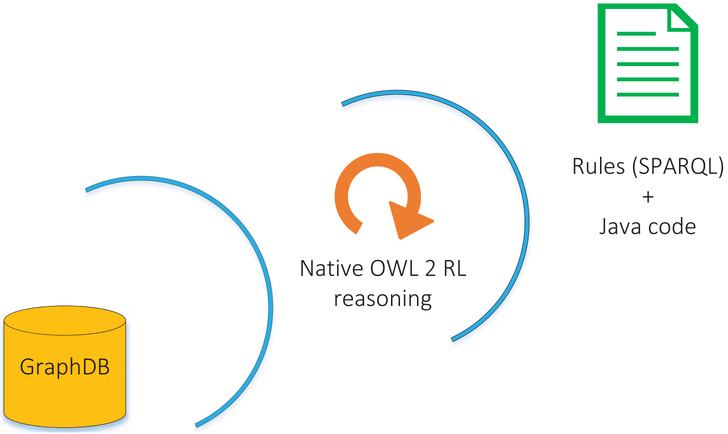


Figure 6. Abstract reasoning architecture.

The OWL 2 RL [35] reasoning, supported by GraphDB, ensures that the properties and characteristics of the OWL 2 language are fully supported. However, OWL 2 reasoning supports limited expressivity and is not able to handle complex domain relations and rules. In addition, it does not allow the dynamic generation of new instances in the knowledge repository. Considering those limitations, we also use a combination of SPARQL queries and Java code (i.e., using the RDF4J framework) to support more advanced and complex relations and enhance the reasoning capabilities of the system. The main idea is to associate each reasoning task with a set of SPARQL queries and algorithms to address specific reasoning requirements. In the following section, we present the requirements that are relevant to the semantic reasoning framework and then the reasoning tasks that were implemented based on the requirements.

Requirements and Competency Questions

A competency question (CQ) is a natural language sentence that expresses a pattern for a type of question people expect an ontology to answer [36]. The answerability of CQs hence becomes a functional requirement of the ontology. This section first provides an overview of the requirements that drove the development process of the proposed reasoning framework and then a list of CQs that the IMMERSE ontology should be able to respond to. Table 2 contains an indicative subset of the reasoning requirements.

Table 2. Requirements relevant to Semantic Reasoning.

Requirement Number	Description
SR_EMPL_01	System should support semantic matchmaking and dynamic discovery of new relationships by automatically analysing the existing content.
SR_EMPL_02	Migrant Users should be able to review matched opportunities, i.e., Jobs matching their CV and in-app interactions.

Based on the list of user requirements above, the ontology is able to respond to several CQs. Table 3 below presents a list of CQs that the IMMERSE Ontology should be able to respond to considering the requirements in Table 2.

Table 3. Competency Questions relevant to Semantic Reasoning.

Code	Description
CQ_01	Which job posts could be recommended to a migrant?
CQ_02	Which job posts match the migrant's language?
CQ_03	Which job posts match the migrant's education?
CQ_04	Which job posts match the migrant's work experience?
CQ_05	Which job posts match the migrant's skills?
CQ_06	Which job posts match the migrant's expectations for finding work?
CQ_07	Which job posts are similar to each other?
CQ_08	Which job posts are similar to posts that the migrant has applied for or saved as a favorite?
CQ_09	Which job posts are closer to the user's location?

The reasoning that is performed is mostly rule-based, meaning that, considering the reasoning requirements, a different set of SPARQL queries is executed along with appropriate Java code. It consists of several individual tasks (see Figure 7), each related to a particular SPARQL rule set. The intermediate results of each task are stored in the IMMERSE domain ontology and then combined to calculate the final recommendations. The role of the KBS is to select which of them should be triggered and execute them. The general goal of the matchmaking process is to recommend posts to users that may be interested in or posts for which the users are appropriate candidates. It is worth mentioning that these tasks can be performed in two cases:

- ***At the user level***, meaning that, given a single user profile (i.e., new or updated), the system will try to find all possible matches against all available posts of a relevant service or,
- ***At the post level***, meaning that, given an individual post (i.e., new or updated) of a service, the system will try to find all possible matches of this post against all relevant profiles.

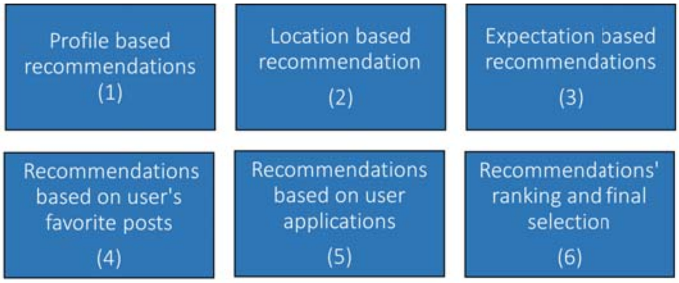

Profile based recommendations (1)
Location based recommendation (2)
Expectation based recommendations (3)
Recommendations based on user's favorite posts (4)
Recommendations based on user applications (5)
Recommendations' ranking and final selection (6)

Figure 7. Individual reasoning tasks.

In both cases, the recommendation results are captured in the ontology and stored under the RDF graph of the particular user. Once they are stored, they are sent to the Data Management Service (DMS) that will store them along with other information relevant to the user. DMS then informs the User Interface (UI) to make the results visible to the user.

6. Simulation Example

Following the methodology proposed in [36], we translated the list of CQs into respective SPARQL queries [37]. In the rest of this section, we provide more details on the individual reasoning tasks and the queries that are performed by the semantic reasoning component. We are going to demonstrate them through a simulation example. The example assumes that a new user has registered and populated the CV, thus the rules are performed at the ***user level***. In case a new post is introduced in the system, the component automatically adjusts the same rules, and calculations are performed at the ***post level***.

6.1. Initialization Step

The step prior to any other execution of the semantic reasoning component is the initialization of the triples that will be used throughout the calculations. The SPARQL query in Figure 8 populates the appropriate properties and creates a link between the user and the posts offered in the user's country as well as sets all scores to zero.

Once the triples for each potential recommendation are initialized, the first actual reasoning task that is executed concerns the recommendations that the system calculates considering only the profile of the migrant user. This reasoning task is executed in four steps:

- ***Step 1*** considers the languages spoken by the user;
- ***Step 2*** considers the user's education;
- ***Step 3*** considers the user's past work experience and;
- ***Step 4*** considers the user's skills;

6.2. Step 1: Language Recommendations

Through the UI, migrant users can define the languages they speak along with their level, and service providers can define the language requirements of the posts they are creating. This information is used by the semantic reasoning component to find which posts match the profile of the migrant considering the language. Even if the user does not provide any language, the system makes use of the user's preferred language that is provided upon registration. The following SPARQL query in Figure 9 considers only posts that could be potentially recommended to a particular user as those have been calculated during the initialization step. It retrieves all possible combinations of languages spoken by a user and languages required by a post.

For example, assuming that a migrant speaks English and Arabic, and a post requires English and French (see Table 4), then the previous query would return the following combinations:

The next step is to check if there is a perfect match, which means that the languages that are spoken by the user match all the required languages. In this case, the system updates the ***langScore*** property by assigning the calculated score; otherwise, it gives a default equal to −10.0 (i.e., meaning that the user is not a suitable candidate for this post based on the language requirements).

6.3. Step 2: Education Recommendations

Through the UI, migrant users can define their education (i.e., the education field and level of education) and service providers can define the education requirements of the posts they are creating (if any). To find which posts match the migrant's education, the system queries the ontology to obtain all possible combinations of the migrant's education and the education requirement of all posts using the SPARQL query depicted in Figure 10. The rule considers only posts that could be potentially recommended to a particular user as those have been calculated during the initialization step.

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX miict: <http://www.semanticweb.org/certh/miict#>
INSERT {
    GRAPH ?g {
        ?s miict:hasRecommendation ?rec_job .
        ?rec_job rdf:type miict:JobRecommendation ;
            miict:pointsToPost ?job ;
            miict:langScore 0.0 ;
            miict:eduScore 0.0 ;
            miict:workScore 0.0 ;
            miict:expScore 0.0 ;
            miict:simScore 0.0 ;
            miict:skillScore 0.0 ;
            miict:distScore 0.0 .
    }
} WHERE {
    GRAPH ?g {
        ?s miict:hasRole miict:Migrant ;
            miict:id ?id ;
            miict:hasLocation ?loc .
            ?loc miict:country ?country .
            ?job rdf:type miict:JobPost ;
                miict:id ?job_id ;
                miict:hasLocation ?job_loc .
            ?job_loc miict:country ?job_country .
            BIND(IRI(CONCAT("miict:", CONCAT(STR(?id),STR(?job_id))))
                AS ?rec_job)
            FILTER(?id = STR("60a6d20d6ec17bc2c599efb9")
                && ?country = ?job_country)
        }
}
```

Figure 8. CQ_01: Which job posts could be recommended to a migrant?

Table 4. Language combinations.

Combination	Spoken Language	Required Language	Score
1	English (Beginner)	English (Beginner)	1.0
2	English (Beginner)	French (Beginner)	0.0
3	Arabic (Advanced)	English (Beginner)	0.0
4	Arabic (Advanced)	French (Beginner)	0.0

For example, assuming that a user has a Master's Degree (i.e., level of education) in Information Technology (i.e., education field), and a post requires a Bachelor's Degree in Information Technology (see Table 5), then the previous query would return the following combination:

The next step is to check if there is a perfect match meaning that the education of the user covers the education requirements of the post. In this case, the system updates the ***eduScore*** property by assigning the calculated score; otherwise, it gives a default value equal to −10.0 (i.e., meaning that the user is not a suitable candidate for this post based on the education requirements).

```
PREFIX miict: <http://www.semanticweb.org/certh/miict#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
    SELECT DISTINCT ?r ?name ?level ?rname ?rlevel WHERE {
        ?s miict:hasRole miict:Migrant ;
            miict:id ?id ;
            miict:speaksLanguage ?l .
        ?l miict:languageName ?name ;
            miict:languageLevel ?level .
        ?s miict:hasRecommendation ?r .
        ?r rdf:type miict:JobRecommendation ;
            miict:pointsToPost ?j .
        ?j rdf:type miict:JobPost ;
            miict:id ?job_id ;
            miict:requiresLanguage ?rl .
        ?rl miict:languageName ?rname ;
            miict:languageLevel ?rlevel .
        FILTER(?id = STR("60a6d20d6ec17bc2c599efb9"))
}
```

Figure 9. CQ_02: Which job posts match the migrant's language?

```
PREFIX miict: <http://www.semanticweb.org/certh/miict#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
    SELECT DISTINCT ?r ?field ?level ?rfield ?rlevel WHERE {
        ?s miict:hasRole miict:Migrant ;
            miict:id ?id ;
            miict:hasEducation ?e .
        ?e miict:educationField ?field ;
            miict:educationDegreeLevel ?level .
        ?s miict:hasRecommendation ?r .
        ?r rdf:type miict:JobRecommendation ;
            miict:pointsToPost ?j .
        ?j rdf:type miict:JobPost ;
            miict:id ?job_id ;
            miict:requiresEducation ?re .
        ?re miict:educationField ?rfield ;
            miict:educationDegreeLevel ?rlevel .
        FILTER(?id = STR("60a6d20d6ec17bc2c599efb9"))
}
```

Figure 10. CQ_03: Which job posts match the migrant's education?

Table 5. Education combinations.

Combination	Education	Required Education	Score
1	IT (Master)	IT (Bachelor)	1.0

6.4. Step 3: Work Experience Recommendations

Similarly, through the UI, migrant users can define their past work experience (i.e., the field of the jobs they have worked) and service providers can define the job category of the posts they are creating. To find which posts match the migrant's past experience, the system queries the ontology to obtain all possible combinations between the job field of the migrant's past experience and the job field of all posts using the SPARQL query depicted in Figure 11. The rule considers only posts that could be potentially recommended to the particular user as those have been calculated during the initialization step.

```
PREFIX miict: <http://www.semanticweb.org/certh/miict#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
    SELECT DISTINCT ?r ?field ?ca_name WHERE {
        ?s miict:hasRole miict:Migrant ;
            miict:id ?id ;
            miict:hasWorkExp ?exp .
        ?exp miict:expField ?field .
        ?s miict:hasRecommendation ?r .
        ?r rdf:type miict:JobRecommendation ;
            miict:pointsToPost ?j .
        ?j rdf:type miict:JobPost ;
            miict:id ?job_id ;
            miict:hasCategory ?ca .
        ?ca miict:categoryName ?ca_name .
        FILTER(?id = STR("60a6d20d6ec17bc2c599efb9"))
}
```

Figure 11. CQ_04: Which job posts match the migrant's work experience?

For example, assuming that a user has worked in the field of Information Technology and Education and Training, and a post is offered under the category Education and Training (see Table 6), then the previous query would return the following combinations:

Table 6. Work Experience Combinations.

Combination	Work Exp.	Required Work Exp.	Score
1	IT	Education & Training	0.0
2	Education & Training	Education & Training	1.0

The last column is the score that is calculated for each combination. More specifically, if any of the past experience matches the job category of the post, then the given score is 1.0. In this case, the system updates the ***workScore*** property by assigning the calculated score. In the case of the past work experience, we do not assign a value of −10.0 if the user has no past experience because this post could be recommended assuming that the user would cover the language and education requirements of the post.

6.5. Step 4: Skills Recommendations

Finally, through the UI, migrant users can define their skills and service providers can define the skill requirements of the posts they are creating. Both sides can select skills from a predefined dropdown list. To find which posts match the migrant's skills, the system queries the ontology to obtain the list of the skills defined by the user along with the list of skill requirements defined for each post using the SPARQL query depicted in Figure 12. The rule considers only posts that could be potentially recommended to the particular user as those have been calculated during the initialization step.

For example, assuming that a user has the skills MS Office and Meeting Deadlines, and a couple of posts require a particular set of skills (see Table 7), then the previous query would return the following combinations:

The last column is the score that is calculated for each combination. More specifically, the score is calculated based on the following equation:

$$score = \frac{\#\ of\ matched\ skills}{\#\ of\ total\ required\ skills} \quad (1)$$

```
PREFIX : <http://www.semanticweb.org/certh/miict>
PREFIX miict: <http://www.semanticweb.org/certh/miict#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
    SELECT ?r ?list ?req_list WHERE {
        ?s miict:hasRole miict:Migrant ;
            miict:id ?id ;
            miict:hasRecommendation ?r ;
            miict:hasSkills ?skills .
        ?skills miict:skillList ?list .

        ?r rdf:type miict:JobRecommendation ;
            miict:pointsToPost ?p .
        ?p rdf:type miict:JobPost ;
            miict:id ?job_id ;
            miict:expRequirements ?req .
        ?req miict:skillList ?req_list .
        FILTER (?id = STR("60a6d20d6ec17bc2c599efb9")
        && ?list != "undefined" && ?req_list != "undefined")
}
```

Figure 12. CQ_05: Which job posts match the migrant's skills?

Table 7. Skills' combinations.

Combination	Skills	Required Skills	Score
1	MS Office, Meeting Deadlines	MS Office, Problem Solving	0.5
2	MS Office, Meeting Deadlines	Java	0.0

6.6. Expectations' Recommendations

The panel of migrant users enables them to define the expectations they have from the IMMERSE platform such as finding work in a particular field where they can also define the work condition they prefer (e.g., part-time). To find which posts match the migrant's expectations for work, the system queries the ontology to obtain all possible combinations between the work expectations of the migrant and the job field of all posts using the SPARQL query depicted in Figure 13. The rule considers only posts that could be potentially recommended to the particular country as those have been calculated during the initialization step.

For example, assuming that a user wants to work in the field of Hospitality and Tourism (i.e., full-time) and there are two posts in the field of Hospitality and Tourism, one offering part-time and the other full-time conditions (see Table 8), then the previous query would return the following combinations:

The last column is the score that is calculated for each combination. More specifically, if a given combination matches only the job field but not the work condition, the score is 0.5, but if both job field and work condition match, the given score is 1.0. The level of the language spoken by the migrant needs to be at least equal to or greater than the level of the required language to obtain a score of 1.0.

```
PREFIX miict: <http://www.semanticweb.org/certh/miict#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
SELECT DISTINCT ?r ?cat ?c ?name ?cond WHERE {
    ?s miict:hasRole miict:Migrant ;
        miict:id ?id ;
        miict:hasExpectation ?e .
    ?e rdf:type miict:WorkExpectation ;
        miict:exp_condition ?c ;
        miict:exp_category ?cat .
    ?s miict:hasRecommendation ?r .
    ?r rdf:type miict:JobRecommendation ;
        miict:pointsToPost ?j .
    ?j rdf:type miict:JobPost ;
        miict:id ?job_id ;
        miict:hasCategory ?ca ;
        miict:workCondition ?cond .
    ?ca miict:categoryName ?name .
    FILTER(?id = STR("60a6d20d6ec17bc2c599efb9"))
}
```

Figure 13. CQ_06: Which job posts match the migrant's expectations for finding work?

Table 8. Expectations' Combinations.

Combination	Work Expectation	Offered Jobs	Score
1	Hospitality & Tourism/full-time	Hospitality & Tourism/full-time	0.5
2	Hospitality & Tourism/full-time	Hospitality & Tourism/part-time	0.0

6.7. Similarities' Recommendations

Considering the interactions of the migrant within the IMMERSE platform, the system needs to be able to also recommend similar posts to those that the user has either showed interest in (i.e., saved as a favorite posts) or applied. By similar posts, we mean posts that have similar requirements (e.g., similar language or past experience requirements). Thus, it is important that the system can identify those posts which are similar. This section provides information on how the system calculates the similarities between different posts. To find which posts are similar based on their requirements, the system first queries the ontology to get all requirements of the posts using the SPARQL query depicted in Figure 14. In our example, those would be the job category, languages, education, country, and work condition. The query considers all job posts that are available on the platform.

The values of the ***category*** and the ***country*** are used by the system to compare only jobs of the same category that are offered in the same country. Then, each post is compared against all other posts and a score is calculated. For example, assuming that we have the following two job posts having the values shown in Table 9, we can see that 3 out 5 properties have the same values; thus, the final score of their similarity is 0.6.

Eventually, every time a new post is added to the platform, the reasoning component first calculates which posts are similar to it. It then starts re-calculating recommendations based on favourite posts and applications for each user living in the same country. The following SPARQL rule, depicted in Figure 15, queries the ontology to find all the posts that the migrant user has either saved or applied and also finds which posts are similar to them. For each similar post, it also queries the final similarity score and uses it to update the property ***simScore*** that holds the score of the recommendation object.

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX miict: <http://www.semanticweb.org/certh/miict#>
SELECT DISTINCT ?category ?language_name ?l_level ?ed_field ?ed_level
    ?condition {
    ?s rdf:type miict:JobPost;
        miict:id ?id ;
        miict:jobName ?name ;
        miict:jobDescription ?description ;
        miict:hasCategory ?c ;
        miict:hasLocation ?loc .
    ?c miict:categoryName ?category .
    ?loc miict:country ?country .
    ?s miict:requiresEducation ?education .
    ?education miict:educationField ?ed_field .
    ?education miict:educationDegreeLevel ?ed_level .
    OPTIONAL {
        ?s miict:requiresLanguage ?language .
        ?language miict:languageName ?language_name .
        ?language miict:languageLevel ?l_level .
    }
    ?s miict:workCondition ?condition .
}
```

Figure 14. CQ_07: Which job posts are similar to each other?

Table 9. Post similarity combinations.

Property	Job #1	Job #2	Score
Category	Human Services	Human Services	-
Country	Spain	Spain	-
Condition	Full-Time	Part-Time	0.0
Education Level	High School	High School	1.0
Education Field	No Requirement	No Requirement	1.0
Language #1	English	English	1.0
Language #2	French	-	0.0
		Total Score	0.6 (=3/5)

For example, assuming that a user has one saved post (Post #1) and applied to another post (Post #2) where Post #1 is similar to Post #3 with a score of 0.5 and with Post #4 with a score of 0.3. In addition, consider that Post #2 is also similar to Post #3 with a score of 0.4. Then, based on the previous query, as shown in Table 10, Post #3 will have a recommendation score (i.e., based on its similarity with posts that the user has interacted with) equal to 0.45 (=0.5 + 0.4/2), which is the average value of the two similarity scores, and Post #4 will have a recommendation score equal to 0.3.

6.8. Distance Recommendations

Each post that is created in the IMMERSE platform and each user that registers has a location. Even if no specific address is provided by the migrant user, the system knows at least the country where the user is located since this information is provided upon registration from all users. Based on those two locations, the system can calculate the distance between users and posts, thus giving more weight to posts that are closer to the user. Table 11 summarizes the distance ranges that are considered by the system and the scores that are assigned to each case.

```
PREFIX miict: <http://www.semanticweb.org/certh/miict#>
DELETE {?r miict:simScore ?old_score}
INSERT {
   GRAPH miict:60a6d20d6ec17bc2c599efb9
   {
      ?r miict:simScore ?avg
   }
} WHERE {
   ?s miict:hasRole miict:Migrant ;
      miict:id ?id .
   ?s miict:hasRecommendation ?r .
   ?r rdf:type miict:JobRecommendation ;
      miict:pointsToPost ?post .
   ?r miict:simScore ?old_score .
   {
      SELECT ?post (AVG(?score) as ?avg) WHERE {
         ?s miict:hasRole miict:Migrant ;
            miict:id ?id ;
            miict:hasAppliedJobs ?applied ;
            miict:hasSavedJobs ?saved ;
            miict:hasRecommendation ?r .
         ?r rdf:type miict:JobRecommendation ;
            miict:pointsToPost ?p ;
            miict:simScore ?old_score .
         ?p rdf:type miict:JobPost ;
            miict:id ?job_id .
         {
            BIND (?saved AS ?posts)
         }
         UNION
         {
            BIND(?applied AS ?posts)
         }
         ?posts miict:hasSimilarity ?similarity .
         ?similarity miict:pointsToPost ?post .
         ?similarity miict:finalScore ?score .
         FILTER(?id = STR("60a6d20d6ec17bc2c599efb9"))
      } GROUP BY ?post
   }
}
```

Figure 15. CQ_08: Which job posts are similar to posts that the migrant has applied for or saved as a favorite?

Table 10. Similarity combinations.

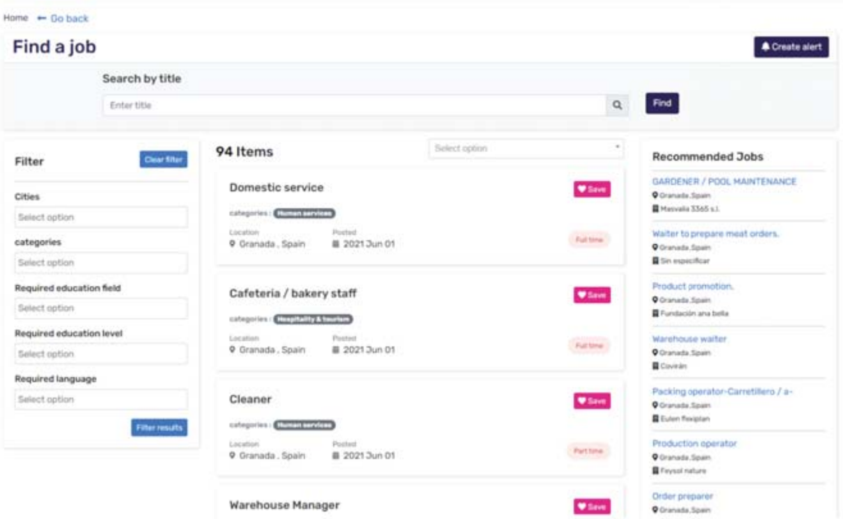
Saved or Applied Post	Similar Posts	Similarity Score
Post #1	Post #3	0.5
Post #1	Post #4	0.3
Post #2	Post #3	0.4

Table 11. Distance ranges.

Distance Range (in km)	Distance Score
[0, 6)	1.0
[6, 15)	0.7
[15, 35)	0.5
[35, 100)	0.2
[100,)	0.0

6.9. Matchmaking Results in the User Interface

Once the scores of the individual tasks are calculated, the system calculates the final recommendation scores by adding intermediate scores and sorts them in descending order. The top 10 posts are then sent back to the DMS through a post request. The body of the request is a JSON array having the IDs of the posts that need to be displayed to the UI. Those IDs are stored in the database of the DMS and then the UI retrieves additional information about those posts and provides it to the user while the user navigates through the respective service. Figure 16 depicts how the results of the reasoning component are displayed in the UI (i.e., in the Recommended Jobs sidebar).



Domestic service
categories : Human services
Location: Granada , Spain
Posted: 2021 Jun 01
Full time

Cafeteria / bakery staff
categories : Hospitality & tourism
Location: Granada , Spain
Posted: 2021 Jun 01
Full time

Cleaner
categories : Human services
Location: Granada , Spain
Posted: 2021 Jun 01
Part time

Warehouse Manager

GARDENER / POOL MAINTENANCE
Granada,Spain
Masvalia 3365 s.l.

Waiter to prepare meat orders.
Granada,Spain
Sin especificar

Product promotion.
Granada,Spain
Fundación ana bella

Warehouse waiter
Granada,Spain
Covirán

Packing operator-Carretillero / a-
Granada,Spain
Eulen flexiplan

Production operator
Granada,Spain
Feysol nature

Order preparer
Granada,Spain

Figure 16. Job recommendations in the UI.

7. Experimental Results

In this section, the proposed recommendation framework is evaluated in a real-world dataset, and the results of the evaluation are presented.

7.1. Dataset Description

To evaluate the responsiveness and scalability of the proposed reasoning framework, a real-world dataset was considered. The dataset was created during the piloting phase of the IMMERSE platform and consists of 100 real job posts and 30 user profiles where users have included in their profile information about their languages, education, work experience, skills, etc.

7.2. Evaluating the Recommender System

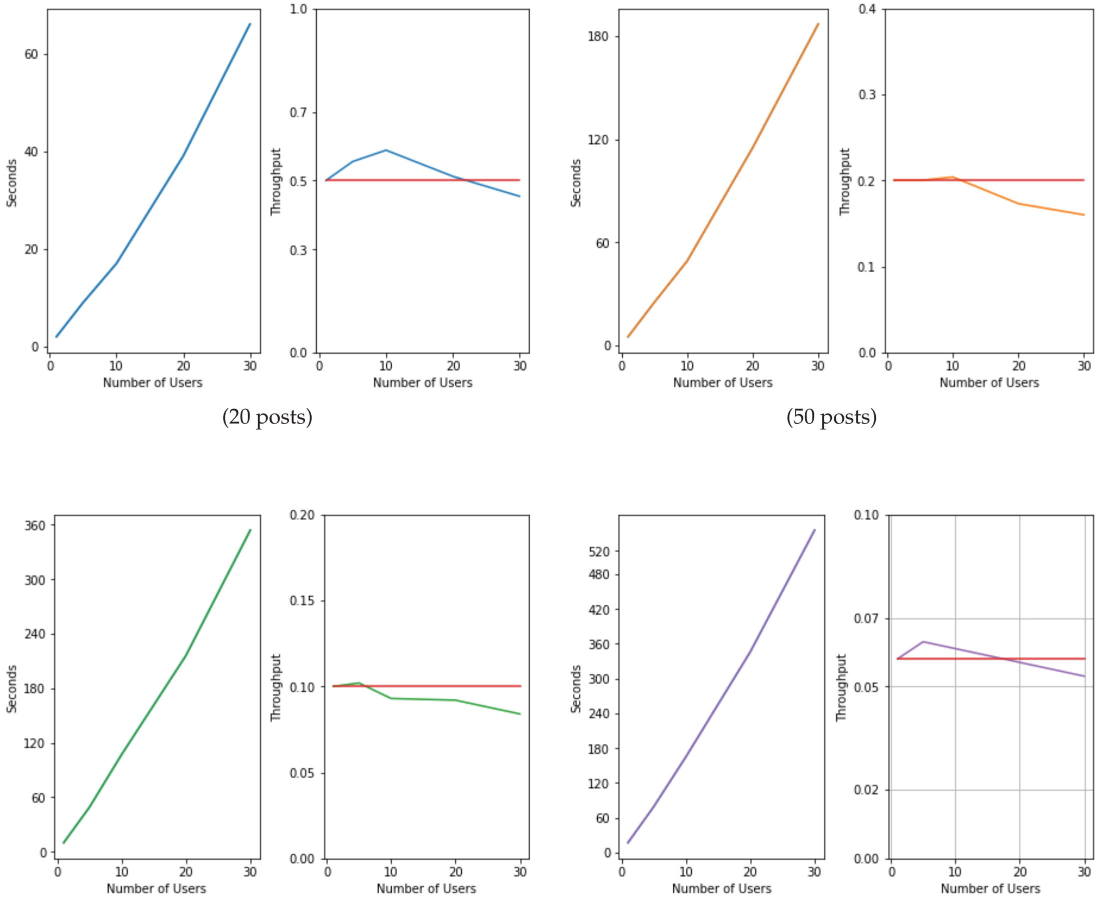
To measure the response time and see how well our system scales while increasing its processing demand, we decided to split the evaluation process into four phases. In each phase, we would gradually increment the number of available posts and then, for every phase, we would also increment the number of concurrent users for which the system would have to calculate recommendations. To measure the scalability of the system, we used a metric called *throughput*, which is defined as the number of recommendations per second.

Throughput is calculated using the following equation:

$$X = N/R, \tag{2}$$

where N is the number of concurrent users, and R is the average time the system needs to completed its calculations.

A good comparison against which we could compare our system would be a *linearly scalable* version of our system, meaning a system that continues to do exactly the same amount of work per second no matter how many users are using it. This does not mean that the system's response time will remain constant. In fact, it will increase but in a perfectly predictable manner. However, its throughput will remain constant. Linearly scalable applications are perfectly scalable in that their performance degrades at a constant rate directly proportional to their demands. The results of our evaluation are depicted in Figure 17.



(20 posts) (50 posts)

(80 posts) (100 posts)

Figure 17. Reasoning response time and scalability.

8. Conclusions

In this paper, we have discussed and demonstrated an ontology-based reasoning framework acting as a matching tool that enables the contexts of individual migrants and refugees, including their expectations, languages, educational background, previous job experience and skills, to be captured in the ontology and facilitate their matching with the job opportunities available in their host country. We have presented the main components of the IMMERSE reasoning framework and how this has been incorporated in the IMMERSE main architecture. In addition, we examined scalability and reported on computational performance as a function of the number of jobs and job seekers available in the knowledge base repository. Our results show that the system scales linearly and can be optimized for large scale implementation. Our current implementation focuses on finding appropriate jobs based on academic qualifications, languages, skills and work experience. For future work, the framework could be extended to consider several other constraints that exist when seeking jobs and these are often subjective i.e., highly dependent on the job seeker in question. Such factors are location flexibility that includes both the option to work remotely or being open to relocation, job seeker's age, type of companies the job seeker has worked with in the past (small, medium, or large size), industry, etc.

Author Contributions: Conceptualization, D.N., A.K. and G.M.; methodology, D.N., A.K. and G.M.; software, D.N., P.M. and G.M.; validation, D.N., S.V. and G.M.; formal analysis, D.N.; investigation, D.N. and G.M.; writing—original draft preparation, D.N., P.M. and G.M.; writing—review and editing, S.V. and I.K.; visualization, D.N.; supervision, S.V. and I.K.; project administration, S.V. and I.K. All authors have read and agreed to the published version of the manuscript.

Funding: This work has received funding by the EC under contracts H2020-822380 MIICT and H2020-870939 SO-CLOSE.

Data Availability Statement: Not applicable.

Acknowledgments: The authors would like to thank Patricia Bueso and Jose Gonzalez-Riera from the Euro-Arab Foundation for Higher Studies (FUNDEA) who organized the pilots where the data used for the evaluation of the proposed framework were collected.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

IMMERSE	Integration of Migrants MatchER Service
MIICT	ICT Enabled Public Services for Migration
CV	Curriculum Vitae
RDF	Resource Description Framework
OWL	Web Ontology Language
FOAF	Friend of a Friend
DOAC	Description of a Career
SPARQL	Simple Protocol and RDF Query Language
DMS	Data Management System
UI	User Interface
AUTH	Authentication
KBS	Knowledge Base Service
MSB	Message Bus
KB	Knowledge Base
KBP	Knowledge Base Population
SR	Semantic Reasoning
DL	Description Logic
W3C	World Wide Web Consortium
CQ	Competency Question

References

1. Zimmermann, K.F. Refugee and migrant labor market integration: Europe in need of a new policy agenda. In *The Integration of Migrants and Refugees. An EUI Forum on Migration, Citizenship and Demography*; European University Institute, Robert Schuman Centre for Advanced Studies: Florence, Italy, 2017.
2. AbuJarour, S.; Wiesche, M.; Andrade, A.D.; Fedorowicz, J.; Krasnova, H.; Olbrich, S.; Tan, C.W.; Urquhart, C.; Venkatesh, V. ICT-enabled refugee integration: A research agenda. *Commun. AIS* **2019**, *44*, 874–891. [CrossRef]

3. AbuJarour, S.; Krasnova, H.; Hoffmeier, F. ICT as an enabler: Understanding the role of online communication in the social inclusion of Syrian refugees in Germany. In Proceedings of the Twenty-Sixth European Conference on Information Systems (ECIS 2018), Portsmouth, UK, 23–28 June 2018.
4. Vernon, A.; Deriche, K.; Eisenhauer, S. *Connecting Refugees—HOW INTERnet and Mobile Connectivity Can Improve Refugee Well-Being and Transform Humanitarian Action*; UNHCR: Geneva, Switzerland, 2016.
5. Andrade, A.D.; Doolin, B. Information and communication technology and the social inclusion of refugees. *Mis Q.* **2016**, *40*, 405–416. [CrossRef]
6. ICT Enabled Public Services for Migration. Available online: https://www.miict.eu/ (accessed on 16 September 2022).
7. Ntioudis, D.; Karakostas, A.; Vrochidis, S.; Kompatsiaris, I. IMMERSE: A Matching Platform Improving Migrant Integration with Semantic Technologies. In *Information and Communications Technology in Support of Migration*; Springer: Cham, Switzerland, 2022; pp. 213–228.
8. Ntioudis, D.; Kamateri, E.; Meditskos, G.; Karakostas, A.; Huber, F.; Bratska, R.; Vrochidis, S.; Akhgar, B.; Kompatsiaris, I. Immerse: A personalized system addressing the challenges of migrant integration. In Proceedings of the 2020 IEEE International Conference on Multimedia & Expo Workshops (ICMEW), London, UK, 6–10 July 2020; pp. 1–6.
9. Decker, S.; Melnik, S.; Van Harmelen, F.; Fensel, D.; Klein, M.; Broekstra, J.E.; Erdmann, M.; Horrocks, I. The Semantic Web: The roles of XML and RDF. *IEEE Internet Comput.* **2000**, *15*, 63–74. [CrossRef]
10. Ma, L.; Yang, Y.; Qiu, Z.; Xie, G.; Pan, Y.; Liu, S. Towards a complete OWL ontology benchmark. In *European Semantic Web Conference*; Springer: Berlin/Heidelberg, Germany, 2006, pp. 125–139.
11. World Wide Web Consortium (W3C). Available online: https://www.w3.org/ (accessed on 28 September 2022).
12. Brickley, D.; Miller, L. FOAF Vocabulary Specification 0.91. 2007. Available online: https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.476.8247&rep=rep1&type=pdf (accessed on 28 July 2022).
13. Core Vocabularies. Available online: https://ec.europa.eu/isa2/solutions/core-vocabularies_en/ (accessed on 25 July 2022).
14. Ahmed, N.; Khan, S.; Latif, K. Job description ontology. In Proceedings of the 2016 International Conference on Frontiers of Information Technology (FIT), Islamabad, Pakistan, 19–21 December 2016; pp. 217–222.
15. Parada, R. Doac Vocabulary Specification. 2006. Available online: https://www.edatasoft.com/en/doac/01/ (accessed on 28 July 2022).
16. Fazel-Zarandi, M.; Fox, M.S. An ontology for skill and competency management. In Proceedings of the 7th International Conference on Formal Ontology in Information Systems, Graz, Austria, 24–27 July 2012; pp. 89–102.
17. Bojārs, U.; Breslin, J.G. ResumeRDF: Expressing skill information on the Semantic Web. In Proceedings of the 1 st International ExpertFinder Workshop, Berlin, Germany, 16 January 2007.
18. Gómez-Pérez, A.; Ramírez, J.; Villazón-Terrazas, B. An ontology for modelling human resources management based on standards. In *International Conference on Knowledge-Based and Intelligent Information and Engineering Systems*; Springer: Berlin/Heidelberg, Germany, 2007; pp. 534–541.
19. Harris, S.; Seaborne, A.; Prud'hommeaux, E. SPARQL 1.1 query language. *W3C Recomm.* **2013**, *21*, 778.
20. Fazel-Zarandi, M.; Fox, M.S. Semantic matchmaking for job recruitment: An ontology-based hybrid approach. In Proceedings of the 8th International Semantic Web Conference, Chantilly, VA, USA, 25–29 October 2009; Volume 525, p. 2009.
21. Malinowski, J.; Keim, T.; Wendt, O.; Weitzel, T. Matching people and jobs: A bilateral recommendation approach. In Proceedings of the 39th Annual Hawaii International Conference on System Sciences (HICSS'06), Kauai, HI, USA, 4–7 January 2006; Volume 6, p. 137c.
22. Lee, D.H.; Brusilovsky, P. Fighting information overflow with personalized comprehensive information access: A proactive job recommender. In Proceedings of the Third International Conference on Autonomic and Autonomous Systems (ICAS'07), Athens, Greece, 19–25 June 2007; p. 21.
23. Lv, H.; Zhu, B. Skill ontology-based semantic model and its matching algorithm. In Proceedings of the 2006 7th International Conference on Computer-Aided Industrial Design and Conceptual Design, Hangzhou, China, 17–19 November 2006; pp. 1–4.
24. Kenthapadi, K.; Le, B.; Venkataraman, G. Personalized job recommendation system at linkedin: Practical challenges and lessons learned. In Proceedings of the Eleventh ACM Conference on Recommender Systems, Como, Italy, 27–31 August 2017; pp. 346–347.
25. Musale, D.V.; Nagpure, M.K.; Patil, K.S.; Sayyed, R.F. Job recommendation system using profile matching and web-crawling. *Int. J.* **2016**, *1*. Available online: http://www.ijasret.com/VolumeArticles/FullTextPDF/24_IJASRET7747.pdf (accessed on 28 July 2022).
26. Koh, M.F.; Chew, Y.C. Intelligent job matching with self-learning recommendation engine. *Procedia Manuf.* **2015**, *3*, 1959–1965. [CrossRef]
27. Rodriguez, L.G.; Chavez, E.P. Feature selection for job matching application using profile matching model. In Proceedings of the 2019 IEEE 4th International Conference on Computer and Communication Systems (ICCCS), Singapore, 23–25 February 2019; pp. 263–266.
28. Drigas, A.; Kouremenos, S.; Vrettos, S.; Vrettaros, J.; Kouremenos, D. An expert system for job matching of the unemployed. *Expert Syst. Appl.* **2004**, *26*, 217–224. [CrossRef]
29. OntoMetrics. Available online: https://ontometrics.informatik.uni-rostock.de/ontologymetrics/ (accesses on 19 September 2022).
30. Description Logic. Available online: https://en.wikipedia.org/wiki/Description_logic (accessed on 19 September 2022).

31. Falco, R.; Gangemi, A.; Peroni, S.; Shotton, D.; Vitali, F. Modelling OWL ontologies with Graffoo. In *European Semantic Web Conference*; Springer: Cham, Switzerland, 2014; pp. 320–325.
32. Tartir, S.; Arpinar, I.B.; Sheth, A.P. Ontological evaluation and validation. In *Theory and Applications of Ontology: Computer Applications*; Springer: Dordrecht, The Netherlands, 2010; pp. 115–130.
33. GraphDB. Available online: https://graphdb.ontotext.com/documentation/10.0/about-graphdb.html (accessed on 19 September 2022).
34. Eclipse RDF4J framework. Available online: https://rdf4j.org/ (accessed on 19 September 2022).
35. OWL 2 Web Ontology Language Profiles (Second Edition). Available online: https://www.w3.org/TR/owl2-profiles/#OWL_2_RL (accessed on 19 September 2022).
36. Uschold, M.; Gruninger, M. Ontologies: Principles, methods and applications. *Knowl. Eng. Rev.* **1996**, *11*, 93–136. [CrossRef]
37. Zemmouchi-Ghomari, L.; Ghomari, A.R. Translating natural language competency questions into SPARQL queries: A case study. In Proceedings of the First International Conference on Building and Exploring Web Based Environments, Seville, Spain, 27 January–1 February 2013; pp. 81–86.

big data and cognitive computing

MDPI

Article

THOR: A Hybrid Recommender System for the Personalized Travel Experience

Alireza Javadian Sabet [1,2,*,†], Mahsa Shekari [3,†], Chaofeng Guan [2], Matteo Rossi [3], Fabio Schreiber [2] and Letizia Tanca [2]

[1] Department of Informatics and Networked Systems, University of Pittsburgh, Pittsburgh, PA 15260, USA
[2] Politecnico di Milano, Dipartimento di Elettronica, Informazione e Bioingegneria, Via Giuseppe Ponzio, 34/5, I-20133 Milano, Italy
[3] Politecnico di Milano, Dipartimento di Meccanica, Via La Masa, 1, I-20156 Milano, Italy
* Correspondence: alj112@pitt.edu
† These authors contributed equally to this work.

Abstract: One of the travelers' main challenges is that they have to spend a great effort to find and choose the most desired travel offer(s) among a vast list of non-categorized and non-personalized items. Recommendation systems provide an effective way to solve the problem of information overload. In this work, we design and implement "The Hybrid Offer Ranker" (THOR), a hybrid, personalized recommender system for the transportation domain. THOR assigns every traveler a unique contextual preference model built using solely their personal data, which makes the model sensitive to the user's choices. This model is used to rank travel offers presented to each user according to their personal preferences. We reduce the recommendation problem to one of binary classification that predicts the probability with which the traveler will buy each available travel offer. Travel offers are ranked according to the computed probabilities, hence to the user's personal preference model. Moreover, to tackle the cold start problem for new users, we apply clustering algorithms to identify groups of travelers with similar profiles and build a preference model for each group. To test the system's performance, we generate a dataset according to some carefully designed rules. The results of the experiments show that the THOR tool is capable of learning the contextual preferences of each traveler and ranks offers starting from those that have the higher probability of being selected.

Keywords: Context-Aware Recommender System; personalization; preferences; user modeling; journey planning; mobility; cold start; classification; clustering

Citation: Javadian Sabet, A.; Shekari, M.; Guan, C.; Rossi, M.; Schreiber, F.; Tanca, L. THOR: A Hybrid Recommender System for the Personalized Travel Experience. *Big Data Cogn. Comput.* **2022**, *6*, 131. https://doi.org/10.3390/bdcc6040131

Academic Editors: Konstantinos Kotis and Dimitris Spiliotopoulos

Received: 21 September 2022
Accepted: 1 November 2022
Published: 4 November 2022

1. Introduction

The recent blazing-fast advancements in big data analysis have unlocked the potential of many novel applications for smart living [1]. In particular, big data techniques can be used to greatly enhance how users experience personalized mobility, for which demand is growing fast [2]. The Internet has become the main channel for travelers to obtain online information before traveling, but they still spend a great effort to find and choose the most desired travel offer(s) among a vast list of non-categorized and non-personalized ones [3].

A Recommender System (RS), a.k.a., Recommendation System, aims at predicting the "preference" of a user about an item [4] and may provide a great help when users have search or selection problems. Recently, there have been various advancements in the field of Context-Aware Recommender Systems (CARS) [5–7]. Indeed, one should also take into account that travel preferences may be significantly influenced by *the context in which a traveler interacts with the system* [8,9]. According to Dey [10], "Context is any information that can be used to characterize the situation of an entity, where an entity can be a person, place, or physical or computational object" and "A system is context-aware if it uses context to provide relevant information and services to the user, where relevancy depends on the user's task". For more information about context-aware systems, we refer the reader to surveys provided in [11–13].

Despite various works that tackled the problem of recommending touristic destinations to travelers [14–16], to the best of our knowledge, no works exist that designed CARS for ranking a list of complete travel offers for travelers.

In this work, we assume that the user has access to an application to ask for travel offers, such as, for example, the Travel Companion (TC) provided by the Shift2Rail ecosystem (see Section 2.1). To provide a personalized experience to travelers, Javadian et al. designed a high-level system architecture in [9] and a contextual preference model in [8] using the Context Dimension Tree methodology [17]. This work aims to implement the recommender core of the TC (or of a TC-like application) based on the contextual preference model presented in [9]. More precisely, this work aims at answering the following research questions (RQs):

RQ1: Using the traveler's historical records, how can we design a personalized preference model which ranks the available travel offers according to the contextual preferences of the traveler?

RQ2: Given a new traveler without any historical records, how can we design an initial personal preference model for them using other travelers' historical data with the most similar characteristics?

To tackle these research questions and provide travelers with a Context-Aware Recommender System, the *contribution* of this work is the development of "The Hybrid Offer Ranker" (THOR), which could be integrated in applications such as the Shift2Rail TC. Unlike the proposed RS in [18], which requires a user-specified list of preferences for filtering and ranking the travel offers, THOR relies only on the historical purchase records of the users while incorporating the user-specified preferences whenever available. THOR assembles various algorithms to create a pipeline that considers the problem of ranking travel offers shown to users according to their preferences as a *binary classification problem* that predicts if the traveler will buy any of the available travel offers or not. First, each traveler is assigned a unique *contextual preference model* built using their data. For this purpose, THOR uses various well-known classification algorithms—i.e., *K-Nearest Neighbors (KNN)* [19], *Support Vector Classifier (SVC)* [20], *Decision Tree (DT)* [21], *Random Forest (RF)* [22], and *Logistic Regression (LR)* [23]—and finds the best set of hyper-parameters. Then, when the system receives a set of travel offers to be shown to the user, it exploits the contextual preference model to determine, for each offer, the probability that the user will buy it and ranks the offers accordingly. Moreover, THOR incorporates the *K-means* [24] and *Density-Based Spatial Clustering of Applications with Noise (DBSCAN)* [25] clustering algorithms to identify users with similar profiles and build a preference model for each group of similar users. In the case of a new user without any record, we identify the group that contains the most similar users and use its associated preference model to provide recommendations for the new user. Notice that the clustering and classification algorithms are used independently of one another. More precisely, during each training phase, all training algorithms are executed separately, and the one that produces the model with the best performance—i.e., the highest accuracy—is saved for future use. To test the system's performance (in terms of both computation time and accuracy), we generate an extensive dataset according to some carefully designed rules. The results show that the proposed framework is promising and can provide benefits to existing systems.

The rest of the paper is organized as follows. Section 2 presents state-of-the-art methodologies concerning the problem of designing RSs in the transportation domain. Section 3 details THOR's implementation. Section 4 shows the validation of THOR using different approaches. Finally, Section 5 concludes and presents some future work.

2. Background and Related Work

This section discusses the relevant related work and how this inspired the methodology we propose. To the best of our knowledge, most previous research on RS for the transportation domain focuses on designing a system that aims at recommending the most suitable destination to a traveler; instead, our approach aims to design an RS that, given a destination and a mobility request (i.e., the action in which a traveler uses an application to request possible travel solutions to go from a source to a destination), ranks a set of travel offers satisfying that request according to the user preferences. In Section 2.1 we briefly discuss the ecosystem in which we chose to integrate our proposed system. Section 2.2 discusses the state-of-the-art about RSs in the transportation domain; Section 2.3 investigates various important features (e.g., time, location, travel purpose) contributing to the travelers' preferences employed by state-of-the-art approaches; finally, Section 2.4 investigates techniques to tackle the crucial *cold start* problem.

2.1. Ecosystem

This work is performed within the Ride2Rail (R2R) project [26] (ride2rail.eu, accessed on 10 October 2022), in the frame of Innovation Programme 4 (IP4) of the Shift2Rail (S2R) initiative (www.shift2rail.org, accessed on 10 October 2022)) which aims at enhancing the European Transportation domain. The S2R initiative aims at implementing a collaborative ecosystem through its Interoperability Framework [27] that provides various modules and services such as automated mapping [28,29], data conversion [30], and ontology management [31]. The ecosystem facilitates the interoperability between IP4 services (e.g., Booking, Journey Planning) and Travel Service Providers (TSP) and permits them to interact, share data, and create more complex services while addressing their security and privacy concerns [32]. The ecosystem includes a component, the so-called Travel Companion (TC), which is an application that can run on various devices—e.g., smartphones—and assists travelers before, during, and after each journey.

As mentioned in Section 1, the existence of an application assisting users in setting up their travels is a prerequisite of our approach. The proposed system (THOR) is general and it could be part of any such application. However, the specific application into which the system is integrated has an impact on THOR's implementation. In particular, the set of travel features handled by the application determines the data fed to THOR for learning preference models (see Table 1) and affects the resulting recommendations. THOR was developed to be part of the S2R ecosystem, hence its implementation is compatible with the S2R TC. More precisely, THOR is part of the R2R Offer Categorizer module (OC, see Section 3.1), which pre-analyzes travel offers before feeding them to THOR and retrieves the results of THOR's computation. The complete implementation of the OC is not part of this work.

2.2. Recommender Systems

Usually, an RS falls into one of the following categories [33,34]: (*i*) *Content-based*, where recommendations are provided based on the user's past purchases; (*ii*) *Collaborative*, where recommendations are based on other users with similar preferences; and (*iii*) *Hybrid*, which combine (*i*) and (*ii*).

Valliyammai et al. [35] propose a model-based, collaborative filtering system based on fuzzy c-means [36] for clustering and A-priori [37] for classification. Motivated by their future work, to enhance the proposed system, we utilize Support Vector Machines (SVM) for our proposed system.

Sebasti et al. [15] developed a framework that collects user profiles by requiring users to enter their details and general preferences and asking them to introduce their specific preferences for the current visit. Then, the system generates a list of activities that are likely of interest to the user. A hybrid RS classifies users into different categories—e.g., "Person with Children". Then, a content-based approach recommends more places according to the user's history, and a filter selects the proper places based on the current request. Unlike Sebasti et al. [15], our proposed approach relies only on the historical records of the travelers and does not require them to specify their preferences. In our system, a user can optionally specify some preferences which will be used by the recommender system. Further details are available in Section 3.

Table 1. Main features used for data generation along with their type and category.

Name	Type	Category	Name	Type	Category
Age	Cat.	Profile	Starting Point	Cat.	Offer
City	Cat.	Profile	Destination	Cat.	Offer
Country	Cat.	Profile	Via	List	Offer
Loyalty Cards	List	Profile	LegMode	List	Offer
Payment Cards	List	Profile	Class	List	Offer
PRM Type	List	Profile	Seats Type	List	Offer
Profile Type	Cat.	Profile	Arrival Time	Cat.	Offer
Quick	Float	Offer	Departure Time	Cat.	Offer
Reliable	Float	Offer	Preferred Transp. Types	List	Search
Cheap	Float	Offer	Preferred Carriers	List	Search
Comfortable	Float	Offer	Preferred Refund Type	Cat.	Search
Door-to-door	Float	Offer	Preferred Services	List	Search
Envir. Friendly	Float	Offer	Max. No. of Transfers	Int.	Search
Short	Float	Offer	Max. Transfers Duration	Cat.	Search
Multitasking	Float	Offer	Max. Walking Dist. to Stop	Cat.	Search
Social	Float	Offer	Walking Speed	Cat.	Search
Panoramic	Float	Offer	Cycling Distance to Stop	Cat.	Search
Healthy	Float	Offer	Cycling Speed	Cat.	Search
Legs Number	Int.	Offer	Driving Speed	Cat.	Search

Lorenzi et al. [38] decompose a recommendation into travel services, and different services are managed by different agents (that work cooperatively). As the recommendation process proceeds, each agent becomes expert in one or more specific travel service. The interaction among these specialized agents results in better recommendations, and an excellent offer will be generated by combining all services recommended by different agents in this method. Although the system could be suitable for one user, the performance can be quite different for different users. In other words, the system lacks personalization at the individual level.

Javadian et al. [39] propose a data-mining-based RS to rank offers. They use association rule mining to calculate the similarities between the user requests and the offers. The work first designs the features through a historical traveler database and, after a pre-processing phase, a knowledge base is set up by mining association rules from the database. Then, the knowledge base enables scoring each offer according to the characteristics of the user's mobility request. The main limitation of their work is the choice of rule-based algorithms (e.g., A priori) which are dependent on extracted/predefined rules which do not guarantee a good performance on the undefined cases. Moreover, like other systems introduced earlier, they rely only on the wisdom of the crowd and ignore the individual-level personalization.

Fang et al. [40] automatically generate, from a collection of documents based on Wikipedia and Twitter, temporal feature vectors of Point Of Interests (POIs) that include seasonal attractions such as water sports, snow festivals, and the viewing of scarlet maple leaves. Similarly, Coelho et al. [41] employ travel-related tweets to personalize recommendations regarding POIs for the user. Firstly, they categorize POIs as historical buildings, museums, parks, and restaurants; then, they build a classification model to classify the tweets. To obtain a better-personalized model, travel-related tweets of the user's friends and followers are also mined. The proposed systems by Fang et al. [40] and Coelho et al. [41] have two main limitations: they recommend POIs rather than complete travel offers for a journey, and they lack user modeling techniques.

Fararni et al. [42] explore a user profiling process according to the following scenarios: (*i*) *Inscription*: Inserting preferences, which is done directly by the user. (*ii*) *Social login*: Obtaining preferences from the user's Social Media (SM) accounts. (*iii*) *Consultation even without login*: Observation and analysis of the user behaviors. (*iv*) *Context*: Contextual information to generate dynamic visit schedules. The system also incorporates tourist services information, a contextual meta-model for analyzing the input data, a hybrid filtering process for storing the list of items with users' appreciation degree, and a trip planner for correlating all the choices as a trip. Our proposed system differs from the one by Fararni et al. [42] in many ways. Considering the *Inscription* scenario, although we incorporate user-provided preferences, our system does not depend on them. Concerning the users' behavioral aspects and context, our system implicitly learns the behavioral changes and contextual preferences by updating the users' preference models with recent purchase histories. Finally, concerning the *Social login* scenario, due to privacy concerns, we did not integrate SM as source of information in the current implementation of the system. Note that our proposed system can be extended to incorporate SM information using the social media core proposed in [9].

2.3. Travelers' Preferences

This section overviews the works analyzing the main characteristics of travels and travelers that potentially affect travelers' preferences. From their analysis, we derived a contextual dimension tree capturing the main features presented in [43].

Basile et al. [44] propose a system to understand if the users' preferences explicitly match their behavior. First, the system builds user profiles, i.e., general descriptions of the users based on their travel preferences; second, it builds profiles using evaluation data collected after the actual travels; and finally, the before- and after-travel user profiles are matched. This type of model can keep improving the accuracy of the computation of user preferences. Consonni et al. [45] collect travel-centered mobility data via crowdsourcing. The time spent on the travel is analyzed from a traveler's perspective: the user is asked which activities they have performed during the trip, and which factors have influenced their trip positively or negatively. After analyzing the data, the system can change the perspective on the travel time: instead of considering it simply as *spent* or *wasted*, the system characterizes the travel time in terms of the activities performed, i.e., *fitness*, *enjoyment*, and *productivity*. Boratto et al. [46] analyze and characterize user behavior during journey planning to get insights from different perspectives related to trip search options, i.e., both sorting and selection actions. While selecting different offers, the system can rapidly learn more about the users' preferences.

2.4. Cold Start Problem

The dependency on the existence of historical data is known as the "cold start problem" [47]. Work in [48–52] explores various techniques for determining the best item recommendations for a new user. These techniques employ strategies based on each item's popularity and/or the user profile. However, to provide the user with reliable recommendations, a content-based RS should have access to a sufficient number of user's records that allow it to determine the user's preferences. Therefore, a new user, having very few records, might not receive accurate recommendations. Moreover, although recommending a new user's top popular offers might increase the user's purchase likelihood, it decreases the personalization. Finally, collaborative methods can help to improve personalization, but the recommendations' precision might be quite low.

Clustering algorithms can be used to group users according to their profiles, and the resulting model can predict the cluster of a new user.

In our work, we design and build the RS block proposed by Javadian et al. [8,9]. To do so, we combine collaborative and content-based methods to develop a *hybrid* approach. Given a new user, we do not aim to find a single similar user, but we look for a group of users with similar characteristics instead. Then, we do not directly recommend to new users the travel offers that have been bought by similar groups. Rather, we use a content-based method to build the recommender model for the group and use the model to predict the new user's recommendations.

3. The Hybrid Offer Ranker (THOR)

This section details THOR's implementation and functions. THOR's source code is publicly available to researchers for testing and possible improvement (github.com/Ride2Rail/Learner_Ranker/tree/main/TravelOffer_RecommenderCore, accessed on 10 October 2022).

3.1. Overview

THOR aims to provide ranked travel offers to TC users according to their contextual preferences. Figure 1 provides a high-level overview of the THOR system.

Travel offers and user profiles are collected when TC users send mobility requests, where each request includes the so-called "search options"—i.e., situational preferences such as the desired means of transportation, the maximum number of connections, and so on (see also Table 1). In response, the Travel Solution Aggregator—a third-party application that queries TSPs for travel solutions—returns the list of offers to fulfill the mobility request. For each of the offers returned by the Travel Solution Aggregator, the Offer Categorizer module (OC) computes scores for the following travel categories: *fast*, *reliable*, *cheap*, *comfortable*, *door-to-door*, *environmentally friendly*, *short*, *multitasking*, *social*, *panoramic*, and *healthy* (the implementation of the OC module, developed within the R2R project, is not part of this work). The OC also extracts specific details of each leg of the travel offer (where a leg is a single piece of travel contained in an end-to-end journey), such as *transportation mode*, *seat type*, *carrier*, and *length*. In the rest of this work, we refer to the combination of category scores and leg-level information as the *enriched offer*. In turn, enriched offers are combined with the user's most recent profile information to build the *Ranker*'s input data. The Ranker uses the *personal, contextual preference* model of the user—elaborated by the *Learner*—to predict if they will buy the offer or not. Next, the ranked list of offers is shown to the user. After a travel offer is selected from the list, the user's historical data are updated with the offers in the list (where each offer is tagged as purchased/not purchased), and the Learner module uses the new records to update the user's contextual preference model.

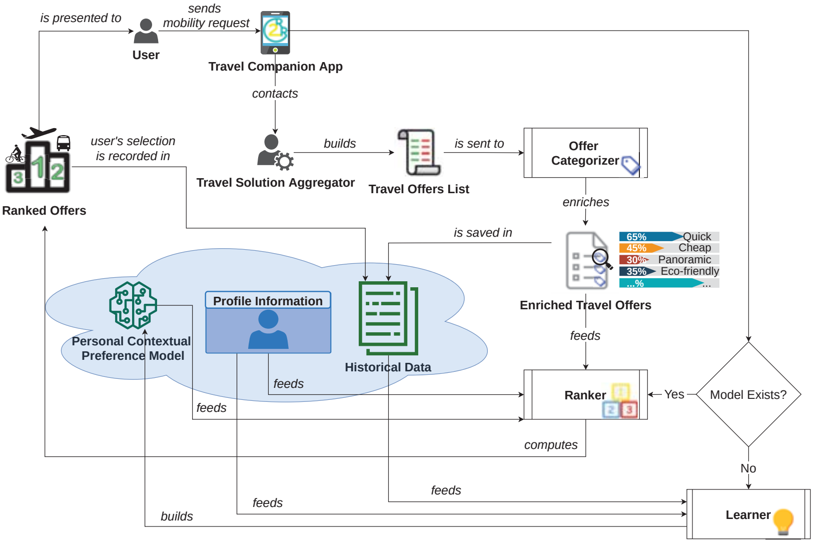


Figure 1. High-level representation of THOR.

We say that a user is *cold* if their historical database contains a number of records smaller than a given threshold (e.g., 100); among cold users, we have *new users*—i.e., users who just registered so the system does not have any historical records for them.

We separate *contextual preference models* into two categories: *user-specific recommender models*—i.e., preference models obtained using data related to a single user—and *cluster-wide recommender models*—i.e., models trained considering data from a set of travelers (instead of a single one) with similar profiles. We recognize users who have similar profiles by building a *cluster model*, which allows us to identify, for each user (even new ones), the cluster to which they belong.

3.2. Data Pre-Processing

In order to prepare the data for the learning and ranking modules, the framework applies the following pre-processing steps.

1. One-Hot Encoding:This step translates the raw textual data into a numerical one-hot version. Moreover, null data are replaced with zeros. Consider, for example, the feature "Profile", which takes one of the following four values: "Basic", "Business", "Family", or "Leisure". During one-hot encoding, "Profile" is split into four features (one for each possible value) that are mutually exclusive—i.e., only one of them can have a value equal to one, while the rest are zero.
2. Information-Less Columns Dropping (ILCD): in this step, the system deletes all columns that have the same value in the dataset. For instance, if the user has never changed their hometown, we can delete it because this feature means nothing to our system. Deleting these columns substantially speeds up the training phase.
3. Data Normalization: Since the scales and magnitudes of the features are not the same, if the original data values are used directly during the prediction phase, their degree of influence is different. The system applies a normalization process through which every feature will have the same influence on the result.

3.3. Learner Module

One of THOR's fundamental blocks is the Learner, which constantly updates users' preferences and builds the most recent contextual preference model for each user individually. Figure 2 details the logic control of the Learner module, and Algorithm 1 provides its pseudo-code.

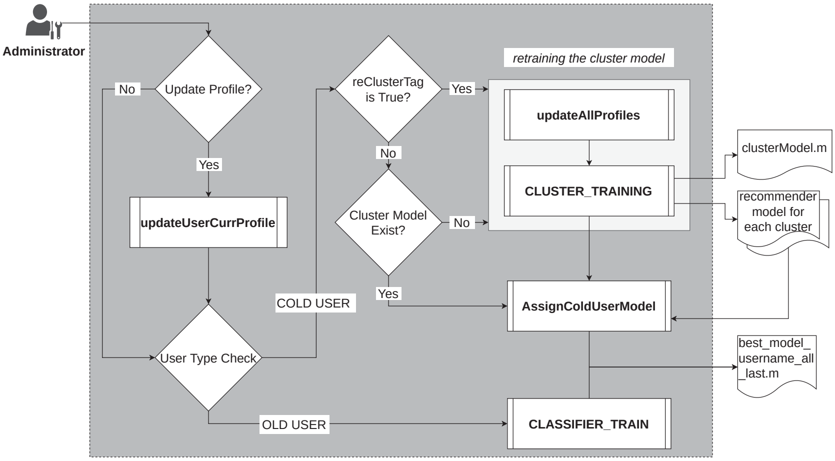


Figure 2. THOR's Training (Learning) Workflow.

An administrator can manually or automatically (according to a schedule) trigger the update of the user models. Before training the model, the administrator can choose to update the current user or not, read all the users' current profiles, and put them all into one table as the input data to the cluster training module. The administrator can also upload other users' profile data supplied by third parties to help the system train a better model at the first stage.

For a cold user, the administrator can re-train the cluster model by setting the value of `reClusterTag` to *true*, and the re-training process will also be completed when the system does not find the cluster model. During the cluster model training phase (which is encapsulated in algorithm `ClusterModelTrain` shown in Algorithm 2), all users' profiles are sent to `CLUSTER_TRAINING`; the system keeps track of the cluster model and of the cluster-wide recommender models. These models are then associated with the cold user based on the cluster to which the user belongs; in this way, we build the cold user's model by training it on the historical data of other—similar—travelers.

For an old user, the administrator uses the `CLASSIFIER_TRAIN` function to train or re-train the user's model and save the user's best model.

Algorithm 1 Learner

```
Input: username; reClusterTag; new profile (optional).
Output: user-specific recommender model or cluster-wide recommender model.
function ModelTrain(username, reClusterTag, new profile)
    user's historical records ← fetch user records from database(username)
    user type ← check the number of user's historical records

    if user new profile is given then
        update the database with new profile
    end if

    if user type is old user then
        user recommender model ← CLASSIFIER_TRAIN(user's historical records)
    else
        if cluster model does not exist OR reClusterTag is True then
            search ranges ← parameters range define
            cluster model, cluster-wide recommender models ← ClusterModelTrain(search ranges)
        end if
        user cluster ← get user cluster(cluster model, username)
        cluster-wide recommender model ← get cluster-wide recommender model(cluster-wide recommender models, user cluster)
        user recommender model ← copy model(cluster-wide recommender model)
    end if
    user-specific recommender model ← save model(user recommender model)
end function
```

Algorithm 2 Cluster Model Training

```
Input: search ranges for each algorithm.
Output: cluster model; cluster-wide recommender models.
function ClusterModelTrain(search ranges)
    user profiles ← fetch all user profiles from the database
    best parameters ← compute best parameters for algorithms(search ranges)
    cluster model ← CLUSTER_TRAINING(best parameters, user profiles)

    for cluster ∈ cluster model do
        cluster historical records ← merge all the users' records in the cluster
        cluster-wide recommender model ← CLASSIFIER_TRAIN(cluster historical records)
    end for
end function
```

3.3.1. Cluster Training for New User

The pseudo-code for the training of the cluster-wide models that occur in case of a new user is presented in Algorithm 2. The system first computes the clusters of users, then, for each cluster, it learns a *cluster-wide recommender model* to solve the problem of recommendations for cold users. This module computes the best parameter setting and uses it to fit the cluster model. Then, it gathers the data for each cluster and trains its models. The function also supports using data supplied by the administrator themselves; otherwise, the system will automatically fetch all valid users' current profiles in the database.

Before fitting the model, the system finds the best parameters by using the parameter-tuning function. To do so, the first step is to define the search range for each parameter in different algorithms. In this work, we used two well-known clustering algorithms: *K-means* [24] and *Density-Based Spatial Clustering of Applications with Noise (DBSCAN)* [25] provided in *Sklearn.cluster* [53]. One of the main strengths of these algorithms is that they can be used easily with any data type, various distance functions, and efficient indexing approaches facilitating the analysis of large datasets [54]. To calculate the optimal number of clusters for *K-means* and DBSCAN, we employ the Elbow method and the Silhouette coefficient, respectively [55], which are briefly recalled here.

In particular, *K-means* aims to minimize the Sum of Squared Error (SSE) between the samples and the mass point of the cluster they belong to; the smaller the value of SSE, the tighter the clusters. Given a set of samples, the value of SSE typically decreases as the number of clusters increases, first slowly, then more steeply, until it again decreases slowly. This gives the SSE curve the shape of an "elbow", and the inflection point (i.e., the elbow) corresponds to the number of clusters that offers the best performance for the algorithm. The Elbow method allows us to determine the inflection point of the SSE curve, which corresponds to the optimal number of clusters to configure *K-means*.

To obtain more precise clustering results with DBSCAN, instead, we calculate the Silhouette coefficient. More precisely, the best coefficient is obtained when the distances among points in the same cluster (resp., different clusters)—i.e., the *cohesion* (resp., the *separation*)—are as small (resp., as big) as possible.

Comparing *K-means* and DBSCAN, although *K-means* requires a prototype-based concept of a cluster (i.e., the number of clusters and their initial centroids), it is useful for sparse and high dimensional data. On the other hand, DBSCAN is powerful in dealing with noises, although it does not perform very well for high dimensional data [56].

After obtaining the best parameters for the algorithms (e.g., the number of clusters), we use `CLUSTER_TRAINING` to compute the set of clusters of users with similar profiles. Next, for each cluster, we combine the historical records of all the users in the cluster and feed them to the `CLASSIFIER_TRAIN` module to build the cluster-wide recommender model.

Finally, when a new user registers to the system, THOR uses their profile information to compute the cluster to which the user belongs and associates the corresponding cluster-wide recommender model with the new user.

3.3.2. User Recommender Model Training

The `CLASSIFIER_TRAIN` function, whose pseudo-code is shown in Algorithm 3, is the core mechanism for building both user-specific and cluster-wide recommender models. More precisely, to train user-specific (resp., cluster-wide) recommender models, function `CLASSIFIER_TRAIN` receives as input the records of a single user (resp., of all users that belong to the cluster). The function first finds the best parameter setting, then uses it to build the recommender model.

To capture the user context, each user record is made of the most recent user profile information (*profile*), the enriched travel offer (*travel offer*, which includes the information whether it was purchased or not), and the search options (*request*) that were used in the mobility request to which the travel offers are the reply.

Algorithm 3 Recommender Model Training

```
Input: user records, i.e., the most recent profile information (profile), purchased and not-
    purchased enriched travel offers (travel offers), as well as their corresponding search
    options (requests)
Output: recommender model.
function CLASSIFIER_TRAIN(user records)
    train data ← data preprocessing(user records)
    search ranges ← parameters range define

    for algo ∈ [KNN, SVC, DT, LR, RF] do
        temp best model, score ← BSCV(algo, train data, search ranges)
        best model ← compare scores of the models
    end for
    recommender model ← save the model in file(best model)
end function
```

To create the training dataset, the system takes the received user records and first performs a pre-processing step (see Section 3.2). Then, we use the data to find the best algorithm and best parameters and then save the final model.

Knowing whether each offer was bought or not allows the system to solve a classification problem, where each new travel offer is classified as "buy" or "not buy" depending on the current profile. More precisely, for each new offer, the system predicts the probability that the user will buy it and considers this value as the score of the offer; finally, offers are ranked according to their scores (i.e., the probability that they will be bought).

To compute the prediction, and in particular to build the personal recommender models, we use popular classification algorithms, all provided by the *Sklearn* package [53]: KNN [19], SVC [20], DT [21], RF [22], and LR [23]. The reason for choosing such algorithms instead of, say, neural networks is that they require much fewer data points, which is a crucial requirement in our case for building personal models [57].

Let us briefly point out the main advantages and disadvantages of each algorithm. KNN is non-parametric and does not make any prior assumption on the data distribution; however, it can exhibit poor performance for high dimensional data and in presence of irrelevant features. SVC is quite robust with respect to the behavior of observations that are far from the decision boundary, but it is not suitable for large data sets. Considering DT, although its results are interpretable, it is sometimes less accurate compared to the other algorithms. RF requires higher training time than other algorithms, but it is suitable when the dataset is large and interpretability is not a major concern. LR is very efficient to train and does not make any assumptions about distributions of classes in the feature space; however, if the number of records is much smaller than the number of features, it may result in overfitting [58].

The set of chosen algorithms covers many cases and situations (e.g., some algorithms work well with small datasets, others are better with big ones); THOR chooses the best-performing algorithm depending on the current training dataset, which ensures the quality of the recommendations. Therefore, a general evaluation method is needed to automatically compare the algorithms among them and find the best fit for the dataset. In this regard, the most popular one is Bayes Search Cross-Validation (BSCV) [59]. Grid Search Cross-Validation (GSCV) [60] is another appropriate method, especially for KNN; however, BSCV can be used for all the algorithms employed in this work. BSCV updates the current best model during each iteration and changes parameter settings according to the search ranges. If the parameter setting is invalid—i.e., the combination of values does not fit the algorithm—the system discards it and searches for another one. The best model for the current user, which depends on the best score given by BSCV, is then generated. Finally, we store the best model to be used in the future.

3.4. Ranker

The Ranker (shown in Algorithm 4) is the main component to get the recommendation results for a user. Its core is represented by function `CLASSIFIER_RESPONSE`, which, for each travel offer, computes the travel offer's score (i.e., the probability that the user will buy that travel offer) using the user's recommender model. The Ranker assumes that a recommender model has already been assigned to the user; if not, prior to the steps described in the following, it invokes the Learner block shown in Algorithm 1.

Initially, the Ranker receives the corresponding preference model of the user, i.e., the user-specific model in the case of the old user or the cluster-wide model in the case of a cold user. After getting the model, the Ranker uses function `CLASSIFIER_RESPONSE` to predict if the user will buy or not any of the offers and saves the results.

Algorithm 4 RANKER

Input: user's recommender model (model), most recent profile information (profile), search options (request), list of enriched travel offers (travel offers)
Output: ranked list of travel offers (ranked offers)

```
function RANKER(model, profile, request, travel offers)
    scored_travel_offers = []

    for offer $\in$ travel offers do
        (offer_id, prediction's score) $\leftarrow$ CLASSIFIER_RESPONSE(model, profile, request, offer)
        scored_travel_offers.append((offer_id, prediction's score))
    end for
    ranked offers $\leftarrow$ sort(scored_travel_offers)
end function
```

More precisely, the `CLASSIFIER_RESPONSE` function preprocesses the input data and makes the predictions by using the user recommender model. Each travel offer in the set (which has been enriched by the OC) is joined with the user's current profile and with the information concerning the mobility request to form the raw data, which is fed as input to the prediction function. Using the recommender model associated with the user, we obtain, for each offer, the probability that it will be bought by the user and use that as the offer score.

Finally, the Ranker sorts the travel offers according to their scores and presents them to the user.

3.5. User Feedback

To have better accuracy for the recommendations, the system needs to update the user's historical records continuously. The collected user feedback consists of the purchasing decision after the recommendation: the user will typically buy an offer from the list shown to them and ignore the rest. However, if the system recommends too many offers to the user, part of them will not be seen by the user, and the model will be updated based only on the best recommendations. Hence, the top 30 recommended offers, plus the one bought by the user, are recorded in the historical dataset; this number may have little impact on a large dataset, but it may change the results a lot for a small dataset, especially for a new user.

4. Experimental Results

To evaluate the features of the THOR system, we carried out a few experiments with the two following goals. Since, as mentioned in Section 1, to the best of our knowledge in the literature, there are no other works directly comparable to THOR, a direct comparison of the performance of THOR with that of similar tools is not possible. However, as described in Section 4.1, we first aim to quantitatively evaluate the accuracy of THOR (and in particular of its classifiers) when predicting the category (bought/not bought) of each travel offer received. Second (Section 4.2), we evaluate its ability to suitably rank the sets of offers received, in particular when the user context (especially the user profile) changes. Since there are no suitable metrics to assess the ranking mechanism in a quantitative manner (e.g., through a notion of accuracy) [61], we designed a controlled experiment to evaluate the quality of the rankings instead.

4.1. Validating the Classifiers

To test THOR, we need an existing dataset. At first, we attempted to find a suitable publicly available dataset for this purpose; although the available datasets had some information about travelers in Europe and public transport facilities, none of them could match (even partially) with the Shift2Rail's data structure. To do so, we designed a data generation pipeline using some rules to avoid having a completely random dataset. An advantage of this line of work is that knowing the distribution of the dataset allows us to validate the results. Table 1 provides the main features that we used during our experiment. "Profile" encompasses a set of features such as age and list of loyalty cards saved in the user's profile. "Search Options (Search)" includes a set of features related to the submitted mobility request by the user (e.g., preferred transportation type). Lastly, "Travel Offer (Offer)" is the set of features extracted from the offer to be ranked (e.g., number of legs). We generated a dataset with 1000 unique user profiles and a total of 101,028 records (approximately 100 records—i.e., travel offers—per user, so each user has enough records to be considered "old", and a user-specific recommender model can be trained for them).

The dataset was generated randomly, but we introduced a few rules to avoid making it uniform and to create "hidden patterns" to check whether our recommendation mechanism was able to pick them up. For example, in the generated dataset, 40% of the travel offers associated with users who are Persons With Reduced Mobility (PRM) have a single leg (i.e., it holds that "Legs Number = 1"), and 80% are *short*. Then, we associated with each offer a "Bought" tag that depended on one of the features of the offers (e.g., "short"), and we finally randomly changed 10% of the *bought* (resp., *not bought*) travel offers to *not bought* (resp., *bought*). In this way, the *bought* travel offers are not evenly distributed across the dataset. We use 80% of the records in the dataset for training the classifiers and 20% for testing their accuracy. A perfect classifier would be able to correctly guess which, among the 20% travel offers used for testing are *bought* and which are not.

Figure 3 reports the box plot of the time required, for each user, to train each algorithm (training was carried out on a MacBook Pro with CPU Core i9 and 16 GB RAM). If we consider, for each user, the cumulative time that it takes to train all algorithms (i.e., the time to train KNN on the user records, plus the time to train SVC on the same user records, and so on), then the sum of the average training times (i.e., the values highlighted as yellow lines in Figure 3) is around 1 s, and the sum of the maximum required training times (i.e., the highest dots in Figure 3) is close to 2.1 s. Moreover, there exists a minor computational cost (a few milliseconds) to retrieve the learned model and predict the scores of the travel offers.

After getting all the best models of the test users—i.e., the combination of parameters for the algorithm with the best performance—we recorded the accuracy value of each algorithm. The accuracy is obtained by considering true positives (resp., true negatives) as the travel offers that are tagged as "purchased" (resp., "not purchased") in the dataset and for which THOR returned a correct prediction of 1 (resp., 0) by computing the following quantity:

$$\frac{\textit{true positives} + \textit{true negatives}}{\textit{number of records}} \times 100$$

Figure 4 details the accuracies of each algorithm for all the users. The last box plot provides the scores obtained from the best algorithm for each user. We optimize the model by selecting the best algorithm automatically. The average accuracy is equal to 91%; the highest accuracy is equal to 100%, while the lowest accuracy is equal to 72%.

Figure 5 shows the proportion of the models obtained by each algorithm as the optimal model. We can see that LR is the most suitable and KNN and RF are the least suitable algorithms for our test data.

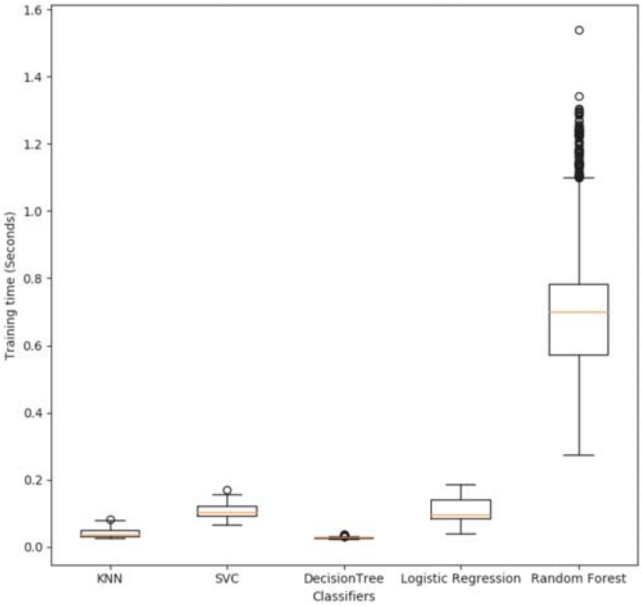


Figure 3. The classifiers' training time for all the users.

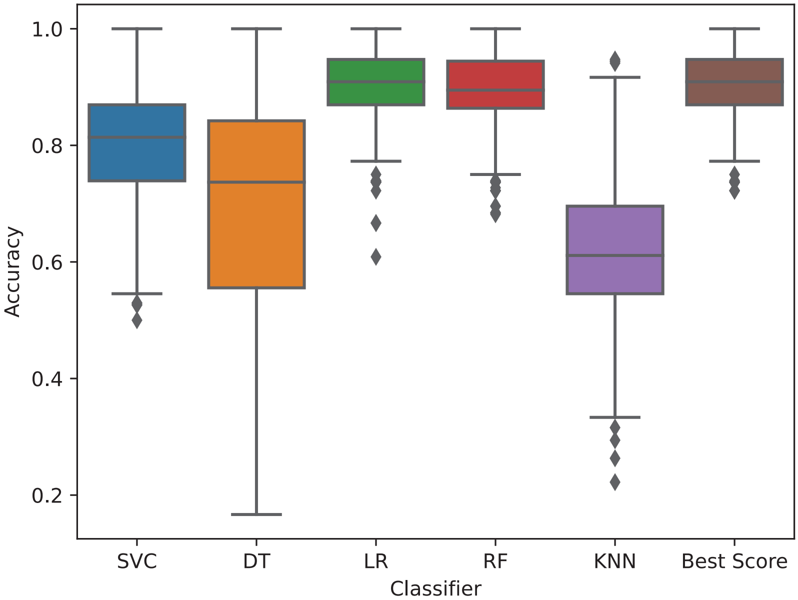


Figure 4. The accuracies of each algorithm for all the users. The last box plot provides the scores obtained from the best algorithm for each user.

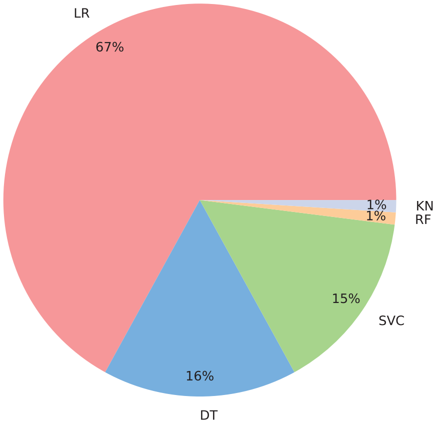


Figure 5. The probability distribution of different algorithms is used as the best model.

4.2. Validating the Ranker

The procedure discussed in Section 4.1 focuses on the validation of the performance of the classifiers, and it is not well suited to validate the proposed ranking mechanism. Therefore, in this section, we present a scenario-based validation procedure to evaluate how the system ranks the travel offers for a specific user depending on the user context. To do so, we manually created a data set for a fictional person named Sarah. The dataset is such that Sarah selects different types of travel offers depending on her context. We expect that THOR detects contextual preferences of Sarah and ranks the travel offers accordingly.

Sarah is a 35-year-old assistant professor at the Polytechnic University of Atlantis (EU). She lives in Atlantis, and due to the research projects she is working on, she frequently travels to many European countries to meet the project's partners. She has been using the TC application for some time to find the most suitable travel offers; therefore, the system has some records of her choices in different contexts. Sarah's travel records can be divided into two typical situations. The first set of records concerns a period in which she is healthy—i.e., she does not have any issues that could make her a PRM. In these cases, she typically selects business class trips, window seats, does not put any constraints (in the search options) on transfer duration and number of stops, and so on. Since she cares a lot about global warming, she prioritizes travel offers with the minimum carbon footprint—i.e., those with the highest environmentally-friendly score even if they are not cheap or are not quick—and this results in her not choosing private taxis and similar options with low environmentally-friendly scores. Moreover, she likes travel offers that have high panoramic scores; therefore, she mostly avoids travel offers that cannot fulfill this preference (e.g., underground transportation). In addition, this results in Sarah choosing mostly travel offers with many legs (changes), low door-to-door scores, and sometimes low comfort scores.

The second set of records is related to a period in which she needed to use a wheelchair due to a car accident. As a result, she updated her profile information to mention her new PRM status, and she started favoring, in her selections, travel offers with characteristics such as door-to-door, comfortable, quick, and so on, even though they are totally different from what she used to choose. As a result, her personal recommender model is updated automatically using the new profile information. For instance, to satisfy the door-to-door requirement, as a PRM, she always chooses travel offers that include taxis on the first and last legs of the trip. Other changes in her selections related to modes of transportation include not choosing ship and bus trips, which she used to choose before. Moreover, if available, she chooses large seats. In other words, Sarah selects offers that are more suitable for PRM people. These offers are now part of her user records, where each record includes the profile information under which the choices were made.

Using these records (first and second sets), we trained a personal recommender model (using again 80% of the records for training and 20% for testing) that, on average, showed 90% accuracy when predicting travel offers that Sarah will buy. This shows that even when the context of the user—hence their patterns of behavior—changes, if the profile information is suitably updated, the system is able to adapt to these patterns and ranks travel offers according to the user's contextual preferences.

Since the purpose of this experiment is to test the ranking mechanism, we assumed a situation in which Sarah becomes healthy again and does not use a wheelchair anymore. For a given mobility request at this time, we generated three potential travel offers: travel offer A, with characteristics most similar to the time when she was not a PRM, travel offer B, with the characteristics most similar to the time when she was a PRM, and travel offer C, with blended characteristics.

As expected, the Ranker could successfully rank travel offer A as the first, C as the second, and B as the third.

5. Conclusions and Future Work

In this work, we designed and implemented The Hybrid Offer Ranker (THOR), a personalized, context-aware, hybrid recommender system that employs various state-of-the-art classification algorithms (DT, KNN, LR, RF, and SVC) to tackle RQ1—i.e., to learn the travelers' contextual preferences and rank travel offers accordingly. Moreover, to address RQ2, we used the *K*-means and DBSCAN clustering algorithms to deal with the cold-start problem for new users. To tune the algorithms' hyper-parameters, we designed a grid search (GSCV) mechanism which finds the set of hyper-parameters automatically. THOR keeps learning as soon as a new record or user is registered in the system, thus, keeping the recommender models up-to-date. Notice that the modular design of THOR allows the integration of classification and clustering algorithms other than those used in this work.

Since the TC application is under development, there exists no real data for testing purposes. Hence, to test the performance of THOR, we automatically synthesized a dataset of 1000 unique user profiles and more than 100,000 travel offers, and we also manually built a controlled dataset for a specific user according to a predefined scenario. On both datasets, THOR showed an accuracy higher than 90%. These are promising results that show that THOR can be integrated with other TC modules to be tested in demo sites.

In the future, we plan to extend/improve THOR in several directions. As soon as we acquire enough real data, we plan to test the performance of THOR by using various feature selection methods [62–64] which potentially might result in reducing the complexity of the model while improving its accuracy.

Additionally, we plan to use deep learning approaches, such as the multimodal deep learning methods presented in [65–67], for training the cluster-wide recommender models. In addition, various transfer learning methods [68] could be exploited to reduce the training time while updating the cluster-wide recommender models.

We plan to build the social media core proposed in [9] as a tool [69,70] to characterize urban mobility patterns [71,72]. Moreover, the social media core will enable the system's stakeholders to understand user preferences during online events [73,74] which bring many travelers to specific European cities. Consequently, we plan to design predictive models as proposed in [75] to predict the popularity of online content generated by the stakeholders to maximize the visibility and popularity of their news and advertisements. Last but not least, the social media core will enable us to extract the conversation graphs [76,77] around specific topics, build conversational agents [78], and facilitate customer relationship management.

Author Contributions: A.J.S.: Conceptualization, Methodology, Software, Validation, Investigation, Data Curation, Writing—Original Draft, Writing—Review & Editing, Visualization. M.S.: Conceptualization, Methodology, Software, Validation, Investigation, Data Curation, Writing—Original Draft, Writing—Review & Editing, Visualization. C.G.: Methodology, Software, Validation, Investigation, Data Curation, Writing—Original Draft, Writing—Review & Editing, Visualization. M.R.: Supervision, Funding acquisition, Conceptualization, Methodology, Validation, Investigation, Writing—Review & Editing. F.S.: Supervision, Conceptualization, Methodology, Validation, Investigation, Writing—Review & Editing. L.T.: Supervision, Conceptualization, Methodology, Validation, Investigation, Writing—Review & Editing. All authors have read and agreed to the published version of the manuscript.

Funding: Work supported by Shift2Rail and the EU H2020 research and innovation programme under grant agreement No: 881825 (RIDE2RAIL).

Data Availability Statement: THOR's source code and the generated datasets are publicly available at: https://github.com/Ride2Rail/Learner_Ranker/tree/main/TravelOffer_RecommenderCore. (accessed on 10 October 2022).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Brambilla, M.; Javadian Sabet, A.; Masciadri, A. Data-driven user profiling for smart ecosystems. In *Smart Living between Cultures and Practices. A Design Oriented Perspective*; Mandragora: Milan, Italy, 2019; pp. 84–98.
2. Ferreira, J.C.; Martins, A.L.; da Silva, J.V.; Almeida, J. T2*—Personalized Trip Planner. In Proceedings of the Ambient Intelligence–Software and Applications—8th International Symposium on Ambient Intelligence (ISAmI 2017), Porto, Portugal, 21–23 June 2017; De Paz, J.F.; Julián, V.; Villarrubia, G.; Marreiros, G.; Novais, P., Eds.; Springer International Publishing: Cham, Switzerland, 2017; pp. 167–175.
3. Chen, X.; Liu, Q.; Qiao, X. Approaching Another Tourism Recommender. In Proceedings of the 2020 IEEE 20th International Conference on Software Quality, Reliability and Security Companion (QRS-C), Macau, China, 11–14 December 2020; pp. 556–562. [CrossRef]
4. Ricci, F.; Rokach, L.; Shapira, B. Introduction to Recommender Systems Handbook. In *Recommender Systems Handbook*; Springer: Boston, MA, USA, 2011; pp. 1–35. [CrossRef]
5. Adomavicius, G.; Tuzhilin, A. Context-Aware Recommender Systems. In *Recommender Systems Handbook*; Springer: Boston, MA, USA, 2011; pp. 217–253. [CrossRef]
6. Abbar, S.; Bouzeghoub, M.; Lopez, S. Context-aware recommender systems: A service-oriented approach. In Proceedings of the VLDB PersDB Workshop, Lyon, France, 28 August 2009; pp. 1–6.
7. Villegas, N.M.; Sánchez, C.; Díaz-Cely, J.; Tamura, G. Characterizing context-aware recommender systems: A systematic literature review. *Knowl.-Based Syst.* **2018**, *140*, 173–200. [CrossRef]
8. Sabet, A.J.; Rossi, M.; Schreiber, F.A.; Tanca, L. Context Awareness in the Travel Companion of the Shift2Rail Initiative. In Proceedings of the 28th Italian Symposium on Advanced Database Systems, CEUR-WS.org, CEUR Workshop Proceedings, Villasimius, Sardinia, Italy, 21–24 June 2020; Volume 2646, pp. 202–209.
9. Javadian Sabet, A.; Rossi, M.; Schreiber, F.A.; Tanca, L. Towards Learning Travelers' Preferences in a Context-Aware Fashion. In *Proceedings of the Ambient Intelligence—Software and Applications*; Springer International Publishing: Cham, Switzerland, 2021; pp. 203–212. [CrossRef]
10. Dey, A. Understanding and Using Context. *Pers. Ubiquitous Comput.* **2001**, *5*, 4–7. [CrossRef]
11. Bolchini, C.; Curino, C.A.; Quintarelli, E.; Schreiber, F.A.; Tanca, L. A data-oriented survey of context models. *ACM Sigmod Rec.* **2007**, *36*, 19–26. [CrossRef]
12. Alegre, U.; Augusto, J.C.; Clark, T. Engineering context-aware systems and applications: A survey. *J. Syst. Softw.* **2016**, *117*, 55–83. [CrossRef]
13. Hong, J.Y.; Suh, E.H.; Kim, S.J. Context-aware systems: A literature review and classification. *Expert Syst. Appl.* **2009**, *36*, 8509–8522. [CrossRef]
14. Ng, P.C.; She, J.; Cheung, M.; Cebulla, A. An Images-Textual Hybrid Recommender System for Vacation Rental. In Proceedings of the 2016 IEEE Second International Conference on Multimedia Big Data (BigMM), Taipei, Taiwan, 20–22 April 2016; pp. 60–63. [CrossRef]
15. Sebastia, L.; Garcia, I.; Onaindia, E.; Guzman, C. e-Tourism: A Tourist Recommendation and Planning Application. In Proceedings of the 2008 20th IEEE International Conference on Tools with Artificial Intelligence, Dayton, OH, USA, 3–5 November 2008; Volume 2, pp. 89–96. [CrossRef]
16. Kbaier, M.E.B.H.; Masri, H.; Krichen, S. A Personalized Hybrid Tourism Recommender System. In Proceedings of the 2017 IEEE/ACS 14th International Conference on Computer Systems and Applications (AICCSA), Hammamet, Tunisia, 30 October–3 November 2017; pp. 244–250. [CrossRef]

17. Bolchini, C.; Quintarelli, E.; Tanca, L. CARVE: Context-aware automatic view definition over relational databases. *Inf. Syst.* **2013**, *38*, 45–67. [CrossRef]
18. Arnaoutaki, K.; Bothos, E.; Magoutas, B.; Aba, A.; Esztergár-Kiss, D.; Mentzas, G. A Recommender System for Mobility-as-a-Service Plans Selection. *Sustainability* **2021**, *13*, 8245. [CrossRef]
19. Jaafar, H.B.; Mukahar, N.B.; Binti Ramli, D.A. A methodology of nearest neighbor: Design and comparison of biometric image database. In Proceedings of the 2016 IEEE Student Conference on Research and Development (SCOReD), Kuala Lumpur, Malaysia, 30 August 2016; pp. 1–6. [CrossRef]
20. Lan, L.S. M-SVC (mixed-norm SVC)—A novel form of support vector classifier. In Proceedings of the 2006 IEEE International Symposium on Circuits and Systems, Kos, Greece, 21–24 May 2006; p. 3264. [CrossRef]
21. Yu, Y.; Fu, Z.-L.; Zhao, X.-H.; Cheng, W.-F. Combining Classifier Based on Decision Tree. In Proceedings of the 2009 WASE International Conference on Information Engineering, Taiyuan, China, 10–11 July 2009; Volume 2, pp. 37–40. [CrossRef]
22. Bingzhen, Z.; Xiaoming, Q.; Hemeng, Y.; Zhubo, Z. A Random Forest Classification Model for Transmission Line Image Processing. In Proceedings of the 2020 15th International Conference on Computer Science Education (ICCSE), Delft, The Netherlands, 18–22 August 2020; pp. 613–617. [CrossRef]
23. Zou, X.; Hu, Y.; Tian, Z.; Shen, K. Logistic Regression Model Optimization and Case Analysis. In Proceedings of the 2019 IEEE 7th International Conference on Computer Science and Network Technology (ICCSNT), Dalian, China, 19–20 October 2019; pp. 135–139. [CrossRef]
24. Lu, S.; Yu, H.; Wang, X.; Zhang, Q.; Li, F.; Liu, Z.; Ning, F. Clustering Method of Raw Meal Composition Based on PCA and Kmeans. In Proceedings of the 2018 37th Chinese Control Conference (CCC), Wuhan, China, 25–27 July 2018; pp. 9007–9010. [CrossRef]
25. Smiti, A.; Elouedi, Z. DBSCAN-GM: An improved clustering method based on Gaussian Means and DBSCAN techniques. In Proceedings of the 2012 IEEE 16th International Conference on Intelligent Engineering Systems (INES), Lisbon, Portugal, 13–15 June 2012; pp. 573–578. [CrossRef]
26. Golightly, D.; Comerio, M.; Consonni, C.; Vaghi, C.; Pistilli, G.; Rizzi, G.; Di Pasquale, G.; Palacin, R.; Boratto, L.; Scrocca, M. Ride2Rail: Integrating ridesharing for attractive multimodal rail journeys. In Proceedings of the World Congress Rail Research 2022, Birmingham, UK, 6–10 June 2022.
27. Sadeghi, M.; Buchníček, P.; Carenini, A.; Corcho, O.; Gogos, S.; Rossi, M.; Santoro, R. SPRINT: Semantics for PerfoRmant and scalable INteroperability of multimodal Transport. In Proceedings of the TRA, Helsinki, Finland, 27–30 April 2020; pp. 1–10.
28. Hosseini, M.; Kalwar, S.; Rossi, M.G.; Sadeghi, M. Automated mapping for semantic-based conversion of transportation data formats. In Proceedings of the 1st International Workshop On Semantics For Transport, Karlsruhe, Germany, 9 September 2019; Volume 2447, pp. 1–6.
29. Kalwar, S.; Sadeghi, M.; Javadian Sabet, A.; Nemirovskiy, A.; Rossi, M.G. SMART: Towards Automated Mapping between Data Specifications. In Proceedings of the 33rd International Conference on Software Engineering and Knowledge Engineering, SEKE 2021, KSIR Virtual Conference Center, Pittsburgh, PA, USA, 1–10 July 2021. [CrossRef]
30. Carenini, A.; Dell'Arciprete, U.; Gogos, S.; Kallehbasti, M.M.P.; Rossi, M.; Santoro, R. ST4RT—Semantic Transformations for Rail Transportation. In Proceedings of the TRA 2018, Vienna, Austria, 16–19 April 2018; pp. 1–10.
31. Alobaid, A.; Garijo, D.; Poveda-Villalón, M.; Santana-Perez, I.; Fernández-Izquierdo, A.; Corcho, O. Automating ontology engineering support activities with OnToology. *J. Web Semant.* **2019**, *57*, 100472. [CrossRef]
32. Sadeghi, M.; Sartor, L.; Rossi, M. A Semantic-Based Access Control Mechanism for Distributed Systems. In Proceedings of the 36th Annual ACM Symposium on Applied Computing, SAC' 21, Gwangju, Korea, 22–26 March 2021; Association for Computing Machinery, New York, NY, USA, 2021; pp. 1864–1873. [CrossRef]
33. Adomavicius, G.; Tuzhilin, A. Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. *IEEE Trans. Knowl. Data Eng.* **2005**, *17*, 734–749. [CrossRef]
34. Cai, Y.; Leung, H.F.; Li, Q.; Min, H.; Tang, J.; Li, J. Typicality-Based Collaborative Filtering Recommendation. *IEEE Trans. Knowl. Data Eng.* **2014**, *26*, 766–779. [CrossRef]
35. Valliyammai, C.; PrasannaVenkatesh, R.; Vennila, C.; Krishnan, S.G. An intelligent personalized recommendation for travel group planning based on reviews. In Proceedings of the 2016 Eighth International Conference on Advanced Computing (ICoAC), Chennai, India, 19–21 January 2017; pp. 67–71. [CrossRef]
36. Cao, Y.; Li, Y. An intelligent fuzzy-based recommendation system for consumer electronic products. *Expert Syst. Appl.* **2007**, *33*, 230–240. [CrossRef]
37. Agrawal, R.; Srikant, R. Fast algorithms for mining association rules. In Proceedings of the 20th International Conference on Very Large Data Bases, Santiago de Chile, Chile, 12–15 September 1994; Volume 1215, pp. 487–499.
38. Lorenzi, F.; Loh, S.; Abel, M. PersonalTour: A Recommender System for Travel Packages. In Proceedings of the 2011 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology, Lyon, France, 22–27 August 2011; Volume 2, pp. 333–336. [CrossRef]
39. Sabet, A.J.; Gopalakrishnan, S.; Rossi, M.; Schreiber, F.A.; Tanca, L. Preference Mining in the Travel Domain. In Proceedings of the 2021 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA), Dalian, China, 28–30 June 2021; pp. 358–365. [CrossRef]

40. Fang, G.S.; Kamei, S.; Fujita, S. Automatic Generation of Temporal Feature Vectors with Application to Tourism Recommender Systems. In Proceedings of the 2016 Fourth International Symposium on Computing and Networking (CANDAR), Hiroshima, Japan, 22–25 November 2016; pp. 676–680. [CrossRef]
41. Coelho, J.; Nitu, P.; Madiraju, P. A Personalized Travel Recommendation System Using Social Media Analysis. In Proceedings of the 2018 IEEE International Congress on Big Data (BigData Congress), Seattle, DC, USA, 10–13 December 2018; pp. 260–263. [CrossRef]
42. Fararni, K.A.; Nafis, F.; Aghoutane, B.; Yahyaouy, A.; Riffi, J.; Sabri, A. Hybrid recommender system for tourism based on big data and AI: A conceptual framework. *Big Data Min. Anal.* **2021**, *4*, 47–55. [CrossRef]
43. Shekari, M.; Sabet, A.J.; Guan, C.; Rossi, M.; Schreiber, F.A.; Tanca, L. Personalized Context-Aware Recommender System for Travelers. In Proceedings of the 30th Italian Symposium on Advanced Database Systems, SEBD 2022, Tirrenia, Italy, 19–22 June 2022; Amato, G., Bartalesi, V., Bianchini, D., Gennaro, C., Torlone, R., Eds.; 2022, Volume 3194, pp. 497–504.
44. Basile, S.; Consonni, C.; Manca, M.; Boratto, L. Matching User Preferences and Behavior for Mobility. In Proceedings of the 31st ACM Conference on Hypertext and Social Media, HT '20, Online, 13–15 July 2020; Association for Computing Machinery: New York, NY, USA, 2020; pp. 141–150. [CrossRef]
45. Consonni, C.; Basile, S.; Manca, M.; Boratto, L.; Freitas, A.; Kovacikova, T.; Pourhashem, G.; Cornet, Y. What's Your Value of Travel Time? Collecting Traveler-Centered Mobility Data via Crowdsourcing. *arXiv* **2021**, arXiv:cs.CY/2104.05809.
46. Boratto, L.; Manca, M.; Lugano, G.; Gogola, M. Characterizing user behavior in journey planning. *Computing* **2020**, *102*. [CrossRef]
47. Schein, A.I.; Popescul, A.; Ungar, L.H.; Pennock, D.M. Methods and metrics for cold-start recommendations. In Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in IR, Tampere, Finland, 11–15 August 2002; pp. 253–260.
48. Rashid, A.M.; Albert, I.; Cosley, D.; Lam, S.K.; McNee, S.M.; Konstan, J.A.; Riedl, J. Getting to Know You: Learning New User Preferences in Recommender Systems. In Proceedings of the 7th International Conference on Intelligent User Interfaces, IUI '02, College Station, TX, USA, 21–25 March 2022; Association for Computing Machinery: New York, NY, USA, 2002; pp. 127–134. [CrossRef]
49. Guo, G. Integrating trust and similarity to ameliorate the data sparsity and cold start for recommender systems. In Proceedings of the 7th ACM conference on Recommender Systems, Hong Kong, 12–16 October 2013; pp. 451–454.
50. Yu, K.; Schwaighofer, A.; Tresp, V.; Xu, X.; Kriegel, H.P. Probabilistic memory-based collaborative filtering. *IEEE Trans. Knowl. Data Eng.* **2004**, *16*, 56–69. [CrossRef]
51. Ghodsad, P.R.; Chatur, P.N. Handling User Cold-Start Problem for Group Recommender System Using Social Behaviour Wise Group Detection Method. In Proceedings of the 2018 International Conference on Research in Intelligent and Computing in Engineering (RICE), San Salvador, El Salvador, 22–24 August 2018; pp. 1–5. [CrossRef]
52. Sang, A.; Vishwakarma, S.K. A ranking based recommender system for cold start data sparsity problem. In Proceedings of the 2017 Tenth International Conference on Contemporary Computing (IC3), Noida, India, 10–12 August 2017; pp. 1–3. [CrossRef]
53. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
54. Schubert, E.; Sander, J.; Ester, M.; Kriegel, H.P.; Xu, X. DBSCAN Revisited, Revisited: Why and How You Should (Still) Use DBSCAN. *ACM Trans. Database Syst.* **2017**, *42*, 3068335. [CrossRef]
55. James, G.; Witten, D.; Hastie, T.; Tibshirani, R. *An Introduction to Statistical Learning*; Springer: Berlin, Germany, 2013; Volume 112.
56. Kanagala, H.K.; Jaya Rama Krishnaiah, V. A comparative study of K-Means, DBSCAN and OPTICS. In Proceedings of the 2016 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, 7–9 January 2016; pp. 1–6. [CrossRef]
57. Kumar, R.; Verma, R. Classification algorithms for data mining: A survey. *Int. J. Innov. Eng. Technol. (Ijiet)* **2012**, *1*, 7–14.
58. Narayanan, U.; Unnikrishnan, A.; Paul, V.; Joseph, S. A survey on various supervised classification algorithms. In Proceedings of the 2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS), Chennai, India, 1–2 August 2017; pp. 2118–2124. [CrossRef]
59. Chung, T.H.; Burdick, J.W. Analysis of Search Decision Making Using Probabilistic Search Strategies. *IEEE Trans. Robot.* **2012**, *28*, 132–144. [CrossRef]
60. Huang, Q.; Mao, J.; Liu, Y. An improved grid search algorithm of SVR parameters optimization. In Proceedings of the 2012 IEEE 14th International Conference on Communication Technology, Chengdu, China, 19–21 October 2012; pp. 1022–1026. [CrossRef]
61. Shani, G.; Gunawardana, A. Evaluating recommendation systems. In *Recommender Systems Handbook*; Springer: Berlin, Germany, 2011; pp. 257–297.
62. Hosseini, M. Feature Selection for Microarray Classification Problems. Master's Thesis, Politecnico di Milano, Milan, Italy, 2018.
63. Brankovic, A.; Hosseini, M.; Piroddi, L. A Distributed Feature Selection Algorithm Based on Distance Correlation with an Application to Microarrays. *ACM Trans. Comput. Biol. Bioinform.* **2019**, *16*, 1802–1815. [CrossRef]
64. Rajeswari, K. Feature selection by mining optimized association rules based on apriori algorithm. *Int. J. Comput. Appl.* **2015**, *119*, 30–34. [CrossRef]
65. Hong, D.; Gao, L.; Yokoya, N.; Yao, J.; Chanussot, J.; Du, Q.; Zhang, B. More Diverse Means Better: Multimodal Deep Learning Meets Remote-Sensing Imagery Classification. *IEEE Trans. Geosci. Remote Sens.* **2021**, *59*, 4340–4354. [CrossRef]

66. Hong, D.; Gao, L.; Yao, J.; Zhang, B.; Plaza, A.; Chanussot, J. Graph Convolutional Networks for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2021**, *59*, 5966–5978. [CrossRef]
67. Wu, X.; Hong, D.; Chanussot, J. Convolutional Neural Networks for Multimodal Remote Sensing Data Classification. *IEEE Trans. Geosci. Remote. Sens.* **2022**, *60*, 1–10. [CrossRef]
68. Pan, S.J.; Yang, Q. A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.* **2009**, *22*, 1345–1359. [CrossRef]
69. Cea-Morán, J.J.; González-Briones, A.; De La Prieta, F.; Prat-Pérez, A.; Prieto, J. Extraction of Travellers' Preferences Using Their Tweets. In *Proceedings of the International Symposium on Ambient Intelligence*; Springer: Berlin, Germany, 2020; pp. 224–235.
70. Rivas, A.; González-Briones, A.; Cea-Morán, J.J.; Prat-Pérez, A.; Corchado, J.M. My-Trac: System for Recommendation of Points of Interest on the Basis of Twitter Profiles. *Electronics* **2021**, *10*, 1263. [CrossRef]
71. Manca, M.; Boratto, L.; Morell Roman, V.; Martori i Gallissà, O.; Kaltenbrunner, A. Using social media to characterize urban mobility patterns: State-of-the-art survey and case-study. *Online Soc. Netw. Media* **2017**, *1*, 56–69. [CrossRef]
72. Balduini, M.; Brambilla, M.; Della Valle, E.; Marazzi, C.; Arabghalizi, T.; Rahdari, B.; Vescovi, M. Models and Practices in Urban Data Science at Scale. *Big Data Res.* **2019**, *17*, 66–84. [CrossRef]
73. Javadian Sabet, A. Social Media Posts Popularity Prediction during Long-Running Live Events. A Case Study on Fashion Week. Master's Thesis, Politecnico di Milano, Milan, Italy, 2019.
74. Brambilla, M.; Javadian Sabet, A.; Hosseini, M. The role of social media in long-running live events: The case of the Big Four fashion weeks dataset. *Data Brief* **2021**, *35*, 106840. [CrossRef]
75. Javadian Sabet, A.; Brambilla, M.; Hosseini, M. A multi-perspective approach for analyzing long-running live events on social media: A case study on the "Big Four" international fashion weeks. *Online Soc. Netw. Media* **2021**, *24*, 100140. [CrossRef]
76. Brambilla, M.; Javadian, A.; Sulistiawati, A.E. Conversation Graphs in Online Social Media. In Proceedings of the Web Engineering, ICWE 2021, Biarritz, France, 18–21 May 2021; Springer International Publishing: Cham, Switzerland, 2021; pp. 97–112. [CrossRef]
77. Brambilla, M.; Javadian Sabet, A.; Kharmale, K.; Sulistiawati, A.E. Graph-Based Conversation Analysis in Social Media. *Big Data Cogn. Comput.* **2022**, *6*, 113. [CrossRef]
78. Scotti, V.; Tedesco, R.; Sbattella, L. A Modular Data-Driven Architecture for Empathetic Conversational Agents. In Proceedings of the 2021 IEEE International Conference on Big Data and Smart Computing (BigComp), Jeju Island, Korea, 17–20 January 2021; pp. 365–368. [CrossRef]

big data and cognitive computing

Article

GTDOnto: An Ontology for Organizing and Modeling Knowledge about Global Terrorism

Reem Qadan Al-Fayez [1,*], Marwan Al-Tawil [1], Bilal Abu-Salih [2] and Zaid Eyadat [3]

[1] Computer Information Systems Department, King Abdullah II School of Information Technology, The University of Jordan, Amman 11942, Jordan
[2] Computer Science Department, King Abdullah II School of Information Technology, The University of Jordan, Amman 11942, Jordan
[3] Prince Al Hussein bin Abdullah II School of International Studies, The University of Jordan, Amman 11942, Jordan
* Correspondence: r.alfayez@ju.edu.jo

Abstract: In recent years and with the advancement of semantic technologies, shared and published online data have become necessary to improve research and development in all fields. While many datasets are publicly available in social and economic domains, most lack standardization. Unlike the medical field, where terms and concepts are well defined using controlled vocabulary and ontologies, social datasets are not. Experts such as the National Consortium for the Study of Terrorism and Responses to Terrorism (START) collect data on global incidents and publish them in the Global Terrorism Database (GTD). Thus, the data are deficient in the technical modeling of its metadata. In this paper, we proposed GTD ontology (GTDOnto) to organize and model knowledge about global incidents, targets, perpetrators, weapons, and other related information. Based on the NeOn methodology, the goal is to build on the effort of START and present controlled vocabularies in a machine-readable format that is interoperable and can be reused to describe potential incidents in the future. The GTDOnto was implemented with the Web Ontology Language (OWL) using the Protégé editor and evaluated by answering competency questions, domain experts' opinions, and running examples of GTDOnto for representing actual incidents. The GTDOnto can further be used to leverage the publishing of GTD as a knowledge graph that visualizes related incidents and build further applications to enrich its content.

Keywords: ontology; semantic web; social data; terrorism; OWL/RDF; knowledge graphs

Citation: Al-Fayez, R.Q.; Al-Tawil, M.; Abu-Salih, B.; Eyadat, Z. GTDOnto: An Ontology for Organizing and Modeling Knowledge about Global Terrorism. *Big Data Cogn. Comput.* **2023**, *7*, 24. https://doi.org/10.3390/bdcc7010024

Academic Editors: Konstantinos Kotis and Dimitris Spiliotopoulos

Received: 19 December 2022
Revised: 18 January 2023
Accepted: 19 January 2023
Published: 28 January 2023

1. Introduction

Recent studies have emphasized the importance of publishing open data despite the challenges faced [1,2]. Having datasets available and stored in data portals or repositories online offers more value for the data, especially when such datasets are available for researchers, businesses, and government to utilize [3]. Researchers and practitioners in different institutions have endorsed publishing datasets as open data [4]. The integration of several open datasets helps in making better decisions in general. Semantic web technologies allow the publishing and sharing of data in machine-readable formats that ease data integration and enable knowledge sharing and analytics capabilities [5–8]. The best example is the recent COVID-19 pandemic, in which sharing open datasets about the viruses and experimental datasets from previous literature enabled scientists to develop vaccines to save humanity in record time [9].

Finding trustworthy datasets is challenging, especially with the abundant datasets available and published under licensing conditions, in different formats, and with varying metadata standards [10]. Technical challenges for publishing open datasets, such as data replication and lack of standards for describing metadata, are discussed in the literature [11]. Furthermore, the semantic web revolutionized the publishing of information on the web since semantic technologies, such as ontologies and the RDF data model, replaced other common and widely used data models, such as HTML, XML, JSON, spreadsheet, or text files [12]. Such techniques solve data ambiguity, interoperability, and integration issues when published online.

The scarcity of publicly available semantic datasets published in social science is the motive behind this study. Literature about semantic web portals in more specific fields, such as world terrorism, is limited. To our knowledge, a few studies have worked in this domain. Some terrorist incidents are chains of operations reported by a small group of people in different places. The semantic modeling of such incidents will facilitate checking terror ties between operations after representing terrorist incidents. In [13], researchers realized the potential utilizing ontologies in analyzing terrorist network. When investigating terrorism, valuable information and external information about incidents, people, and targets are immensely needed. It is hard to label any violent attack as a terrorist incident, given the saying, "Terrorism is in the eyes of the beholder". The semantic modeling of social data, including terrorism information can serve as a valuable tool for terrorism investigators. It might not succeed in identifying links between operations happening in real time. Still, it can gather and organize information and help investigators to better explore and link information about a specific situation. For instance, disambiguating attacker names or places mentioned in terrorism incidents over media and social media can be performed via the semantic web. Referring to characters or places using URIs will enable better exploration of related information without hard linking articles and news. Hence, the semantic web will facilitate the intuitive process of exploring information and seeing patterns.

In the field of terrorism data organization, non-computer science specialists built an ontology for terrorism analysis and published it in an operational semantic web portal, Profiles in Terror (PiT) [14]. The researchers confessed that developing an ontology covering all aspects of terrorist activities is time-consuming. The semantic web portal link provided in the project (profilesinterror.mindswap.org) was not available at the time of writing this study. Another research found in the literature developed an ontology to organize the data collected from news articles about terrorism [15].

A team of multidisciplinary researchers at the University of Maryland worked on developing the criteria and attributes of each potential terrorist incident. With more than 50 years of experience, the National Consortium for the Study of Terrorism and Responses to Terrorism (START) at the University of Maryland maintained a project named the Global Terrorism Database (GTD) [16]. The team at START has developed the best practices for collecting information about terrorist incidents from 1970 onwards, containing almost 200,000 records with more than 130 variables describing incidents around the world [17].

In this work, we build on the efforts of GTD and propose the GTD ontology (GTDOnto) to define the concepts, vocabularies, and relations to describe any terrorist incident in the GTD using the OWL format. Based on the NeOn methodology guidelines, we developed this ontology from scratch to define vocabulary and relations used for representing the incidents of the GTD dataset in a machine-readable and interoperable format. We evaluated the applicability of the proposed ontology by describing incidents of the GTD using GTDOnto, in addition to domain expert feedback and competency questions answering. The remainder of this paper starts with the background knowledge section, followed by the methodology section. Next, we discuss the results and evaluation of the proposed ontology. Finally, the paper concludes with a summary and future work plans for using the proposed ontology.

2. Background

2.1. Ontologies

Ontologies have been central in developing the semantic web [18]. Gruber provided a famous definition of ontology as explicit formal specifications of the terms in the domain and relations among them [19]. At its most basic, an ontology consists of classes, instances, and properties [7].

Classes: define the main concepts in the domain. An example of a class in the musical instrument domain is *Guitar*. Characteristics of classes apply to their instances or individuals. For example, the fact that *Guitar* has nick and strings can be used with instances (i.e., individuals) of *Guitar*, such as *Vietnamese Guitar*. Every ontology has a class hierarchy consisting of all classes linked via the subsumption relationship *rdfs:subClassOf*. The set of concepts in the class hierarchy can be divided into the following three types: (i) *root entity* (the superclass for all entities in the class hierarchy); (ii) *Category entities* (set of all inner entities other than the root entity, that have at least one subclass; and (iii) *Leaf entities* (set of entities that have no subclasses). Every subclass in the hierarchy inherits the characteristics of its superclass.

Instances: (also called individuals) can be concrete objects, such as people, animals, and musical instruments, or abstract individuals, such as numbers and words (strings). Every instance belongs to at least one class (i.e., one instance can belong to more than one class). An instance inherits the attributes of its class and has specific values that differentiate them from other individuals in the class.

Relationships: (also called properties) in an ontology are used to define the characteristics of the classes. Relationships have labels, and they represent links between classes and instances. For example, a necessary type of relationship between classes is the subsumption relationship *rdfs:subClassOf*, which is used to identify the subclass/ superclass relationships in the class hierarchy (also called the subsumption class hierarchy). While the subsumption relationship is typical in different ontologies, other relationship types called object properties and data properties relationships are domain-specific and used in ontology.

2.2. Ontology Development Methodologies

In the literature, various methodologies for developing well-founded domain ontology are summarized [20]. The Uschold and King Methodology is one of the first ontology development approaches proposed [21]. It includes four stages: the first stage defines the purpose of the ontology (i.e., why the ontology is being developed and its intended uses). The second stage focuses on building the ontology and consists of the following phases: capture the ontology by identifying key concepts and relationships, code the ontology in a formal language, and integrate existing ontologies. The third stage focuses on evaluating the ontology, and the final step is documenting the ontology.

The Human-Centered Ontology Engineering Methodology (HCOME) ontology development methodology has been proposed as an approach that considers the active participation of knowledge in the ontology life cycle [22]. The HCMOE has three phases for ontology development: specification, conceptualization, and exploitation. In the specification phase, knowledge workers collaborate in defining the scope and aim of the ontology and producing specification documents. The second phase acquires knowledge from existing ontologies to develop and maintain the ontology. The final phase focuses on utilizing and browsing the ontology within an application and evaluating the ontology.

The work of [23] proposed an iterative approach to a simple knowledge-engineering ontology development methodology. The approach consists of three steps that start with a rough ontology, then revise and refine the ontology and finally discuss the modeling decisions, including the pros and cons of these decisions. This iterative approach continues during the lifecycle of the ontology that starts with defining the domain and scope of the ontology. The domain represents the main concepts used in the ontology and uses competency questions to determine the scope of the ontology. These questions test whether the ontology has enough information to answer them. The second step is about reusing existing ontologies. The third step focuses on writing essential concepts. Then the fourth step uses top-down, bottom-up, or a combination of both to define the class hierarchy. In step five, classes' properties are defined, and in step six, slots (i.e., objects' properties) between classes are defined. Finally, in step seven, instances (i.e., individuals) of classes in the class hierarchy are created. Kanga methodology engages domain experts in ontology development [24]. This methodology combines two aspects: the conceptual aspect written by domain experts. The logical aspect is performed by converting the conceptual knowledge into a machine-readable format, such as OWL. The methodology has five phases: (i) ontology requirements, where scope and purpose are identified; (ii) source knowledge capture, where knowledge sources and core concepts and relationships are identified; (iii) populating knowledge glossary, which covers the glossary of key concepts and relationships; (iv) formal structuring where a controlled natural language (OWL, RDF) is used to define concepts, relationships and axioms; and finally, (v) the evaluation and verification phase of the ontology using different techniques.

The NeOn methodology for ontology engineering suggests a variety of pathways for developing ontologies [25]. The methodology follows a Waterfall Ontology Network Life Cycle Model. This model describes four main phases for ontology development: (i) the initiation phase, which focuses on ontology requirements, (ii) the design phase, where the main concepts in the ontology are defined (i.e., ontology conceptualization), (iii) the implementation phase, which focuses on coding the logical ontology, and (iv) the maintenance phase, which concerns the documentation and validation of the developed ontology.

This study aims to develop a new ontology for organizing the description of global incidents based on the domain expert knowledge collected from the GTD project run by START. Hence, among the previously described ontology development methodologies, we selected the NeOn methodology since it provides scenarios for developing a new domain ontology and enables the development of an ontology from scratch using clear guidelines.

3. GTDOnto Development Methodology

This research adopts the NeOn methodology in developing the GTDOnto [26]. The NeOn methodology was followed in this research for several reasons: (1) It has been used in the literature to build ontologies in different areas by different people from various backgrounds [27–29]. (2) The NeOn methodology proposes several scenarios for developing an ontology, including a scenario for developing an ontology from scratch [30]. (3) Since this research aims to define controlled vocabulary for describing global terrorism-related incidents using RDF, the NeOn methodology enables having a glossary of terms that will be beneficial for building this vocabulary. (4) Compared to other methodologies for developing ontologies, NeOn is one of the most recent methodologies published in the literature and captures several older techniques presented in previous suggested work.

The proposed ontology in this work was developed by following the guidelines of Scenario 1: From Specification to Implementation. This scenario outlines the steps for implementing an ontology from scratch. Since there is little research about organizing information about terrorism, this scenario fits to develop GTDOnto. The proposed ontology will provide a standardized model to describe incidents from the GTD using entities and relations connecting these entities based on the GTD codebook.

The National Consortium START at the University of Maryland defined the variables used to describe incidents and published several research papers on their data collection methodology. Their work is documented in the GTD codebook and is used as the primary source for building the proposed GTDOnto [31]. Therefore, before starting with the NeOn methodology for developing GTDOnto, the acquisition process is explained in the next section.

3.1. Knowledge Acquisition

The current Global Terrorism Database (GTD) maintained by START is the product of several phases of data collection efforts. The data collected are published under a EULA agreement providing a conditional agreement to access and use the GTD. In addition, START offers an interface for browsing its content through its website (https://www.start.umd.edu/gtd/, accessed on 16 June 2021). As explained in their codebook, the data were collected based on media articles, electronic news archives, existing data sets, and other sources, such as books, journals, and legal documents. Several parties performed the data-collection process over different periods, all documented in their codebook [31].

The efforts performed by START are immense. The START institute explained all the legacy issues in the data collected regarding the information available about which incidents based on the data-collection date. During the data collection process, they committed to coding some variables and were transparent in explaining the coding decisions wherever possible. Furthermore, their data-collection methodology added the inclusiveness criteria for each incident. Since the definitions of terrorism vary and START targets the public mass, they describe the inclusiveness criteria for considering an incident as an act of terrorism.

Additionally, START introduced the doubted variable to document if an incident was arguably doubted to be a terrorist incident. Based on the trustworthiness of the GTD data-collection process, the work of START is considered the source for developing GTDOnto in its first version. The attributes of a terrorism incident in the GTD are translated into classes and properties describing different entities related to an incident.

The downloadable version of GTD is available in a spreadsheet format. It consists of (191465) rows representing terrorist incidents and (135) columns to describe each incident. The codebook explains each attribute defined in the columns, its use, coding (if any), and other issues in the data collected for this attribute. The attributes explain details about each incident in several categories: (1) GTD ID, incident date, (2) incident location, (3) incident information, (4) attack information, (5) weapon information, (6) target/victim information, (7) perpetrator information, perpetrator statistics, claims of responsibility, (8) casualties and consequences information, and (9) additional information, and source information.

3.2. Specification Phase

For writing the ontology specification, NeOn provides precise guidelines to build the Ontology Requirement Specification Document (ORSD) [32]. The ORSD states why the ontology is being developed. Table 1 illustrates the process of developing GTDOno using the ORSD template.

Based on one of the NeOn methodology scenarios for developing an ontology, *Scenario 1: from specification to implementation*, it is recommended to use competency questions (CQs) to create the ontology requirements [30]. Therefore, in this first draft of GTDOnto ontology, we identified the need to answer multiple competency questions about several categories related to incidents in the functional requirements of the ORSD table. In addition, the ORSD identifies the key terms from the GTD codebook associated with a terrorist incident. A sample of these terms is summarized in Table 2 and will be further detailed to build the GTDOnto.

Table 1. GTDOnto ontology requirements specification document (GTDOnto ORSD).

Purpose
GTDOnto ontology stands for Global Terrorism Database Ontology. The GTDOnto ontology represents the knowledge necessary to describe an incident considered an act of terrorism as in the GTD.
Scope
GTDOnto ontology identifies entities representing attackers, targets or victims, perpetrators, weapons, and detailed information about the causalities and consequences. The ontology provides many attributes to define values for the following categories: 1. GTD ID, incident date, 2. incident location, 3. incident information, 4. attack information, 6. target/victim information, 7. perpetrator information, perpetrator statistics, claims of responsibility, 5. weapon information, 8. casualty information, consequences, kidnapping/hostage taking information, 9. additional information, and source information.
Implementation Language
GTDOnto ontology is implemented in OWL/RDF using Protégé. The implementation process is performed manually to define all top-level concepts of the ontology, and further automation is performed to build lower-level concepts and properties.

Table 1. *Cont.*

Intended End-Users
GTDOnto ontology is vital to (i) end users who may browse the dataset, (ii) curators interested in integrating datasets, and (iii) researchers who may use the data for network analysis and further analytics studies.
Intended Uses
GTDOnto ontology models the controlled vocabulary describing GTD incidents, such as attack types, weapon types, and target types. The ontology can be used to publish the incidents from the GTD to form a knowledge graph. In addition, the GTDOnto helps visualize the relations between potential attacks or attackers. Publishing the GTD using this ontology will result in a knowledge graph of related incidents that can be further explored for advanced data analysis. Furthermore, the possible knowledge graph representing incidents using GTDOnto can be enriched with content from other datasets and social media content.
Ontology Requirements
Non-Functional Requirements
Not applicable Functional Requirements
It can be set with groups of Competency Questions (CQG) that cover all the concepts in the incident. **CQG1: General information about an incident/incidents:** CQ1.1: How many incidents occurred in the year 2005? CQ1.2: What is the detailed location of the incident with ID ”200109110004”? CQ1.3: Is there any doubt that the incident with ID ”200109110004” is a terrorist incident or not? CQ1.4: If the incident is part of multiple events, what are the related incidents? … … **CQG2: Attack details related to an incident/incidents:** CQ2.1: What are the possible types of attacks associated with global incidents? CQ2.2: Was the terrorist attack successful, and was it a suicidal attack? CQ2.3: What are the attack types recorded in the incident with an ID ”200109110004”? CQ2.4: What are the incidents that recorded ”BombingExolosive” attacks? … … **CQG3: Weapons used in an incident/incident:** CQ3.1: What are the possible weapon types recorded with global incidents? CQ3.2: What are the possible weapons subtypes of ”explosives” weapons type? CQ3.3: What types and subtypes of weapons were used in the incident with an ID ”200109110004”? CQ3.4: What incidents used a weapon of type "explosives"? **… …** **CQG4: Targets and victims details related to an incident/incidents:** CQ4.1: What are the possible targets for global incidents? CQ4.2: What are the possible categories for a military-targeted attack? CQ4.3: What are the nationalities of all targets/victims of a terrorist incident? CQ4.4: Who is the specific target/victim of a terrorist incident? **… …** **CQG5: Perpetrators details for an incident/incidents:** CQ5.1: Does the terrorist incident claim responsibility by a group? If yes, what is the group name that carried out the incident? CQ5.2: How many individuals are reported to participate in the incident, and how many are taken into custody? CQ5.3: What methods are used to announce the claim of responsibility for a terrorist incident? **… …** **CQG6: Casualties and consequences of an incident:** CQ6.1: How many confirmed fatalities and injuries were reported in a terrorist incident? CQ6.2: How many perpetrators were killed and injured in a terrorist incident? CQ6.3: Is there any property damage reported in the incident? If yes, what is the damage to property that occurred in the incident? CQ6.4: Were there any hostages in the incident? What is the outcome of reported hostage/kidnapping incidents? What is the total number of hostages? **… …**

Table 2. Sample of terms found in GTD.

Incident	Attack	Target
Perpetrator	Weapon	Casualties
Ransom	Hijacking	Country/Nationality

3.3. Conceptualization Phase

Following the GTD codebook and the ORSD for the proposed GTDOnto, many concepts describe an incident illustrated in a conceptual model shown in Figure 1.

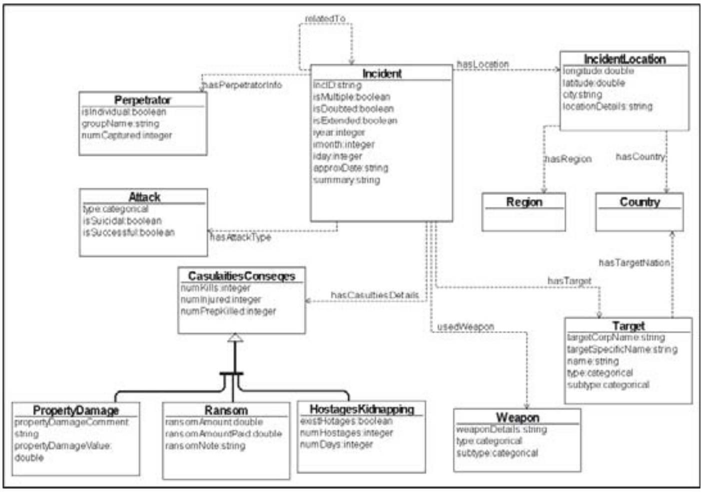


Figure 1. Conceptual model of GTDOnto ontology.

The conceptual model is structured in a class view, with subclasses showing further details related to each concept. The conceptual model defines the core concepts and examples of properties describing each concept. For example, the casualties and consequences class have further details, such as ransom demand, hijacking or kidnapping victims, or property damaged during an attack. In GTDOnto, the goal is to describe each class with properties and relations with other classes. Table 3 lists examples of the data properties needed to describe each class in a data dictionary.

Table 3. Data attributes examples.

Concept	Data Attributes	Description
Incident	id	A numeric variable follows a 12-digit Event ID system ex. "199307250001".
	day	A numeric variable records the day of the month on which the incident occurred.
	Month	A numeric variable records the month in which the incident occurred.
	year	A numeric variable records the year in which the incident occurred.
	approxdate	A text variable is used whenever the exact incident date is unknown or remains unclear. It records the approximate date of the incident.
	summary	A brief narrative summary of the incident, noting the "when, where, who, what, how, and why".

Table 3. *Cont.*

Concept	Data Attributes	Description
Attack	isSuccessful	A categorical type records whether the incident was successful or not. The definition of a successful attack depends on the type of attack. The key question is whether or not the attack type took place.
	isSuicidal	A categorical type records whether the incident was a suicide attack or not.
	type	A categorical type records the general attack method, consisting of nine categories.
Casualties and Consequences	numKills	A numeric variable stores the number of confirmed fatalities for the incident, including all victims and attackers who died as a direct result of the incident.
	numInjured	A numeric variable records the number of confirmed non-fatal injuries to both perpetrators and victims.
	numPerpKilled	A numeric variable limited to only perpetrator fatalities.
Hijacking	existHostage	A categorical variable records whether the victims were taken hostage or kidnapped during the incident.
	numHostages	A numeric variable records the total number of hostages or kidnapping victims.
	numHours	A numeric variable records the duration of the incident if the incident lasted for less than 24 h.
	numDays	A numeric variable records the duration of the incident in days if the kidnapping/hostage incident lasts more than 24 h.
	numReleased	A numeric variable records the number of hostages who survived the incident

After defining the data properties used to describe concepts in the GTDOnto, object properties are defined to describe the relations between different concepts. Table 4 represents a sample of possible relations between other concepts.

Table 4. GTDOnto object properties examples.

Object Property	Domain	Range
hasCountry	Incident	Country
hasAttackType	Incident	Attack
usedWeapon	Incident	Weapon
IsDoubted	Incident	DoubtStatus
hasAlternative	YesDoubted	AlternativeDesignation
hasHostKid	Incident	HostagesKidnappingStatus
hasHostKidOutcome	VictimsHostageKidnapped	HostKidnappingOutcome

3.4. Formalization Phase

After modeling the concepts, properties, and relations between concepts, the GTDOnto ontology is formalized. This process involves identifying subsumption relations and identifying domains and ranges for data and object properties. These properties are the semantic relations between pairs of classes to build relations between ontology instances in the future. The formalization phase identified classes, subsumption relations, object properties, and data properties. This version for the GTDOnto ontology is the first proposed version and is keen to further changes or addition to its properties after practice and usage of the ontology.

The subsumption relation builds the taxonomy of classes and subclasses in the GTDOnto ontology. For example, the *Weapon* class represents weapons used in the attacks, and it consists of 13 subclasses of weapons coded in the GTD codebook, such as *Biological*, *Chemical*, *Explosives*, *Firearms*, *Nuclear*, and others. Some of these weapon types have subtypes. For example, the *Chemical* weapon type can be *Explosive* or *Poisoning*. Meanwhile, the *Explosives* weapon type class has subtypes of weapons such as *Dynamite TNT*, *Grenade*, *Landmine*, and others.

The object properties build relations between classes. For example, to detail that some incidents in the GTD had hostages or kidnapping of victims, several features are recorded about this. In GTDOnto onology, the "*hasHostKid*" object property relates the *Incident* class with *HostagesKidnappingStatus* class that includes three subclasses to represent *NoVictimsHostageKidnapped*, *VictimsHostageKidnapped*, or *UnknownHostageKidnapped*. The GTD has cases where no information was recorded regarding hostages or the kidnapping of victims. Hence, the unknown status is required to confirm that information is missing in some incidents. If it was confirmed that victims were kidnapped or taken as hostages, further details are required to be recorded: *numHours*, *numDays*, *numHostKid*, *numReleased*, *KidhijCountry,* and others. Furthermore, the outcome of this attack is formalized by "*hasHostKidOutcome*" object property relating the Incident class with *HostKidnappingOutcome* class, which has seven possible outcomes represented as classes: *AttemptedRescue*, *HostagesEscaped*, *HostagesKilled*, *SuccessfulRescue*, and others.

To formalize the conceptual model and the relations for describing terrorism incidents, first-order-logic (FOL) was used before developing the GTDOnto. The FOL syntax defines knowledge about concepts in any domain as objects, relations, and functions close to natural human language. A sample of the FOL formulas and their representation statements are detailed below. These statements describe incidents and other concepts in the GTDOnto ontology:

- Any incident with victims taken as hostages or kidnapped has an attack-type of hijacking or hostages taken.

$$\begin{gathered}\forall x(\text{Incident}(\text{x}) \wedge \text{ hasHostKid}(\text{x}, \textit{ VictimsHostageKidnapped}) \\ \rightarrow \text{ hasAttackType}(\text{x}, \textit{ Hijacking}) \text{ hasAttackType}(\text{x}, \textit{ HostageTaking}))\end{gathered}$$

- A weapon of type chemical can be an explosive or poisoning weapon.

$$\exists y(\text{Weapon}(\text{y}) \ \wedge \ \text{Chemical}(\text{y}) \rightarrow \ \text{Explosive}(\text{y}) \vee \textit{Poisoning}(y))$$

- All incidents must have at least one weapon type recorded.

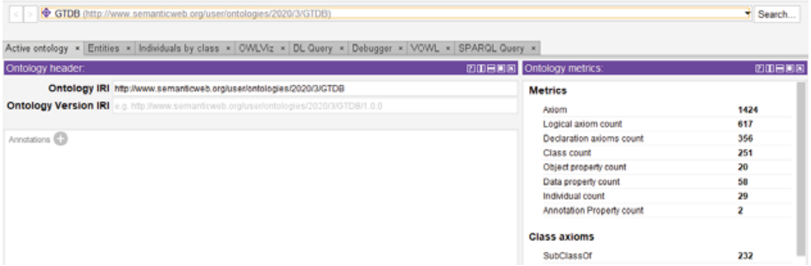
$$\forall \text{x}\exists \text{y}(\text{Incident }(\text{x}) \ \wedge \ \text{Weapon}(\text{y}) \rightarrow \ \text{hasWeaponType}(\text{x}, \text{ y}))$$

- For some incidents, more specific sub-weapon types can be recorded.

$$\begin{gathered}\exists \text{x}\exists \text{y}(\text{Incident }(\text{x}) \ \wedge \ \text{Weapon}(\text{y}) \wedge \text{Chemical}(\text{y}) \wedge (\text{Explosive}(\text{y}) \ \vee \textit{Poisoning}(y\ ) \\ \rightarrow \text{hasWeaponType}(\text{x}, \text{ y}) \wedge \text{ hasWeaponSubType}(\text{x}, \text{ y})\end{gathered}$$

3.5. Implementation Phase

The GTDOnto ontology is implemented in OWL/RDF using Protégé. The GTDOnto ontology contains 251 classes, 232 subsumption relations, 20 object properties, and 58 data properties, as shown in Figure 2. In addition, the ontology includes 29 individuals of different class types for evaluation purposes. The results and evaluation section explains the detailed description and analysis of this ontology's components.



GTDB (http://www.semanticweb.org/user/ontologies/2020/3/GTDB)

Ontology header:

Ontology IRI http://www.semanticweb.org/user/ontologies/2020/3/GTDB

Ontology Version IRI e.g. http://www.semanticweb.org/user/ontologies/2020/3/GTDB/1.0.0

Ontology metrics:

Metrics

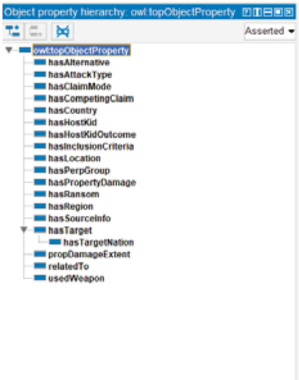
Axiom	1424
Logical axiom count	617
Declaration axioms count	356
Class count	251
Object property count	20
Data property count	58
Individual count	29
Annotation Property count	2

Class axioms

SubClassOf	232

Figure 2. GTDOnto ontology metrics in Protégé.

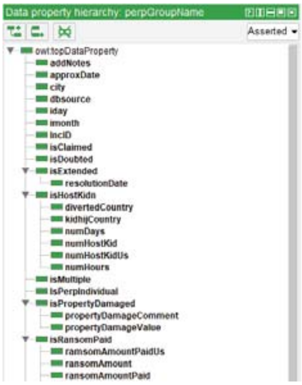
All the concepts depicted in the previous phases are built as classes in protégé for the GTDOnto. Each class was assigned its object properties and data properties as conceptualized. Figure 3 illustrates the object properties built in GTDOnto Ontology. Each object property has a domain and a range of different related classes. The data properties are used to describe literal values for classes. Figure 4 illustrates the GTDOnto implementation of these properties.



Object property hierarchy: owl:topObjectProperty

- owl:topObjectProperty
  - hasAlternative
  - hasAttackType
  - hasClaimMode
  - hasCompetingClaim
  - hasCountry
  - hasHostKid
  - hasHostKidOutcome
  - hasInclusionCriteria
  - hasLocation
  - hasPerpGroup
  - hasPropertyDamage
  - hasRansom
  - hasRegion
  - hasSourceInfo
  - hasTarget
    - hasTargetNation
  - propDamageExtent
  - relatedTo
  - usedWeapon

Figure 3. Object properties implementation.



Data property hierarchy: perpGroupName

- owl:topDataProperty
  - addNotes
  - approxDate
  - city
  - dbsource
  - iday
  - imonth
  - IncID
  - isClaimed
  - isDoubted
  - isExtended
    - resolutionDate
  - isHostKidn
    - divertedCountry
    - kidhijCountry
    - numDays
    - numHostKid
    - numHostKidUs
    - numHours
  - isMultiple
  - IsPerpIndividual
  - isPropertyDamaged
    - propertyDamageComment
    - propertyDamageValue
  - isRansomPaid
    - ramsomAmountPaidUs
    - ransomAmount
    - ransomAmountPaid

Figure 4. Data properties implementation in GTDOnto in GTDOnto.

4. The GTDOnto Ontology

The developed GTDOnto ontology is available upon request from the authors, as this is an ongoing effort to advance this work. However, due to the scarcity of ontologies in the terrorism domain and the lack of documentation for some, GTDOnto does not reuse any existing ontology. Figure 5 represents the result of all the object properties linked to the incident class being its domain and other classes being its range. For example, the *hasAttackType* object property is represented by the edge connecting the *Incident* with the *Attack*. The *relatedTo* object property is represented by the loop edge on the *Incident* class.

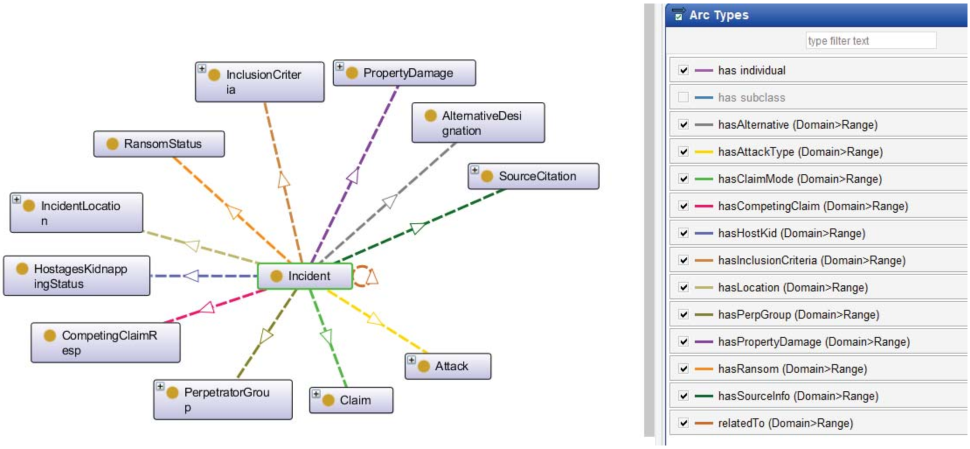
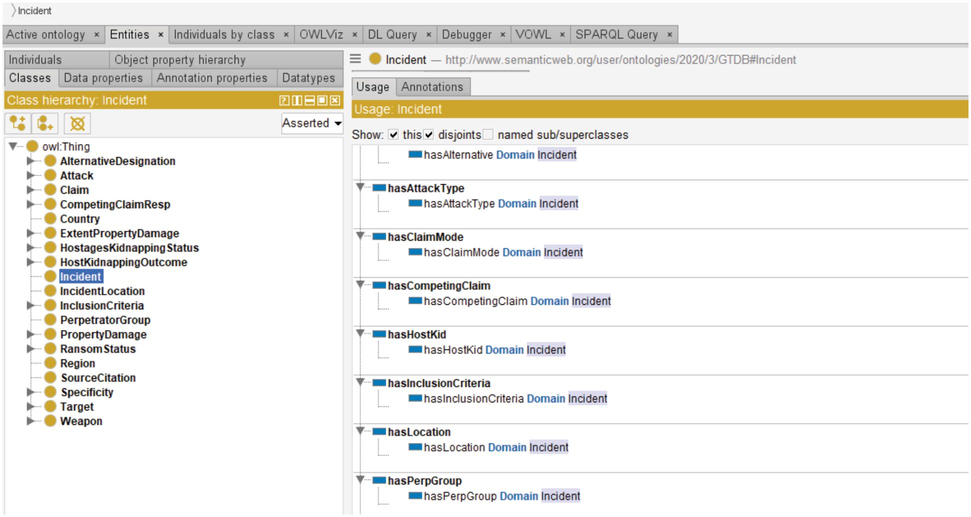

InclusionCriteria
PropertyDamage
AlternativeDesignation
RansomStatus
SourceCitation
IncidentLocation
HostagesKidnappingStatus
Incident
CompetingClaimResp
Attack
PerpetratorGroup
Claim
Arc Types
type filter text
☑ has individual
☐ has subclass
☑ hasAlternative (Domain>Range)
☑ hasAttackType (Domain>Range)
☑ hasClaimMode (Domain>Range)
☑ hasCompetingClaim (Domain>Range)
☑ hasHostKid (Domain>Range)
☑ hasInclusionCriteria (Domain>Range)
☑ hasLocation (Domain>Range)
☑ hasPerpGroup (Domain>Range)
☑ hasPropertyDamage (Domain>Range)
☑ hasRansom (Domain>Range)
☑ hasSourceInfo (Domain>Range)
☑ relatedTo (Domain>Range)

Figure 5. Incident class and its relations.

Figure 6 shows a snippet of the class hierarchy of the GTDOnto ontology implemented on Protégé. The incident class is highlighted in the hierarchy panel on the left side, and its related object properties are detailed on the right side of the figure. All the object properties connect the Incident class with other classes to record all the information about an incident.



〉Incident

Active ontology × | Entities × | Individuals by class × | OWLViz × | DL Query × | Debugger × | VOWL × | SPARQL Query ×

Individuals | Object property hierarchy

Classes | Data properties | Annotation properties | Datatypes

Class hierarchy: Incident

Asserted

- owl:Thing
  - AlternativeDesignation
  - Attack
  - Claim
  - CompetingClaimResp
  - Country
  - ExtentPropertyDamage
  - HostagesKidnappingStatus
  - HostKidnappingOutcome
  - Incident
  - IncidentLocation
  - InclusionCriteria
  - PerpetratorGroup
  - PropertyDamage
  - RansomStatus
  - Region
  - SourceCitation
  - Specificity
  - Target
  - Weapon

Incident — http://www.semanticweb.org/user/ontologies/2020/3/GTDB#Incident

Usage | Annotations

Usage: Incident

Show: ☑ this ☑ disjoints ☐ named sub/superclasses

hasAlternative Domain Incident

hasAttackType
hasAttackType Domain Incident

hasClaimMode
hasClaimMode Domain Incident

hasCompetingClaim
hasCompetingClaim Domain Incident

hasHostKid
hasHostKid Domain Incident

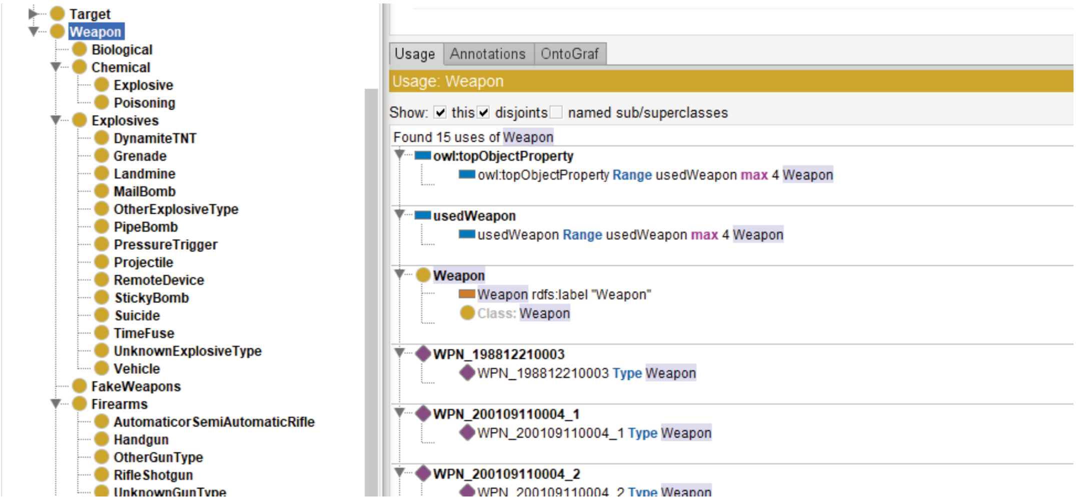
hasInclusionCriteria
hasInclusionCriteria Domain Incident

hasLocation
hasLocation Domain Incident

hasPerpGroup
hasPerpGroup Domain Incident

Figure 6. GTDOnto classes and the details of the Incident class.

The GTDOnto ontology models any knowledge about global incidents stored in the GTD. For example, Figure 7 details the Weapon class, including all its types and subtypes implemented as subsumption relation with the Weapon class—additionally, an object property relating Weapon with an incident that used a specific weapon type.



- Target
- Weapon
  - Biological
  - Chemical
    - Explosive
    - Poisoning
  - Explosives
    - DynamiteTNT
    - Grenade
    - Landmine
    - MailBomb
    - OtherExplosiveType
    - PipeBomb
    - PressureTrigger
    - Projectile
    - RemoteDevice
    - StickyBomb
    - Suicide
    - TimeFuse
    - UnknownExplosiveType
    - Vehicle
  - FakeWeapons
  - Firearms
    - AutomaticorSemiAutomaticRifle
    - Handgun
    - OtherGunType
    - RifleShotgun
    - UnknownGunType

Usage | Annotations | OntoGraf

Usage: Weapon

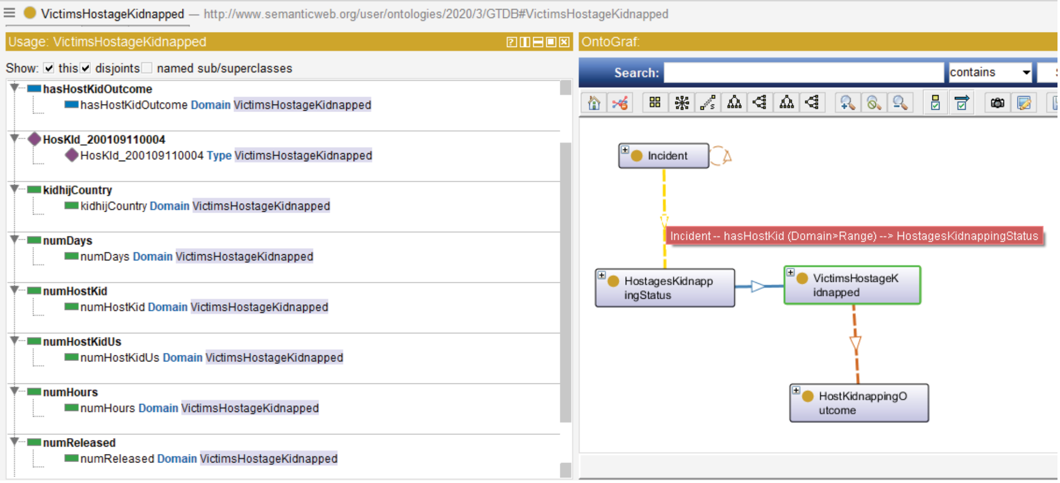
Show: ☑ this ☑ disjoints □ named sub/superclasses

Found 15 uses of Weapon

- owl:topObjectProperty
  - owl:topObjectProperty Range usedWeapon max 4 Weapon
- usedWeapon
  - usedWeapon Range usedWeapon max 4 Weapon
- Weapon
  - Weapon rdfs:label "Weapon"
  - Class: Weapon
- WPN_198812210003
  - WPN_198812210003 Type Weapon
- WPN_200109110004_1
  - WPN_200109110004_1 Type Weapon
- WPN_200109110004_2
  - WPN_200109110004_2 Type Weapon

Figure 7. GTDOnto Weapon class details.

Furthermore, many details about an incident are recorded using data properties. For example, more information on the incident should be reported if an incident involved the hijacking or kidnapping of victims. Figure 8 shows an example of the many data properties for the class *VictimsHostageKidnapped*—these data properties record information about the incident victims who were taken as hostages or kidnapped.



VictimsHostageKidnapped — http://www.semanticweb.org/user/ontologies/2020/3/GTDB#VictimsHostageKidnapped

Usage: VictimsHostageKidnapped

Show: ☑ this ☑ disjoints □ named sub/superclasses

- hasHostKidOutcome
  - hasHostKidOutcome Domain VictimsHostageKidnapped
- HosKId_200109110004
  - HosKId_200109110004 Type VictimsHostageKidnapped
- kidhijCountry
  - kidhijCountry Domain VictimsHostageKidnapped
- numDays
  - numDays Domain VictimsHostageKidnapped
- numHostKid
  - numHostKid Domain VictimsHostageKidnapped
- numHostKidUs
  - numHostKidUs Domain VictimsHostageKidnapped
- numHours
  - numHours Domain VictimsHostageKidnapped
- numReleased
  - numReleased Domain VictimsHostageKidnapped

OntoGraf:

Search: contains

Figure 8. GTDOnto hostages or kidnapping details.

In addition to building classes and properties, cardinality restrictions are added to some properties to denote the number of maximum relations a class can have. For example, an incident can have multiple attack types in the same terrorist incident. Such restriction is represented in the GTDOnto object property "hasAttackType" which limits the maximum cardinality to three types of attacks for an incident. The sample OWL/RDF code details this restriction on the object property hasAttackType.

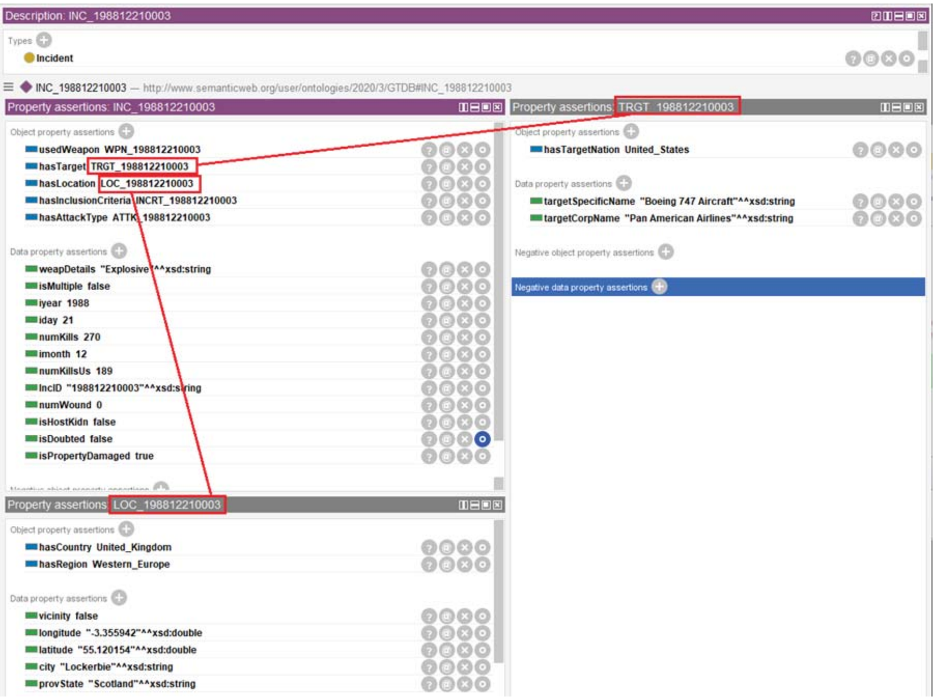
```
<!- http://www.semanticweb.org/user/ontologies/2020/3/GTDB#hasAttackType ->
   <owl:ObjectProperty
rdf:about="http://www.semanticweb.org/user/ontologies/2020/3/GTDB#hasAttackType">
       <rdfs:subPropertyOf
rdf:resource="http://www.w3.org/2002/07/owl#topObjectProperty"/>
       <rdfs:domain
rdf:resource="http://www.semanticweb.org/user/ontologies/2020/3/GTDB#Incident"/>
       <rdfs:range>
<owl:Restriction> <owl:onProperty
rdf:resource="http:
//www.semanticweb.org/user/ontologies/2020/3/GTDB#hasAttackType"/>
<owl:maxQualifiedCardinality
rdf:datatype="http:
//www.w3.org/2001/XMLSchema#nonNegativeInteger">3</owl:maxQualifiedCardinality>
<owl:onClass
rdf:resource="http://www.semanticweb.org/user/ontologies/2020/3/GTDB#Attack"/>
           </owl:Restriction>
       </rdfs:range>
   </owl:ObjectProperty>
```

5. Evaluation and Discussion

Ontology evaluation aims to assess the developed ontology's quality and correctness, where the evaluation change according to the ontology development method [33]. The work of [34] summarized ontology evaluation approaches into four approaches: (1) gold standard approach that compares the developed ontology to an existing (gold standard) and finds similarities/differences; (2) data-driven approach evaluates against a given corpus (set of terms or documents); (3) the application-based approach evaluates the ontology in performing a specific task; and (4) criteria-based approaches such as human assessment, which asks humans (usually domain experts) to evaluate the ontology. Further evaluation is conducted on the schema design of the ontology based on the OntoQA evaluation tool [35] and presented in this section. In this work, the gold standard approach does not apply since no existing ontology exists to match. On the other hand, the data-driven approach can be used in future work to measure the usefulness of the GTDOnto with several applications developed to use it. Hence, the latter two approaches were used to evaluate the GTDOnto.

5.1. Running Examples

Based on the task-based approach, two instances of the incident class are created to evaluate the GTDOnto and assess if the ontology can be used to describe different incidents from the GTD and have all the terms and properties of that incident represented. Therefore, a couple of individuals were instantiated of type Incident class. Two incidents were selected from the GTD, and the information described in the database was taken as is and represented using our proposed ontology. Other types of individuals were created to represent all information about these incidents, as shown in Figures 9 and 10. The famous 1988 Lockerbie aircraft bombing incident is illustrated in Figure 9, and the September 11 incident is represented in Figure 10. The primary and first task for building GTDOnto is to represent the incidents in a graph structure that can be queried efficiently. The GTDOnto exploration is further investigated by answering several competency questions using SPARQL queries.



Description: INC_198812210003

Types
Incident

INC_198812210003 — http://www.semanticweb.org/user/ontologies/2020/3/GTDB#INC_198812210003

Property assertions: INC_198812210003

Object property assertions
usedWeapon WPN_198812210003
hasTarget TRGT_198812210003
hasLocation LOC_198812210003
hasInclusionCriteria INCRT_198812210003
hasAttackType ATTK_198812210003

Data property assertions
weapDetails "Explosive"^^xsd:string
isMultiple false
iyear 1988
iday 21
numKills 270
imonth 12
numKillsUs 189
IncID "198812210003"^^xsd:string
numWound 0
isHostKidn false
isDoubted false
isPropertyDamaged true

Property assertions: TRGT_198812210003

Object property assertions
hasTargetNation United_States

Data property assertions
targetSpecificName "Boeing 747 Aircraft"^^xsd:string
targetCorpName "Pan American Airlines"^^xsd:string

Negative object property assertions

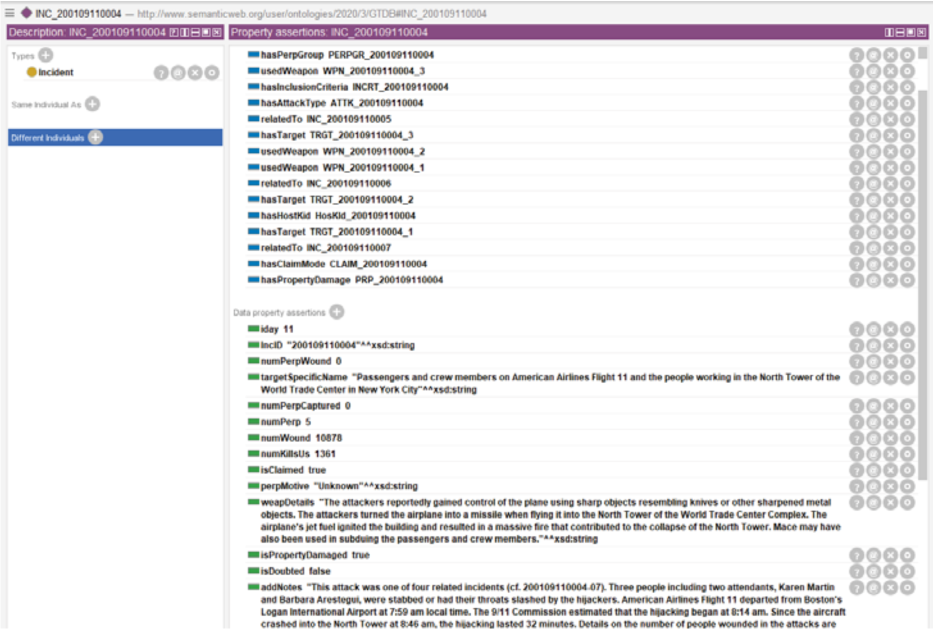
Negative data property assertions

Property assertions: LOC_198812210003

Object property assertions
hasCountry United_Kingdom
hasRegion Western_Europe

Data property assertions
vicinity false
longitude "-3.355942"^^xsd:double
latitude "55.120154"^^xsd:double
city "Lockerbie"^^xsd:string
provState "Scotland"^^xsd:string

Figure 9. GTD incident (ID: 198812210003) representation in the GTDOnto (1988 Lockerbie incident).



INC_200109110004 — http://www.semanticweb.org/user/ontologies/2020/3/GTDB#INC_200109110004

Description: INC_200109110004

Types
Incident

Same Individual As

Different Individuals

Property assertions: INC_200109110004

hasPerpGroup PERPGR_200109110004
usedWeapon WPN_200109110004_3
hasInclusionCriteria INCRT_200109110004
hasAttackType ATTK_200109110004
relatedTo INC_200109110005
hasTarget TRGT_200109110004_3
usedWeapon WPN_200109110004_2
usedWeapon WPN_200109110004_1
relatedTo INC_200109110006
hasTarget TRGT_200109110004_2
hasHostKid HosKid_200109110004
hasTarget TRGT_200109110004_1
relatedTo INC_200109110007
hasClaimMode CLAIM_200109110004
hasPropertyDamage PRP_200109110004

Data property assertions
iday 11
IncID "200109110004"^^xsd:string
numPerpWound 0
targetSpecificName "Passengers and crew members on American Airlines Flight 11 and the people working in the North Tower of the World Trade Center in New York City"^^xsd:string
numPerpCaptured 0
numPerp 5
numWound 10878
numKillsUs 1361
isClaimed true
perpMotive "Unknown"^^xsd:string
weapDetails "The attackers reportedly gained control of the plane using sharp objects resembling knives or other sharpened metal objects. The attackers turned the airplane into a missile when flying it into the North Tower of the World Trade Center Complex. The airplane's jet fuel ignited the building and resulted in a massive fire that contributed to the collapse of the North Tower. Mace may have also been used in subduing the passengers and crew members."^^xsd:string
isPropertyDamaged true
isDoubted false
addNotes "This attack was one of four related incidents (cf. 200109110004-07). Three people including two attendants, Karen Martin and Barbara Arestegui, were stabbed or had their throats slashed by the hijackers. American Airlines Flight 11 departed from Boston's Logan International Airport at 7:59 am local time. The 9/11 Commission estimated that the hijacking began at 8:14 am. Since the aircraft crashed into the North Tower at 8:46 am, the hijacking lasted 32 minutes. Details on the number of people wounded in the attacks are

Figure 10. GTD Incident GTD (ID: 200109110004) representation in the GTDOnto (the September 11 incident).

The experiment for representing the above two incidents indicates that the GTDOnto can precisely represent all the properties describing incidents, whether they have detailed information or lack some information. Figure 9 details the data and object properties, with snippets of the other related individuals to the Lockerbie incident. This incident was used with 38 properties and relations—for example, the individuals of type location and target detail extra information about the incident. The GTDOnto representation for all incidents in this form will result in a knowledge graph of incidents and other class types. Figure 10 represents one of the September 11 incidents. It was used with 82 properties and relations. Full details about this incident were detailed in the GTD; hence, its representation in the GTDOnto is detailed. The object properties for this incident relate it to three different weapons used, three different targets, and three related incidents, as shown in the object properties.

5.2. Competency Questions Answering

The ontology covers the main concepts to describe an act of terrorism. Regarding the competency questions used to define the GTDOnto in the ORSD presented in Table 1, the ontology can answer all the competency questions asked in the functional requirements definition. Additionally, answering competency questions is part of the NeOn development methodology for the assessment of the GTDOnto ontology.

A sample of the competency questions from the groups specified in Table 1 is answered using SPARQL to evaluate the completeness criteria of the GTDOnto. Additionally, this evaluation assesses the correct representation of domains and ranges for the properties.

First, the Incident class, with all the associated data and object properties, can provide general answers about the dataset and much more detailed information about a specific incident. For example, the CQ1.1 (How many incidents occurred in the year 2005?) is resolved with the following SPARQL query:

```
PREFIX xsd:  <http://www.w3.org/2001/XMLSchema#>
PREFIX rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs:  <http://www.w3.org/2000/01/rdf-schema#>
PREFIX : <http://www.semanticweb.org/user/ontologies/2020/3/GTDB#>
select (count(?inc) as ?IncNum) where {
   ?inc :iyear "2001"^^xsd:integer.
   ?inc :hasLocation ?loc.
   ?loc :hasCountry ?country.
   ?country rdfs:label "United States"^^xsd:string.
}
```

With the Attack class modeled and the data and object properties associated with it, it is possible to answer several competency questions about that concept. The CQ2.1 (What are the possible types of attacks associated with global incidents?) is resolved with the following SPARQL query:

```
PREFIX xsd:  <http://www.w3.org/2001/XMLSchema#>
PREFIX rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs:  <http://www.w3.org/2000/01/rdf-schema#>
PREFIX : <http://www.semanticweb.org/user/ontologies/2020/3/GTDB#>
select ?attkTypes where {
   ?attkTypes rdfs:subClassOf :Attack.
}
```

The result of this query is all the types of attack associated with any act of terrorism as coded in the GTD codebook, e.g., armed assault, bombing explosion, assassination . . . , etc. Furthermore, the GTDOnto can answer CQ2.4 (What are the incidents that recorded "BombingExolosive" attacks?) to enlist all incidents of attack type Bombing explosion with the following SPARQL query:

```
PREFIX xsd:  <http://www.w3.org/2001/XMLSchema#>
PREFIX rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs:  <http://www.w3.org/2000/01/rdf-schema#>
PREFIX : <http://www.semanticweb.org/user/ontologies/2020/3/GTDB#>
select ?inc where {
   ?inc :hasAttackType ?attk.
   ?attk rdf:type :BombingExplosion.
}
```

Furthermore, with the Weapon class modeled and the data and object properties associated with it, it is possible to answer more detailed questions. For example, CQ3.2 (What are the possible weapons subtypes of "explosives" weapons type?) enquires about subtypes of explosive weapons. It can be resolved with the following SPARQL query.

```
PREFIX xsd:  <http://www.w3.org/2001/XMLSchema#>
PREFIX rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs:  <http://www.w3.org/2000/01/rdf-schema#>
PREFIX : <http://www.semanticweb.org/user/ontologies/2020/3/GTDB#>
select ?explosiveTypes where {
   ?explosiveTypes rdfs:subClassOf :Explosives.
}
```

It is also possible to answer CQ3.3 (What types and subtypes of weapons were used in the incident with an ID "200109110004"?) about a specific incident with the following SPARQL query.

```
PREFIX xsd:  <http://www.w3.org/2001/XMLSchema#>
PREFIX rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs:  <http://www.w3.org/2000/01/rdf-schema#>
PREFIX : <http://www.semanticweb.org/user/ontologies/2020/3/GTDB#>
select ?wpn ?subWpn where {
   ?inc :IncID "200109110004"^^xsd:string.
   ?inc :usedWeapon ?w.
   ?w rdf:type ?wpn.
   ?wpn rdfs:subClassOf :Weapon.
   Optional {
       ?w rdf:type ?subWpn.
       ?subWpn rdfs:subClassOf ?wpn.  }
}
```

The result of this SPARQL query returns the following weapon types (subtypes): Incendiary (Gasoline Alcohol), Melee (Knife or Other Sharp Object), and Vehicle for the incident illustrated in Figure 10.

With the Targets class modeled and the data and object properties associated with it, it is possible to answer several competency questions, such as CQ4.3 (What are the nationalities of all targets/victims of a terrorist incident?) for a specific incident. Such questions can be resolved with the following SPARQL query.

```
PREFIX xsd:  <http://www.w3.org/2001/XMLSchema#>
PREFIX rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs:  <http://www.w3.org/2000/01/rdf-schema#>
PREFIX : <http://www.semanticweb.org/user/ontologies/2020/3/GTDB#>
Select distinct ?nationality where {
   ?inc :hasTarget ?trgt.
   ?trgt :hasTargetNation ?country.
   ?country rdfs:label ?nationality.
}
```

It is possible to answer several conditional competency questions with information about perpetrators modeled in the Perpetrator class and the associated data and object properties. For example, to answer CQ5.1 (Does the terrorist incident claim responsibility by a group? If yes, what is the group name that carried out the incident?), the following SPARQL query is used.

```
PREFIX xsd:  <http://www.w3.org/2001/XMLSchema#>
PREFIX rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs:  <http://www.w3.org/2000/01/rdf-schema#>
PREFIX : <http://www.semanticweb.org/user/ontologies/2020/3/GTDB#>
select ?name ?claim where {
   ?inc :IncID "200109110004"^^xsd:string.
   ?inc :isClaimed ?claim.
   Optional {
       ?inc :hasPerpGroup ?group.
       ?group :perpGroupName ?name.  }
}
```

More descriptive information can be enquired about the casualties and consequences of the September 11 incident (Figure 10). For example, a simple question can be CQ6.1 (How many confirmed fatalities and injuries were reported in a terrorist incident?). A more detailed question is CQ6.4 (Were there any hostages in the incident? What is the outcome of reported hostage/kidnapping incidents? And what is the total number of hostages?). The CQ6.1 is resolved by the following SPARQL query to show the number of kills and wounded people for an incident.

```
PREFIX xsd:  <http://www.w3.org/2001/XMLSchema#>
PREFIX rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs:  <http://www.w3.org/2000/01/rdf-schema#>
PREFIX : <http://www.semanticweb.org/user/ontologies/2020/3/GTDB#>
select ?nKills ?nWounded where {
   ?inc :IncID "200109110004"^^xsd:string.
   ?inc :numKills ?nKills.
   ?inc :numWound ?nWounded.
}
```

While CQ6.4 considers the incident of hijacking a plane, the following SPARQL details the number of hostages, hours, and the outcome of this hijacking.

```
PREFIX xsd:  <http://www.w3.org/2001/XMLSchema#>
PREFIX rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs:  <http://www.w3.org/2000/01/rdf-schema#>
PREFIX : <http://www.semanticweb.org/user/ontologies/2020/3/GTDB#>
select ?numHostKilledk ?numHours ?hostageStatus where {
   ?inc :IncID "200109110004"^^xsd:string.
   ?inc :isHostKidn "true"^^xsd:boolean.
   ?inc :hasHostKid ?hostKid.
   ?hostKid :numHostKid ?numHostKilledk.
   ?hostKid :numHours ?numHours.
   ?hostKid :hasHostKidOutcome ?outcome.
   ?outcome rdf:type ?hostageStatus.
   ?hostageStatus rdfs:subClassOf :HostKidnappingOutcome.
}
```

This evaluation provided a sample of SPARQL used to answer all the competency questions in the functional requirements of the ORSD for the GTDOnto (Table 1).

5.3. GTDOnto Ontology Quality

The current ontology version utilizes all the concepts, relations, and properties described in the codebook for the GTD project maintained by START at the University of Maryland. Nevertheless, a logical evaluation was conducted by domain experts to verify the accuracy of the naming of concepts and to validate the hierarchy of the terms presented in GTDOnto ontology. Human assessment of an ontology evaluates its quality. Hence, experiments were conducted with experts from the University of Jordan Center of Strategic Studies (CSS). The center primarily studies and researches regional conflicts, international relations, and security.

The criteria described in [36] are assessed as part of the evaluation process. These criteria are listed below with an explanation of their application in GTDOnto.

- Accuracy: The ontology development was assisted by the guidelines from the GTD codebook. Furthermore, the ontology evaluation was assessed by domain experts from JCSS. Classes and relations were evaluated regarding the accuracy of terms developed in the GTDOnto, compared to the terms and concepts of the GTD codebook. The assessment was performed via a questionnaire sent to the experts via email, and further meetings were conducted to demo the GTDonto.
- Adaptability: Each concept in the GTDOnto is represented using URIs and can be reused by other linked datasets when published. Thus, the GTDOnto can be reused and extended easily, making it adaptable.
- Clarity: All the classes, subclasses, and properties names defined in the GTDOnto are non-ambiguous names and ease human readability, which facilitates the creation of individuals of incidents and their related concepts without confusion. Experiments with domain experts from the CSS assessed the clarity by comparing the incidents described in the GTD excel sheet to the incident representation in the GTDOnto (Figures 9 and 10).
- Completeness: The GTDOnto ontology can answer all the competency questions specified in the functional requirements presented in the ORSD document (Table 1). Details for answering these competency questions are discussed in the next section.
- Efficiency: Creating instances is simple due to the clarity of naming and relations between concepts. Furthermore, the process of querying the GTDOnto is seamless with the sample of individuals created. However, further investigation should be carried out with a more significant number of individuals.
- Consistency: No inconsistencies were found in GTDOnto after performing reasoning using HermiT 1.4.3.456 reasoner.

5.4. Metric-Based Ontology Evaluation

To provide further evaluation for GTDOnto, the schema-based metrics that address the design of the ontology provided by the OntoQA evaluation tool [35] are addressed below:

- **The number of classes**: The total number of classes in GTDOnto is 251, as indicated in Figure 3.
- **The number of properties**: The number of properties in GTDOnto is 78, the combination of data and object properties describing all the classes' attributes and their relations.
- **The number of root classes**: Despite having the Incident class as the main class for describing terrorist events, the GTDOnto has 19 root classes, indicating that the ontology is comprehensive and describes several concepts related to a global incident in its design.
- **Relationship Richness (RR)**: The relationship richness is represented as the percentage of the sub-class relationships between classes compared to all the possible connections between the ontology classes. It is computed as the ratio of the number of (non-inheritance) relationships (P), divided by the total number of relationships defined in the schema, where (P) is the number of (non-inheritance) relationships and (H) is the number of inheritance relationships using the equation

$$\mathrm{RR} \ = |\mathrm{P}|/(|\mathrm{H}| + |\mathrm{P}|)$$

In GTDOnto, the RR is around 0.08 due to many subclasses in the GTDOnto schema.

- **Inheritance Richness (IR)**: Inheritance richness is a good indication of how well knowledge is grouped into different concepts in the ontology. The IR is defined as the average number of subclasses per class (C):

$$\mathrm{IR} \ = |\mathrm{H}|/|\mathrm{C}|$$

In GTDOnto, the IR is around 0.9 since the ontology describes several concepts related to terrorist incidents, such as Attack types, Target types, Weapon types, and others.

- **Attribute Richness (AR)**: The number of attributes defined for each class indicates the amount of information conveyed describing incidents. The AR is calculated as the average number of attributes per class. It is computed as the number of attributes for all classes (att) divided by the number of classes (C):

$$\text{AR } = |\text{att}|/|\text{C}|$$

The AR in GTDOnto is around 0.32. In the previous section, visualizing instances of class *Incident* indicates how much knowledge is conveyed to the Incident class but not to other classes, such as *Attack*, *Weapon*, and *Target*.

Evaluating GTDOnto proved that it satisfied its main goal: to represent all the information about any incident in the GTD in a machine-readable format. The evaluation focused on assessing the applicability of GTDOnto based on representing GTD incidents in a task-based evaluation followed by competency questions answering. This evaluation proved that all the details about any terrorism incident are covered with GTDOnto representation. Furthermore, evaluation based on human assessment and competency questions answering assured that this work covers the information used to describe any terrorism incident in detail. Finally, schema-based evaluation of the GTDOnto focused on the design showed that the GTDOnto covers vast concepts related to the main class, which is the *Incident* class in the GTDOnto.

The development of GTDOnto is an ongoing effort. One goal is to keep the GTDOnto updated to cover all the classes and subclasses in the GTD database as the database update regularly. Furthermore, this GTDOnto is available for download for future efforts of researchers and developers to enhance. The uploaded ontology contains two instances from the GTD provided as examples that can be queried using SPARQL. In its current version, the GTDOnto can be the base for several applications in the future. We consider using the GTDOnto as the base for building a knowledge graph representing all the terrorism incidents of GTD, which will be helpful for further analysis tasks. We envision the GTDOnto expanding by incorporating rules to classify terrorism incidents of similar types.

Furthermore, GTDOnto terms and relations can be incorporated into tools for annotating content published on the web that might indicate terrorist-like intentions. Other than that, in the near future, we aim to reach out and work with the GTD project to integrate the GTDOnto with their efforts and publish the dataset in a machine-readable format. Furthermore, we hope publishing this dataset as part of the linked open data cloud will leverage the work and connect it with other existing datasets about the media or social media incidents.

6. Conclusions

The Global Terrorism Database (GTD) is made available by organizations such as the National Consortium for the Study of Terrorism and Responses to Terrorism (START). Experts in the domain gather this dataset, but technical modeling for its metadata is lacking. Hence, based on the guidelines of Scenario 1: from specification to implementation, from the NeOn ontology development methodology, we designed the GTD Ontology (GTDOnto). The aim was to model the incidents, targets, attackers, weapons, and other associated information and organize the knowledge on terrorism. Expanding on the work of START, this project aims to provide controlled vocabularies in a machine-readable, interoperable format, thereby establishing a conceptual model that can be utilized and expanded to characterize potential instances.

Furthermore, evaluation based on running examples, human assessment, and competency questions answering was undertaken to verify the utility of the developed ontology. Hence, future work will expand the GTDOnto to infer types of incidents based on specific criteria and examine the use of GTDOnto as an underlying schema to build a knowledge graph for terrorism. The work is an ongoing effort that we hope can be further used to serve researchers in this field for enhanced research, such as prediction and other downstream tasks.

Author Contributions: Conceptualization, R.Q.A.-F., M.A.-T., B.A.-S. and Z.E.; methodology, R.Q.A.-F. and M.A.-T.; software, R.Q.A.-F. and M.A.-T.; validation, R.Q.A.-F., M.A.-T., B.A.-S. and Z.E.; formal analysis, R.Q.A.-F. and M.A.-T.; data curation, R.Q.A.-F. and M.A.-T.; writing—original draft preparation, R.Q.A.-F., M.A.-T., B.A.-S. and Z.E.; writing—review and editing, R.Q.A.-F., M.A.-T., B.A.-S. and Z.E.; visualization, R.Q.A.-F., M.A.-T., B.A.-S. and Z.E. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: GTDOnto Ontology. Triplydb. https://triplydb.com/ReemAlfayez/GTDOnto/, accessed on 12 January 2020.

Acknowledgments: We would like to thank the Centre of Strategic Studies at the University of Jordan for sharing their expertise to review and evaluate the work presented in this paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Inkpen, R.; Gauci, R.; Gibson, A. The Values of Open Data. *Area* **2021**, *53*, 240–246. [CrossRef]
2. Mendes, P.S.F.; Siradze, S.; Pirro, L.; Thybaut, J.W. Open Data in Catalysis: From Today's Big Picture to the Future of Small Data. *ChemCatChem* **2021**, *13*, 836–850. [CrossRef]
3. Ruijer, E.; Grimmelikhuijsen, S.; van den Berg, J.; Meijer, A. Open Data Work: Understanding Open Data Usage from a Practice Lens. *Int. Rev. Adm. Sci.* **2018**, *86*, 3–19. [CrossRef]
4. Stieglitz, S.; Wilms, K.; Mirbabaie, M.; Hofeditz, L.; Brenger, B.; López, A.; Rehwald, S. When Are Researchers Willing to Share Their Data?—Impacts of Values and Uncertainty on Open Data in Academia. *PLoS ONE* **2020**, *15*, e0234172. [CrossRef] [PubMed]
5. Pietriga, E.; Gözükan, H.; Appert, C.; Destandau, M.; Čebirić, Š.; Goasdoué, F.; Manolescu, I. Browsing Linked Data Catalogs with LODAtlas. In Proceedings of the International Semantic Web Conference, Monterey, CA, USA, 8–12 October 2018; Springer: Berlin/Heidelberg, Germany, 2018; pp. 137–153. [CrossRef]
6. Alvite-Díez, M.L. Linked Open Data Portals: Functionalities and User Experience in Semantic Catalogues. *Online Inf. Rev.* **2021**, *45*, 946–963. [CrossRef]
7. Bizer, C.; Heath, T.; Berners-Lee, T. Linked Data the Story so Far. In *Semantic Services, Interoperability and Web Applications: Emerging Concepts*; IGI global: Hershey, PA, USA, 2011; pp. 205–227.
8. Park, S.; Gil-Garcia, J.R. Open Data Innovation: Visualizations and Process Redesign as a Way to Bridge the Transparency-Accountability Gap. *Gov. Inf. Q.* **2022**, *39*, 101456. [CrossRef]
9. Donaldson, D.R.; Koepke, J.W. A Focus Groups Study on Data Sharing and Research Data Management. *Sci. Data* **2022**, *9*, 1–7. [CrossRef] [PubMed]
10. Chapman, A.; Simperl, E.; Koesten, L.; Konstantinidis, G.; Ibáñez, L.D.; Kacprzak, E.; Groth, P. Dataset Search: A Survey. *VLDB J.* **2019**, *29*, 251–272. [CrossRef]
11. Brickley, D.; Burgess, M.; Noy, N. Google Dataset Search: Building a Search Engine for Datasets in an Open Web Ecosystem. In Proceedings of the World Wide Web Conference, WWW 2019, San Francisco, CA, USA, 13–17 May 2019; Association for Computing Machinery, Inc.: New York, NY, USA, 2019; pp. 1365–1375. [CrossRef]
12. Charalabidis, Y.; Zuiderwijk, A.; Alexopoulos, C.; Janssen, M.; Lampoltshammer, T.; Ferro, E. Open Data Interoperability. In *The World of Open Data. Public Administration and Information Technology*; Springer: Berlin/Heidelberg, Germany, 2018; pp. 75–93. [CrossRef]
13. Mannes, A.; Golbeck, J. Building a Terrorism Ontology. In Proceedings of the ISWC Workshop on Ontology Patterns for the Semantic Web, Galway, Ireland, 6–10 November 2005.
14. Mannes, A.; Golbeck, J. Ontology Building: A Terrorism Specialist's Perspective. In Proceedings of the IEEE Aerospace Conference, Big Sky, MT, USA, 3–10 March 2007; pp. 1–5.
15. Inyaem, U.; Haruechaiyasak, C.; Meesad, P.; Tran, D. Ontology-Based Terrorism Event Extraction. In Proceedings of the 2009 1st International Conference on Information Science and Engineering, Nanjing, China, 26–28 December 2009; pp. 912–915. [CrossRef]
16. START (National Consortium for the Study of Terrorism and Responses to Terrorism). Global Terrorism Database 1970–2020. Available online: https://start.umd.edu/gtd/ (accessed on 15 July 2022).
17. LaFree, G.; Dugan, L. Introducing the Global Terrorism Database. *Terror. Political Violence* **2007**, *19*, 181–204. [CrossRef]
18. Shadbolt, N.; Hall, W.; Berners-Lee, T. The Semantic Web Revisited. *IEEE Intell. Syst.* **2006**, *21*, 96–101. [CrossRef]
19. Gruber, T.R. A Translation Approach to Portable Ontology Specifications. *Knowl. Acquis.* **1993**, *5*, 199–220. [CrossRef]
20. Aminu, E.F.; Oyefolahan, I.O.; Abdullahi, M.B.; Salaudeen, M.T. A Review on Ontology Development Methodologies for Developing Ontological Knowledge Representation Systems for Various Domains. *Int. J. Inf. Eng. Electron. Bus.* **2020**, *2*, 28–39. Available online: https://www.mecs-press.org/ijieeb/ijieeb-v12-n2/IJIEEB-V12-N2-5.pdf (accessed on 10 January 2023). [CrossRef]

21. Uschold, M.; King, M. *Towards a Methodology for Building Ontologies*; Presented at "Workshop on Basic Ontological Issues in Knowledge Sharing"; Artificial Intelligence Applications Institute, University of Edinburgh: Edinburgh, UK, 1995.
22. Kotis, K.; Vouros, G.A. Human-Centered Ontology Engineering: The HCOME Methodology. *Knowl. Inf. Syst.* **2006**, *10*, 109–131. [CrossRef]
23. Noy, N.F.; McGuinness, D.L. Ontology Development 101: A Guide to Creating Your First Ontology. 2001. Available online: https://protege.stanford.edu/publications/ontology_development/ontology101.pdf (accessed on 10 December 2022).
24. Denaux, R.; Dolbear, C.; Hart, G.; Dimitrova, V.; Cohn, A.G. Supporting Domain Experts to Construct Conceptual Ontologies: A Holistic Approach. *J. Web Semant.* **2011**, *9*, 113–127. [CrossRef]
25. Suárez-Figueroa, M.C.; Gómez-Pérez, A.; Fernández-López, M. The Neon Methodology for Ontology Engineering. In *Ontology Engineering in a Networked World*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 9–34. [CrossRef]
26. Suárez-Figueroa, M.C. NeOn Methodology for Building Ontology Networks: Specification, Scheduling and Reuse. Ph.D. Thesis, Universidad Politécnica de Madrid facultad de informática, Madrid, Spain, 2010.
27. Zemmouchi-Ghomari, L.; Ghomari, A.R. Process of Building Reference Ontology for Higher Education. In Proceedings of the World Congress on Engineering: WCE, London, UK, 3–5 July 2013; pp. 1595–1600.
28. Fonou-Dombeu, J.V.; Achary, T.; Genders, E.; Mahabeer, S.; Pillay, S.M. COVIDonto: An Ontology Model for Acquisition and Sharing of COVID-19 Data. In Proceedings of the International Conference on Model and Data Engineering, Tallinn, Estonia, 21–23 June 2021; Springer: Berlin/Heidelberg, Germany, 2021; pp. 227–240. [CrossRef]
29. Ibanescu, L.; Dibie, J.; Dervaux, S.; Guichard, E.; Raad, J. PO2—A Process and Observation Ontology in Food Science. Application to Dairy Gels. In *Metadata and Semantics Research. MTSR 2016. Communications in Computer and Information Science*; Garoufallou, E., Subirats Coll, I., Stellato, A., Greenberg, J., Eds.; Springer: Berlin/Heidelberg, Germany, 2016; Volume 672, pp. 155–165. [CrossRef]
30. Suárez-Figueroa, M.C.; Gómez-Pérez, A.; Fernández-López, M. The NeOn Methodology Framework: A Scenario-Based Methodology for Ontology Development. *Appl. Ontol.* **2015**, *10*, 107–145. [CrossRef]
31. START. Global Terrorism Database Codebook: Inclusion Criteria and Variables. 2019. Available online: https://www.start.umd.edu/gtd/downloads/Codebook.pdf (accessed on 23 July 2022).
32. Suárez-Figueroa, M.C.; Gómez-Pérez, A.; Villazón-Terrazas, B. How to Write and Use the Ontology Requirements Specification Document. In *OTM Confederated International Conferences "On the Move to Meaningful Internet Systems"*; Springer: Berlin/Heidelberg, Germany, 2009; Volume 5871, pp. 966–982.
33. Sabou, M.; Fernandez, M. Ontology (Network) Evaluation. In *Ontology Engineering in a Networked World*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 193–212.
34. Raad, J.; Cruz, C. A Survey on Ontology Evaluation Methods. In Proceedings of the 7th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management, Lisbon, Portugal, 12–14 November 2015.
35. Tartir, S.; Arpinar, I.B.; Sheth, A.P. Ontological Evaluation and Validation. In *Theory and Applications of Ontology: Computer Applications*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 115–130. [CrossRef]
36. Vrandečić, D. Ontology Evaluation. In *Handbook on Ontologies*; Springer: Berlin/Heidelberg, Germany, 2009; pp. 293–313. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

big data and
cognitive computing

MDPI

Article

Refining Preference-Based Recommendation with Associative Rules and Process Mining Using Correlation Distance

Mohd Anuaruddin Bin Ahmadon [1,*,†], Shingo Yamaguchi [1,†], Abd Kadir Mahamad [2] and Sharifah Saon [2]

[1] Graduate School of Sciences and Technology for Innovation, Yamaguchi University, Tokiwadai 2-16-1, Ube 755-8611, Japan
[2] Faculty of Electrical and Electronic Engineering, Universiti Tun Hussein Onn Malaysia, Batu Pahat 86400, Johor, Malaysia
* Correspondence: anuar@yamaguchi-u.ac.jp

† This paper is an extended version of our paper published in Mohd Anuaruddin Bin Ahmadon, Shingo Yamaguchi, "Mining Consumer Services Based on User Preference with Associative and Process Mining", Proc. of ZINC 2021, May 2021; pp. 185–190 (Received Special Merit Award).

Abstract: Online services, ambient services, and recommendation systems take user preferences into data processing so that the services can be tailored to the customer's preferences. Associative rules have been used to capture combinations of frequently preferred items. However, for some item sets X and Y, only the frequency of occurrences is taken into consideration, and most of the rules have weak correlations between item sets. In this paper, we proposed a method to extract associative rules with a high correlation between multivariate attributes based on intuitive preference settings, process mining, and correlation distance. The main contribution of this paper is the intuitive preference that is optimized to extract newly discovered preferences, i.e., implicit preferences. As a result, the rules output from the methods has around 70% of improvement in correlation value even if customers do not specify their preference at all.

Keywords: process mining; associative mining; personalization; recommendation; decision support

Citation: Bin Ahmadon, M.A.; Yamaguchi, S.; Mahamad, A.K.; Saon, S. Refining Preference-Based Recommendation with Associative Rules and Process Mining Using Correlation Distance. *Big Data Cogn. Comput.* **2023**, *7*, 34. https://doi.org/10.3390/bdcc7010034

Academic Editors: Konstantinos Kotis and Dimitris Spiliotopoulos

Received: 24 November 2022
Revised: 4 February 2023
Accepted: 6 February 2023
Published: 10 February 2023

1. Introduction

In the context of online services, ambient services, and recommendation systems, user preferences are usually related to multiple attributes specified by the user based on certain factors such as environment, culture, and psychological factors. In general, preference analysis includes the perception and sentiment of the users toward selecting the target services or items. Users are becoming more attracted to tailored preferences in on-demand online services related to health, music, movies, food, and fashion. Online service providers are always keeping up with recommendation technology to tune and match user preferences so that current users will continue using their services. For some preference analyses, it is hard to recommend accurate results that match the user's experience due to the limitation of references in the database. A system that supports user decision-making or provides personalized services can only be implemented if the users explicitly define their preferences. In general, these systems can only provide excellent and meaningful results during certain events when these explicit and implicit preferences or actions take place. Based on the insight and related information, the systems can analyze the causal relationship between these references.

For recommendation of preferences, explicit and implicit references are always related to spatio-temporal concepts where the time and space of the users are considered. For example, a system can judge the user's motives for entering a space, such as entering a kitchen in a house to prepare breakfast in the morning. Smart systems can identify their implicit needs to suggest meals with low calories, such as coffee and scrambled eggs, by relating the reference to information on available ingredients in the house. The information here shows that users may have many other choices for breakfast, but it is hard to decide what to choose. Spatio-temporal reasoning is usually effective when the references include a sequence of actions, frequencies, and repetitions of the user's behavior. However, recommending implicit preferences requires a method to analyze the causal relationship with the user's explicit preference and determine the best selection based on the relationship.

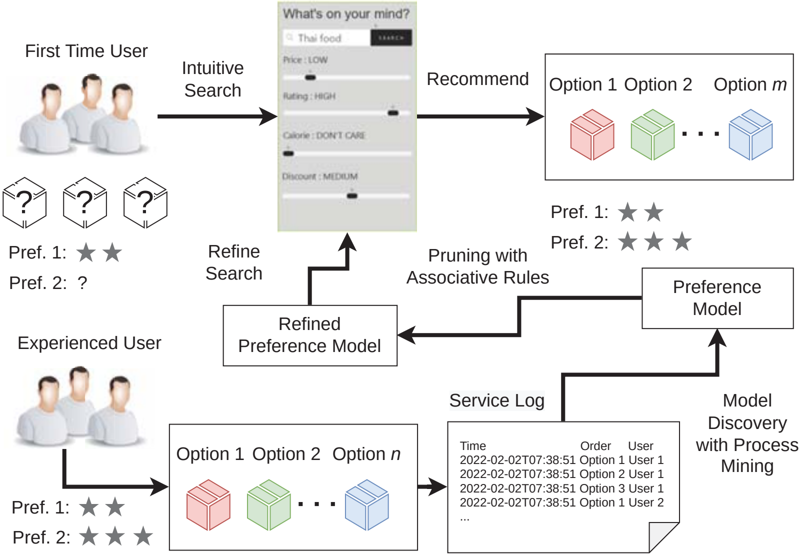
Learning and identifying the implicit preferences of users is challenging due to limited spatial-temporal references. Moreover, the causal relationship between explicit and implicit preferences must be strong enough for the recommendation to be meaningful and reliable. It is also essential to differentiate the user's *'needs'* and *'wants'*. Usually, the 'needs' of the users are defined explicitly. In contrast, identifying the 'wants' is the problem that should be addressed. Decisions are even harder for spatio-temporal needs to learn the specific attributes of the available options. Moreover, users are always influenced by the gain of making certain decisions and the fear of loss. Users usually do not make purely analytical decisions and are affected by sentiments, cultures, and psychological factors. Hence, we often see that users always make decisions by relying on other users' sentiments, news, or rumors. This is even harder for inexperienced or first-time users who spend extra time and effort comparing and choosing based on limited information.

The motivation of this research is many research focuses on finding explicit preferences from existing data such as product specification, sales log, reviews, and surveys but do not focus on highly related implicit preference. Explicit preference can be easily extracted from existing data mining methods such as clustering analysis, machine learning, associative mining, and genetic algorithm. However, the gap lies when extracting implicit preference [1]. He et al. [2] stated the problem in extracting implicit reference where even though the explicit preference was given, we need to determine the co-occurrences in the explicit preference of another user. However, there are some cases where users do not have any explicit preference at all. Implicit preference in our study is regarded as an *"unknown preference"* by the user. Moreover, the implicit preference must have strong relations with each other. It means that the customer only knows their implicit preference if some relation related to their explicit preferences is given. In this method, we use a combination of process and associative mining to extract implicit preferences even if the users do not specify their explicit preferences.

In this paper, we proposed a method to extract associative rules with a high correlation between X and Y based using process mining, associative mining, and correlation distance. The overview is shown in Figure 1. The figure shows that for first-time users, it is easier to make decisions intuitively. Therefore, they need an 'intuitive recommendation' based on their preference. Our approach takes a service log and extracts the preference model using process mining. Then the service model is refined by pruning associative rules. A result is an option for selection that discovers the implicit preference that even the user does not know before.

The main contributions of this paper are as follows:

1. A method that combines process mining and data mining to extract preference models for implicit preferences.
2. Extract implicit preferences that have a strong correlation between attributes even if the user sets some of their preferences as 'no-interest.'
3. The method outputs not a single choice of an item but multiple combinations of highly correlated items as recommendations to both first-time users and experienced users.



What's on your mind?
Thai food
Price : LOW
Rating : HIGH
Calorie : DON'T CARE
Discount : MEDIUM

```
Time                    Order    User
2022-02-02T07:38:51 Option 1 User 1
2022-02-02T07:38:51 Option 2 User 1
2022-02-02T07:38:51 Option 3 User 1
2022-02-02T07:38:51 Option 1 User 2
...
```

Figure 1. An overview of our proposed method.

The benefit of this study is that even though the users do not give any explicit preference, which is *'no preference'* set for all attributes of any item, the method can still extract what they might prefer by referring to the previous preference of previous customer as a starting point. If any explicit preferences are given, the preferences will be used as a reference. Intuitively, the user requires less effort but more flexibility in making a decision.

This paper is organized as follows; After the introduction, we give the preliminary to introduce process mining, association rule, and Cook's distance. Next, we define the problem of service preference extraction and explain the problem. Finally, we evaluate the proposed method to show its effectiveness.

2. Preliminary

2.1. Process Mining

Process mining [3] is a technique to extract process models from event logs such as sales logs, system logs, and business process logs. For example, a sales record of a transaction between a client and business or customer purchase records. We can utilize process mining to extract a process model called Petri net from an event log. Petri net [4] can represent the execution order or sequence of actions. Process mining links data and processes where some data can only be visualized as a process or as the action of a sequence. Process mining can be performed with process mining tools such as ProM [5], Disco [6], or RapidProM [7].

In this paper, we utilize a process mining method called inductive miner [8]. Inductive miner is a method to extract logically correct and sound processes. A logically correct process does not contain any conflict, such as deadlock or overflow. An inductive miner can extract sound Petri net or an equivalent form of Petri net called process tree [9]. The representation of the process using sound Petri net and process trees ensures that the process does not fall into spaghetti conditions. Therefore, the inductive miner is suitable for our approach. We utilize the Petri net to represent items' selections in sequences with strong correlation and frequency.

Concretely, the inductive miner extracts the process model in a block model representing a sequence, exclusive choice, parallel, and loop constructs. These four constructs represent the basic constructs in ost business process workflows. Business workflows, including customer services, can be modeled with Petri net.

2.2. Association Rule

An association rule represents a rule of antecedent and consequent. An association rule can be expressed with $\mathcal{A} \Rightarrow \mathcal{C}$ where $\mathcal{A}$ is the antecedent for consequent $\mathcal{C}$ such that when $\mathcal{A}$ occurs then $\mathcal{C}$ will occur. As an example, we take an association rule such as $outlook{=}sunny,\ windy{=}no \Rightarrow play{=}yes$ represents that if the outlook is sunny or windy, then do play outside. Association rule can be extracted from popular algorithms such as Apriori [10,11] or FPGrowth [12]. The rule is decided by proportional values such as support, confidence, and lift. To extract associative rules, we utilize the Apriori algorithm.

Let $\mathcal{A}$ or $\mathcal{C}$ be an itemset. Support $supp(L)$ is the number of instances that satisfy the rule. The confidence ratio is the ratio of a rule to be true in a dataset. It is given as in Equation (2).

$$conf(L \Rightarrow R) \; = \; \frac{supp(L \cup R)}{supp(L)} \tag{1}$$

Lift is the ratio of confidence and unconditional probability of the consequent. lift > 1.0 shows that L and R are dependent on each other.

$$lift(L \Rightarrow R) \; = \; \frac{supp(L \cup R)}{supp(L) \times supp(R)} \tag{2}$$

2.3. Cook's Distance

Cook's distance $Dist(i)$ [13] is used to measure the influence of an independent variable against another multi-dependent variable. The approach is by removing the target independent variable from the observation. The Cook's distance can be calculated with Equation (3). Equation (3) shows the average of y at observation j when i is removed from the observation. r is the regression model's coefficient.

$$Dist(i) \; = \; \frac{\sum_{j=1}^{n} (\hat{y}_{j(i)} - \hat{y}_j)^2}{r\hat{\sigma}^2} \tag{3}$$

The sum of $(\hat{y}_{j(i)} - \hat{y}_j)^2$ is calculated at each observation j where the distance is calculated based on the regression line of each observation when i-th observation is removed. Since all points on the distribution are considered, Cook's distance is calculated based on the regression by the concept of 'leave-one-out.' We can say that Cook's distance is the distance between the point of regression line produced by averaged y value and when i is removed from the observation. The calculated distance can measure the influence of i in the distribution group because Equation (3) averages the sum of residuals y with the MSE.

The given Cook distance shown in Equation (3) utilizes Manhattan distance [14] when calculating the absolute sum of $\hat{y}_{j(i)} - \hat{y}_j$. It is the L_1-norm (Manhattan's generalized form) derived from a multidimensional numerical distance called Minkowski distance [15] as shown in Equation (4). Given an L_p-norm in Equation (4), the sum of the absolute value of $\hat{y}_{j(i)} - \hat{y}_j$ in Equation (3) satisfies L_p-norm when p $=$ 1.

$$L_p(\hat{y}_{j(i)}, \hat{y}_j) \; = \; (\sum_{j=1}^{n} |\hat{y}_{j(i)} - \hat{y}_j)|^p)^{\frac{1}{p}} \tag{4}$$

Cook's distance is simply the distance between averaged regression value of $\hat{y}_{j(i)}$ (when i is removed) and the normal average value $\hat{y}_j$. By replacing the numerator with L_2-norm, we can obtain the Euclidean version of Equation (3).

Distance $Dist(i)$ is a metric if it satisfies (i) $\delta(x,y) \geq 0$ (non-negativity); (ii) $\delta(x,y)$ $= \; \delta(y,x)$ (symmetry); (iii) $\delta(x,y) \geq 0$ if and only if $x = y$ (coincidence axiom); and (iv) $\delta(x,y) \; \leq \; \delta(x,z) + \delta(z,y)$ (triangular inequality axiom) where $\delta(x,y)$ is the distance between x and y, and z is the point between them. The properties (i), (ii), and (iii) are trivial when x and y are obtained from absolute value, and if x and y are the same, the distance is 0, also symmetrically, the distance is the same. To show Cook's distance satisfies the triangular inequality axiom in (iv), we can utilize the generalized form of Manhattan distance and Euclidean distance shown in Equation (4). Since Cook's distance is L_1-norm, it is well known that we can set $p = 2$ to obtain the L_2-norm (Euclidean distance). Therefore, we can write the distance $Dist_{L_2}(i)$ shown in Equation (5). From Equation (5), we can identify that $Dist_{L_2}(i)$ is Euclidean and satisfies the triangular inequality axiom.

$$Dist_{L_2}(i) = \frac{1}{r\sigma^2}\sqrt{\sum_{j=1}^{n}(y_{j(i)} - \hat{y}_j)^2} \tag{5}$$

As stated by Chai et al. [16], the value of σ^2 satisfies triangular inequality (see Equation (6)). The regression coefficients are represented by p, and σ^2 is the regression's Mean Squared Error (MSE). The value of σ^2 is the Euclidean distance of residual error on how close a point y is to the regression line (when averaging the y values), assuming that the distribution is a Gaussian distribution. The value of σ^2 can be calculated as in Equation (6). The Euclidean distance is divided with $\sqrt{n}$ where n is the number of samples in the distribution.

$$\sigma^2 = \sqrt{\frac{\sum_{j=1}^{n}(y_{(j)} - \hat{y}_j)^2}{n}} \tag{6}$$

The residuals $y_j(j = 1, 2, \cdots, n)$ can be represented by n-dimensional vector V. Let X, Y, and Z be n-dimensional vectors. Based on the metric properties [17], Equation (7) shows that σ^2 satisfies the triangular inequality axiom [16] such that

$$\sqrt{\frac{1}{n}\sum_{j=1}^{n}(x_j - y_j)^2} \leq \sqrt{\frac{1}{n}\sum_{j=1}^{n}(x_j - z_j)^2} + \sqrt{\frac{1}{n}\sum_{j=1}^{n}(y_j - z_j)^2} \tag{7}$$

We can also obtain L_2-norm (the Euclidean version) of Cook's distance as follows:

$$Dist(i) = \frac{\sqrt{\sum_{j=1}^{n}(y_{j(i)} - \hat{y}_j)^2}}{r\sqrt{\frac{\sum_{j=1}^{n}(y_{(j)} - \hat{y}_j)^2}{n}}} = \frac{1}{r}\sqrt{\frac{\sum_{j=1}^{n}(y_{j(i)} - \hat{y}_j)^2 \times n}{\sum_{j=1}^{n}(y_{(j)} - \hat{y}_j)^2}} \tag{8}$$

Cook's distance is usually used for outlier detection where the outlier value deviates far from other independent variables' values. From Equations (3) and (8), $Dist(i)$ can be used to remove outlier or weak associative rules. For simplicity, we utilize Equation (3) in this paper.

3. Related Work

Mining customer preferences involves various parameters and decision support tools such as user preferences attributes, product specifications, and sentiments. Many related works focus on both user preferences and product specifications. The commonly used method includes cluster analysis, machine learning, genetic algorithm, and associative mining. The related works are shown in Table 1.

Clustering analysis groups preferences by performing clusters on the available data. Zhang et al. [18] consider product competitiveness when mining customer preferences. They proposed an information mining method that utilizes entropy and density-based clustering analysis to the customer preferences. Chong et al. [19] proposed a method to identify preference by clustering items based on multi-attributes such as consumer sensory scores. Seo et al. [20] proposed a recommender system for a group of items that are based on genre preference that may reduce clustering computation cost. Other clustering-based analyses were also proposed by Osama et al. [21] and Wang et al. [22].

Table 1. Summary of related works.

Literature	Year	Clustering Analysis	Machine Learning	Genetic Algorithm	Collaborative Filtering	Associative Mining	Process Mining
Zhang et al. [18]	2022	√					
Chong et al. [19]	2020	√					
Seo et al. [20]	2021	√					
Osama et al. [21]	2019	√					
Wang et al. [22]	2019	√					
Xiao et al. [23]	2022		√				
Zheng et al. [24]	2022		√				
Sun et al. [25]	2021		√				
Aldayel et al. [26]	2020		√				
Bi et al. [27]	2020		√				
Gkikas et al. [28]	2022			√			
Das et al. [29]	2022			√			
Jiang et al. [30]	2019			√			
Petiot et al. [31]	2020			√			
Alhijawi et al. [32]	2020			√	√		
Liu et al. [33]	2022				√		
Liang et al. [34]	2022				√		
Valera et al. [35]	2021				√		
Fkih et al. [36]	2021				√		
Davis et al.[37]	2021				√		
Qi et al. [38]	2022					√	
Tan et al. [39]	2020					√	
Chen et al. [40]	2021					√	
Ait-Mlouk et al. [41]	2017					√	
Kaur et al. [42]	2016					√	
Our Method	2023					√	√

Machine learning can predict user preferences by learning from recorded transaction data. Xiao et al. [23] focus on the sentiment tendencies of the customer to extract their preferences. These tendencies are fine-grained to improve the performance of the analysis. The fine-grained sentiment analysis problem is converted into a sequence labeling problem to predict the polarity of user reviews. Since the problem involves sentiment analysis, the user-feature focus on the review dataset with text information, such as words with emotional words. Conditional Random Field (CRF) and neural networks were applied to analyze the text sequence. Zheng et al. [24] focus on immersive marketing and applied graph neural network models that consider essential attributes to improve the consumer shopping experience. Other related works related to machine learning were also proposed by Sun et al. [25], Aldayel et al. [26], and Bi et al. [27].

Genetic algorithms can find the most optimal preferences from various preferences patterns based on evolutionary algorithms. Gkikas et al. [28] proposed a combination of a method using binary decision trees and genetic algorithm wrappers to enhance marketing decisions. They focus on customer survey data to classify customer behaviors. As a model to classify customer behavior, Optimal decision trees are generated from binary decision trees, representing the chromosomes handled in the genetic algorithm wrapper. Das et al. [29] used a genetic algorithm to predict the premium of life insurance based on consumer behavior toward the insurance policies before and after-pandemic situations. Other work was proposed by Jiang et al. [30] and Petiot [31].

Collaborative filtering collects preferences from many customers and predicts a user's preferences. Alhiijawi et al. [32] applied a genetic algorithm with collaborative filtering to generate recommended preferences using multi-filtering criteria. Liu et al. used weighted attribute similarity and rating similarity in collaborative filtering that can alleviate data sparseness. Liang et al. [34] focus on diversifying recommendations using neural collaborative filtering that can achieve high diversity in a recommendation. Valera et al. [35] proposed a method that uses collaborative filtering not for single-user preferences but for group preferences. The method takes into account taking individual preferences and context in the group. Other work includes Fkih et al. [36] and Davis et al. [37].

Associative mining extracts associative rules from available data to recognize patterns based on basket analysis. Qi et al. [38] proposed an algorithm that utilized weighted associative rules to discover frequent item sets with high values. Tan et al. [39] proposed top-k rules mining based on MaxClique for contextual preferences mining. In the method, they applied the problem to association rules extracted from preference databases. They offered a conditional preference rule with context constraints to model the user's positive or negative interests. Other work includes Ait-Mlouk et al. [41] and Kaur et al. [42].

The given related works focus only on the interestingness of an item, such as buyer sentiments towards one attribute, i.e., rating or prices. However, it is hard to simultaneously extract implicit preferences with multi-variate attributes such as price, rating, and discount. There exists a trade-off between choices and attributes of target items. Moreover, it is hard for users to decide on many multi-variate attributes simultaneously. Therefore, there is a need to balance between choices and attributes in customer preferences.

4. The Problem of Implicit Preference Extraction

First, we formalized the problem of extracting service preference from the service model representing the business rule based on the sales log. First, we define preference as follows:

Definition 1 (Preference)**.** *A preference σ is denoted by n-tuple $(\alpha_1, \alpha_2, \cdots, \alpha_n)$ where α_n is called as preference attribute of σ.*

We formalize a problem which is to achieve a goal for service preference extraction.

Definition 2 (Service Preference Extraction Problem)**.**
Input: *Sales log S containing items $i_1, i_2, \cdots, i_n$, explicit preferences set $P = (\alpha_1, \alpha_2, \cdots, \alpha_n)$*
Output: *Implicit preferences set $P' = (\beta_1, \beta_2, \cdots, \beta_n)$*

Based on the problem definition above, we input a sales log S and explicit preferences $P = (\alpha_1, \alpha_2, \cdots, \alpha_n)$. The sales log contains items $i_1, i_2, \cdots, i_n$. The preferences $\alpha_1, \alpha_2, \cdots, \alpha_n$ corresponds to each item $i_1, i_2, \cdots, i_n$. Example of explicit preferences P and implicit preferences P' can be given as $P = (High, Low, No-Interest)$ and $P' = $ (*High, Low, High*) where abstract attributes value such as *High*, *Low* and *No-Interest* corresponds to the preference of item $(Price, Rating, Discount)$. Here, the implicit preference reveals *No-Interest* in attribute *Discount* can be replaced with *High* value.

First, we extract implicit preferences set $P' = (\beta_1, \beta_2, \cdots, \beta_n)$ then output the new preference P' to represent the choices of items $i_1, i_2, \cdots, i_k$ that have strong correlation between attributes. The implicit preferences do not satisfy $\beta_n = \alpha_n$ for all of its elements. If $\beta_n = \alpha_n$, then the explicit preference for item i_n does not change. However, if there is at least one element that satisfies $\beta_n \neq \alpha_n$, we can call $P' = (\beta_1, \beta_2, \cdots, \beta_n)$ as implicit preference. Therefore, we define implicit preference as follows:

Definition 3 (Implicit Preference)**.** *For a given item set $I = \{i_1, i_2, \cdots, i_n\}$ and its explicit preference $\sigma = (\alpha_1, \alpha_2, \cdots, \alpha_n)$, implicit preference is defined as $\pi = (\beta_1, \beta_2, \cdots, \beta_n)$ where α_n and β_n satisfy the following:*

(i) There exists at least one $\beta_n \neq \alpha_n$ such that each α_n and β_n represents the preference of item i_n.

(ii) Item set $I_{\mathcal{E}} \subseteq I$ for explicit preference σ and item set $I_{\mathcal{I}} \subset I$ for implicit preference π satisfy $(I_{\mathcal{I}} \subset I_{\mathcal{E}})$.

Definition 3 denotes implicit preference $\pi = (\beta_1, \beta_2, \cdots, \beta_n)$ that satisfies $\beta_n \neq \alpha_n$, respectively. Implicit preferences are regarded as preferences that are not shown in explicit preferences. However, no implicit preferences are extracted if $\beta_n = \alpha_n$ holds for all elements.

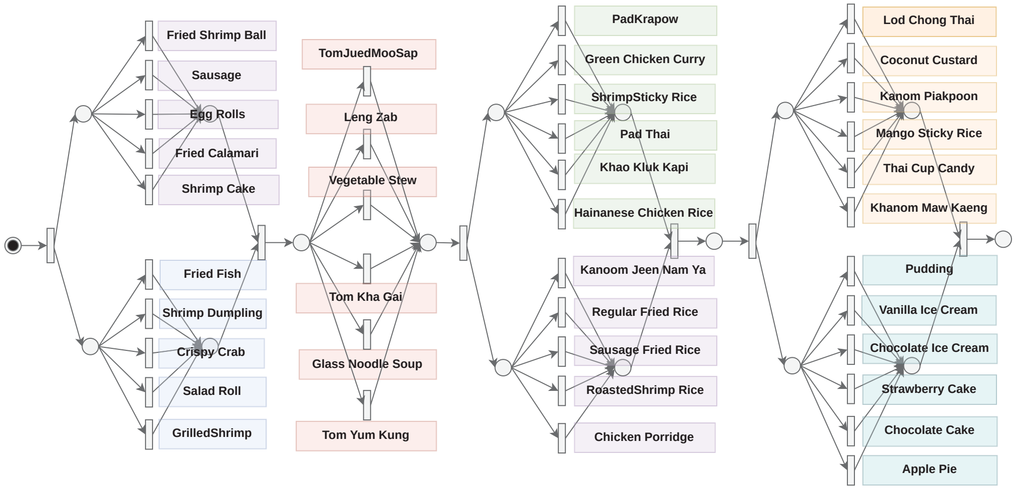
We use an online ordering system as an example to demonstrate our ideas and approach. The online ordering system takes orders for a Thai cuisine menu. The menu can be modeled with Petri net as shown in Figure 2. Figure 2 is represented by a Petri net that shows a selection of items from the menu in a sequence form from left to right. Each node and arc shown in the model represents a possible combination of the items. The menu contains eight categories, which are *Appetizer1*, *Appetizer2*, *Soup*, *MainCourse1*, *MainCourse2*, *Dessert1*, and *Dessert2* is shown in Figure 2 as a group of connected nodes and arc from left to right. The combination of items can be decided by looking at each attribute, i.e., *Price*, *Rating*, *Calorie*, *Discount*. The customer can choose to select one from each category as their choice from the course menu. The menu shows the restaurant's price, rating, calories, and discounts. The combination of selections that the user can make is 192,400 combinations. Therefore, it is hard for users to decide on the course that suits their preferences.

For further explanation, we give an example of three customers with different preferences. Customers usually express their preferences intuitively. We can conclude that customers make decisions based on specific attributes, but it is hard to look into the details of attribute values, especially for first-time customers. Moreover, they usually depend on the server or service advisor for recommendations. Therefore, the server or service advisor must make suitable recommendations that satisfy the customer. Let us consider the preference of the following customers:

1. *Alice*: She prefers food with low calories and prices. She will usually ask, "I prefer a healthy diet. What is the food with low calories but not too expensive." However, she might be implicitly attracted to try highly-rated food or discounted menu.
2. *Bob*: He prefers food with a less expensive and good rating. Therefore, he will ask, "What is the best affordable food you recommend?". It seems he does not care about calorie consumption and is also not interested in discounted food.
3. *John*: He has no specific preference. Therefore, he will ask, "What is today's special?" or "What is your recommendation?".

Since *Alice* prefers low-price and low-calorie, we can offer a new menu that allows the customer to choose items that strongly relate to low-price and low-calorie, for example, *Grilled Shrimp* for appetizers are low cost and always requested with *Pudding* because it has low calories. However, because only appetizers and desserts have a frequent pattern for a low price and low calories, other types of items more explanation, we'd like to give you Jued for soup and *Kanoom Jeen Namya*, will also be requested by the customer because they have attributes with low price and low calorie. In the case of Bob, he prefers low-price but good ratings. Similarly, he might be attracted to calorie and discounted food if he looks into the details. In the case of John, it is the hardest to meet his demands since he also needs to know what he prefers so that he will set all his preferences as *No-Interest*.

Customers with explicit preferences will be restricted to a few uninteresting choices. Fewer choices may reduce repeat customers. The case is similar to Bob, who prefers inexpensive food with high ratings where he does not care about calorie consumption and discount. Sometimes, customers such as John do not know what he likes. Therefore, to respond to preferences that are not specific, the customers should be given a range of attractive selections on the menu that might satisfy their implicit preferences.



Appetizer 1	**Soup**	**Main Course 1**	**Dessert 1**
Fried Shrimpball (30, 1, 468, 0)	Tom Jued Moo Sap (40, 2.7, 80, 0)	Pad Krapow (35, 5, 372, 0)	Lod Chong Thai (20, 1.8, 215, 0)
Sausage (30, 2, 224, 0)	Leng Zab (110, 5, 140)	Green Chicken Curry (60, 2.4, 240, 0)	Coconut Custard (20, 4, 540, 10)
Egg Rolls (40, 3, 480, 5)	Vegetable Stew (55, 4.1, 180, 5)	Shrimp Sticky Rice (60, 2.8, 477, 20)	Kanom Piakpoon (20, 1.8, 172, 0)
Fried Calamari (80, 4, 187, 0)	Tom Kha Gai (60, 3.8, 357, 10)	Pad Thai (90, 5, 486, 0)	Mango Sticky Rice (30, 5, 270, 0)
Shrimp Cake (120, 5, 990, 20)	Glass Noodle Soup (80, 2.9, 300, 0)	Khao Kluk Kapi (120, 4.7, 1028, 20)	Thai Cup Candy (30, 4, 1000, 20)
	Tom Yum Kung (90, 5, 280, 0)	Hainanese Chicken Rice (80, 3.5, 597, 0)	Khanom Maw Kaeng (40, 4, 244, 0)
Appetizer 2		**Main Course 2**	**Dessert 2**
Fried Fish (35, 5, 199, 0)		Kanoom Jeen Nam Ya (45, 2, 81, 5)	Pudding (25, 3.6, 120, 10)
Shrimp Dumpling (60, 4, 300, 5)		Regular Fried Rice (60, 3, 790, 0)	Vanilla Ice Cream (25, 1, 5, 330, 15)
Crispy Crab (30, 4, 544, 0)		Sausage Fried Rice (70, 1.2, 610, 15)	Chocolate Ice Cream (25, 2, 335, 25)
Salad Roll (60, 3, 1182, 10)		Roasted Shrimp Rice (80, 3.7, 510, 0)	Strawberry Cake (40, 3.2, 170, 0)
Grilled Shrimp (83, 2.7, 125, 10)		Chicken Porridge (50, 4.2, 228, 25)	Chocolate Cake (45, 5, 424, 0)
			Apple Pie (55, 4.3, 296, 10)

Figure 2. Menu of items with their attributes $(\mathit{Price}, \mathit{Rating}, \mathit{Calorie}, \mathit{Discount})$.

5. Preference Refinement by Associative Rule and Process Mining

We illustrate the details of our method in Figure 3. First, we extract frequent items that include associative rules with low confidence and lift value using Apriori. Then, we remove irrelevant rules that are (i) duplicate rules, (ii) not satisfying user preferences, and (iii) outlier rules in which correlation distances are too high.

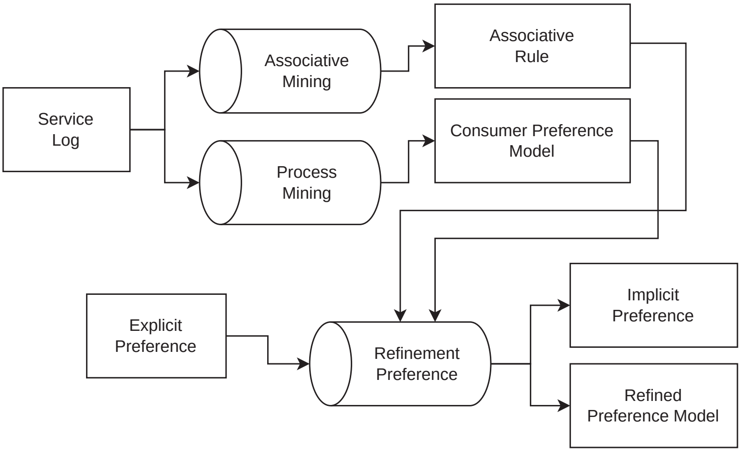
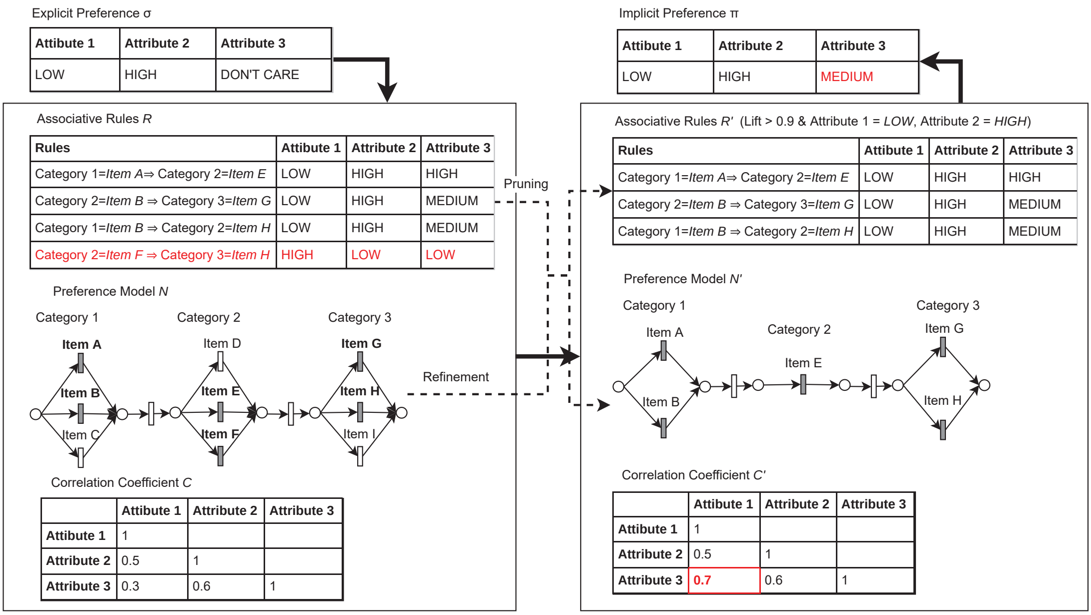


Figure 3. Our approach. We used data mining and process mining to extract the customer behavior model.

Figure 4 shows the overview of our approach. Given a preference P of some attributes such as *Attribute* 1, *Attribute* 2, and *Attribute* 3. For example, *Price*, *Calorie* and *Rating*. These attributes as given as $(Low, High, No-Interest)$ where each represents the preference of *Price*, *Calorie*, and *Rating*. The value can describe a range of values, such as 0 to 100. For example, if the value is 0 to 50, then the value is represented by *Low*; if the value is between 51 to 100, then the value is represented by *High*. If no preferences are given, then the value can be between 0 to 100, which $No-Interest$ represents.



Explicit Preference σ

Attibute 1	Attribute 2	Attribute 3
LOW	HIGH	DON'T CARE

Associative Rules *R*

Rules	Attibute 1	Attribute 2	Attribute 3
Category 1=*Item A*⇒ Category 2=*Item E*	LOW	HIGH	HIGH
Category 2=*Item B* ⇒ Category 3=*Item G*	LOW	HIGH	MEDIUM
Category 1=*Item B* ⇒ Category 2=*Item H*	LOW	HIGH	MEDIUM
Category 2=*Item F* ⇒ Category 3=*Item H*	HIGH	LOW	LOW

Correlation Coefficient *C*

	Attibute 1	Attribute 2	Attribute 3
Attibute 1	1		
Attribute 2	0.5	1	
Attribute 3	0.3	0.6	1

Implicit Preference π

Attibute 1	Attribute 2	Attribute 3
LOW	HIGH	MEDIUM

Associative Rules *R′* (Lift > 0.9 & Attribute 1 = *LOW*, Attribute 2 = *HIGH*)

Rules	Attibute 1	Attribute 2	Attribute 3
Category 1=*Item A*⇒ Category 2=*Item E*	LOW	HIGH	HIGH
Category 2=*Item B* ⇒ Category 3=*Item G*	LOW	HIGH	MEDIUM
Category 1=*Item B* ⇒ Category 2=*Item H*	LOW	HIGH	MEDIUM

Correlation Coefficient *C′*

	Attibute 1	Attribute 2	Attribute 3
Attibute 1	1		
Attribute 2	0.5	1	
Attribute 3	0.7	0.6	1

Figure 4. The detailed we added the explanation of red part overview of our approach. The input is explicit preference σ, and the output is implicit preference π. Both preferences are bounded by attributes and items in associative rules R and preference model N. The red part shows preferences that adapt to changes.

From here, the preferences are taken as a reference when extracting the associative rules from the service log. The associative rules can be represented such as $(Category\,1 = Item\ A)$ $\Rightarrow (Category\,2 = Item\ E)$. $Category\,1$ is the category of item that corresponds to the general options available in the services. For example, *Appetizer*, *Main Dish*, *Soup*, and, *Desert*. Each item, such as $Item\ E$, represents the selection within the category. For example, for *Appetizer*, some items such as $Item\ E$ are available for selection. The rules are accompanied by a set of attributes with values calculated from the average of the attributes of each item found in the rules. For example, for rules $(Category\,2 = Item\ F) \Rightarrow (Category\,3 = Item\ H)$, the preference of attributes found in the rules is $(High, Low, Low)$.

Next, the correlation between attributes is calculated. Based on the correlation values, i.e., Pearson correlation [43], we filter out some rules with attributes that have a high distance value using Cook's distance [13]. Then, we recalculate the correlation and value of attributes of the rules. Since the rules with high correlation were pruned, we can obtain the new preferences P' that are optimized for the customer. The changes are then used for the refinement of the service workflow. The new service workflow represents a workflow that satisfies the new preference P'. The rules which satisfy Lift ≥ 0.9 are preserved. Associative rules with a Lift value that is larger than 1 show a strong relationship between an item set X and Y. Rules with Lift ≥ 1 are considered strong. Note that the value is not absolute; some rules with a high lift value do not always have higher confidence than those with a lower lift value. If we increase the threshold of the Lift value, the number of extracted rules will be reduced. Therefore, we recommend reducing the Lift value to Lift ≥ 0.9.

The next step is to find out which menu the customer prefers. We set the parameters within a specific range to identify the relationship between items. For example, for *Price* attribute, we set the range between 0.0 to 80.0 as *Low*, and 81.0 to 140.0 as *High*. This range can be decided using discretization such as the binning method [44] or pre-defined by the user. We call the set of values as preference class $\mathcal{X}$.

Definition 4 (Preference Class)**.** *For a given attribute value α, the value of α can be represented with preference class γ if α ranges between $[u, v]$ denoted by $(\gamma, [u, v])$.*

For example, we separate *Low* and *High* values based on the median value. Therefore, we can give the parameters of preference α based on preference class $\mathcal{X}$ as follows:

(i) *Price* : (Low, $[0, 80]$), (High, $[81, 140]$)
(ii) *Rating* : (Low, $[0, 3]$), (High, $[3.1, 5]$)
(iii) *Calorie* : (Low, $[0, 600]$), (High, $[601, 1040]$)
(iv) *Discount* : (Low, $[0, 10]$), (High, $[15, 30]$)
(v) *No-Interest* : (NI, $[0, \infty)$)

As a practical method to capture customers' intuitive preferences, Based on the range of values, we allow setting the preferences intuitively rather than giving specific values.

First, we extract the set of associative rules R from the sales logs. A threshold of more than 0.9 is used for Apriori. Next, we extract the customer process model using process mining. The process model represents the ordering sequence of items shown in the sales log. Then, all rules with a low Pearson's correlation coefficient, i.e., less than 0.5 are removed. For rules r_i, we check the regression. If the Cook's distance of r_i is more than the threshold $h = 1.5$, we remove the rule r_i. The reason for setting the threshold value of 1.5 times is that most distance that is too high exceeds 1.5 times more said to deviate from the average value. However, depending on the analysis of the services, the user must decide on a suitable value.

The procedure for extracting implicit preference is given in Procedure ≪Procedure of Implicit Preference Extraction≫. The procedure discovers the process model N, and extracts the set of associative rules R. Then the procedure filters out weakly correlated associative rules and items in preference model N and the set of rules R using Cook's distance. Table A2 shows the rules that satisfy the explicit preferences, and the Pearson correlation value is shown for each relation between attributes *Price-Rating* (PR), *Price-Calorie* (PC), *Price-Discount* (PD), *Rating-Calorie* (RC), *Rating-Discount* (RD), and *Calorie-Discount* (CD). The table also shows the number of rules extracted. We only focus on preferences that can extract more than 30 variations of rules. Note that the set of rules also includes duplicated rules at this phase. From the set of rules, we filter out the weak rules that have weak correlation values.

We utilize Cook's distance and associative mining. Cook's distance characteristic is that during regression, it can detect highly influential items from a set of items. The measurement is regarded as the value of the influence of an item if it is removed from the group. In Cook distance, we can control how large the influence can be tolerated for an item by maintaining the sensitivity of the distance. For example, as a rule of thumb, a distance 1.5 times above the mean indicates that an item correlates less with other items in the group. Depending on the situation, the user can adjust the control parameter. Cook's distance is a distance based on regression and is suitable for multivariate analysis. Therefore, we handle all items as a highly correlated group with high sensitivity between them. Items that are far from the group have more distance. In Cook's distance, we perform a least-squares regression analysis to measure the influence of one item on another item in the same group.

Figure 5 illustrates the overview of removing weak correlated multivariate associative rules with our method. The rule is removed by averaging the value of $\alpha_1, \alpha_2, \cdots, \alpha_i$ when α_i is removed from the attributes observation i. Value u and v represent the value of each rule shown as $\circ$. The dotted line between $\circ$ and the regression line (solid line) shows the residual of the regression. The absolute difference $(\hat{y}_{j(i)} - \hat{y}_j)^2$ is averaged by regression MSE using the residual value. Implicit preference Π can be produced from the difference of explicit preference before and after the removal such that $\Pi = \sigma - \sigma'$.

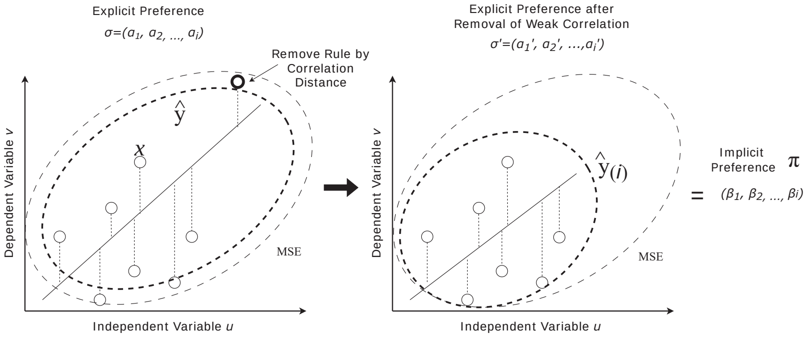


Figure 5. Overview of removing weak correlated multivariate rules and how residual difference $\hat{y}_{(i)} - \hat{y}$ can produce implicit preference Π.

Next, to calculate the cut-off point, we utilize Cook's distance, denoted by $Dist(r_m)$ as shown in Equation (3). We calculate the correlation distance if the set or extracted associative rules is not an empty set such as $R \neq 0$. Therefore, the equation can be given as in Equation (9). $cutoff(Dist(R)$ is the adjusted mean value for the cut-off point of the correlation distance.

$$cutoff(Dist(R,h)) = \begin{cases} h \times \frac{1}{m} \sum_{i=1}^{n} Dist(r_m), & \text{if } R \neq \phi, r_m \in R \\ 0, & \text{otherwise} \end{cases} \tag{9}$$

The value of $mean(Dist(R))$ denotes the mean of the correlation distance calculated by Cook's distance. The sensitivity of the outlier removal is controlled by variable h, which is the distance ratio from the mean value that is calculated from $Dist(r_m)$ using Cook's distance formula shown in Equation (3).

Algorithm 1 shows our main procedure in extracting the implicit preference. The complexity of the Algorithm 1 is $O(|R||\mathcal{A}| + |R|^2 + |T||R|)$ where $|R|$ is the number of rules, $|\mathcal{A}|$ is the number of preference attributes and $|T|$ is the number of action labels. The ≪Procedure of Implicit Preference Extraction≫ shows the whole procedure using process mining, associative mining, and implicit preference extraction using Cook's distance.

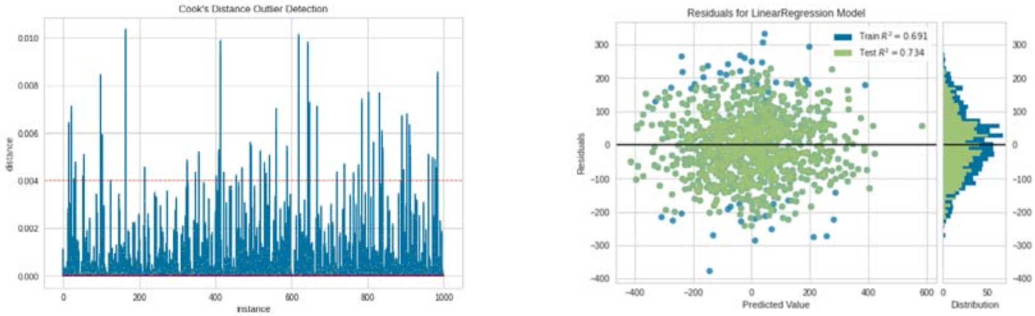
≪Procedure of Implicit Preference Extraction≫
Input: Sales Log S, set explicit preference $\Sigma = (\sigma_1, \sigma_2, \cdots, \sigma_m)$, lift threshold $\mathcal{T}$
Output: Refined Preference model N' and set of implicit preference $\Pi = (\pi_1, \pi_2, \cdots, \pi_n)$

1° Discover process model $N = (P, T, A)$ from order log E using process mining.
2° Extract associative rules $R = \{r_1, r_2, \cdots, r_n\}$ where $r_n = (X, Y, \mathcal{A})$ from event log E which satisfies Lift$\geq\mathcal{T}$.
3° Extract implicit preference Π for N using Algorithm 1.
4° Output refined process model with implicit preferences (N', π) and stop.

Algorithm 1: Implicit Preference Extraction

```
Input: Preference Model $N = (P, T, A)$, Explicit preference $\sigma = (\alpha_1, \alpha_2, \cdots, \alpha_m)$,
    Cut-off threshold $h$, Set of associative rules $R$
Output: Refined Preference model $N'$ and implicit preference $\pi = (\beta_1, \beta_2, \cdots, \beta_n)$
$R' \leftarrow \emptyset$, $\delta \leftarrow 0$
for each $r_i \in R$ do
    for each $\alpha_m \in \mathcal{A}$ of $r_i$ do
        if $\alpha_m \simeq \sigma_m$ then
            $R' \leftarrow R' \cup \{r_i\}$ ▷ Add $r_i$ to new set of rules $R'$ if $\alpha_m$ satisfies $\sigma_m$
        end if
    end for
end for
for each $r_i \in R'$ do
    for each $r_j \in R' (i \neq j)$ do
        if $r_i : X \Rightarrow Y$ and $r_j : Y \Rightarrow X$ then
            $R' \leftarrow R' - \{r_i\}$ ▷ Remove duplicated rules
        end if
    end for
    if Cook's distance $Dist(r_i) \geq cutoff(Dist(R', h))$ then
        $R' \leftarrow R' - \{r_i\}$ ▷ Remove outliers with low correlation
    end if
end for
for each task label $t \in T$ do
    for each $(r_i : X \Rightarrow Y) \in R'$ do
        if $t \notin X$ or $t \notin Y$ then
            $T' \leftarrow T - \{t\}$ ▷ Remove label t from N if t is not in item sets X or Y
            $N' \leftarrow N(P, T', A)$ ▷ Create new process model $N'$ with $T'$
        end if
    end for
    for each $\alpha \in \sigma$ do
        $\delta \leftarrow mean(T', \alpha_m)$ ▷ Calculate mean of attribute $\alpha_m$ for each $t \in T'$
        $\beta_m \leftarrow prefClass(\delta, \mathcal{X})$ ▷ Set $\beta_m$ with value $\delta$ by mapping to preference class $\mathcal{X}$
        $\delta \leftarrow 0$
    end for
    $\pi \leftarrow (\beta_1, \beta_2, \cdots, \beta_m)$ ▷ Construct the implicit preference $\pi$
end for
return $(N', \pi)$ ▷ Output refined process model with implicit preferences and stop
```

The proposed procedure utilizes the mechanism of Cook's distance when removing one variable α from the observation. For example, for a given preference on *Price*, *Rating*, *Calorie*, and *Discount*, we can observe the influence of *Price* by setting *Price* as target i. We can observe the improvement of residual coefficient R^2 in the regression. Figure 6a shows the distance for each item of $n = 1000$ rules. The cutoff line shows the threshold at $4/n$. Figure 6b compares residuals before and after removing weakly correlated rules. The R^2 value was 0.691, but the normal distribution distorted slightly to the positive value. Figure 6b shows the residual coefficient improved to 0.736. Figure 6b shows that the correlation improved because the distribution changed from a more dispersed distribution into a tighter distribution resulting in a higher R^2 value. The green dots represent the distribution after weakly correlated rules were removed, and the blue dot shows the distribution before the removal. We can see the blue dots are the outliers that were removed from the set of rules. Here, we confirmed that Cook's distance is effective in our procedure.



(**a**) Cook's distance-based cutoff for attribute α_i.

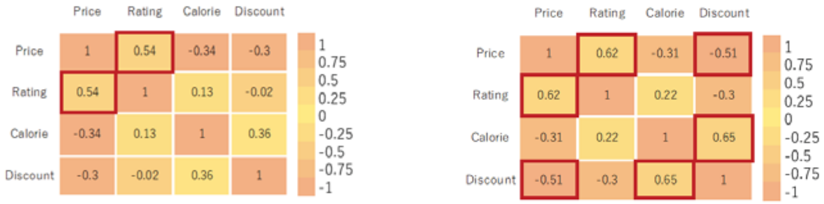
(**b**) Residual plot comparison.

Figure 6. The effect on residual after using Cook's distance. The residual coefficient improved after removing weakly correlated rules.

6. Application Example and Evaluation

We applied our method to an order transaction that included 40 items, as shown in Figure 2. As stated in Section 4, customers can have at most 192,400 combinations of unique orders. Here, we evaluated the data with at least 60,000 data recorded in the sales log. Based on the steps in Procedure 1, we perform data cleaning to remove duplicate rules. We filter rules using Cook's distance to preserve rules with strong correlations. The procedure will remove rules that exceed the threshold value such that $\mathcal{T} > 1.5$. Figure A1a,b shows the detection of irrelevant rules which are over the threshold value. Each figure shows *Price*, *Rating*, and *Calorie* as independent variables. The same procedure also applies to *Price* and *Discount*. Cook's distance is calculated by taking the independent variables from the observation. The figures show that the Cook's distance of index i that is over the red line corresponds to rule r_i will be removed. For example, Figure A1a show that rules r_6, r_7, r_{14}, and r_{15} were removed from the set of rules R. Figure A1b,c also shows the removal of rules that exceeds the threshold value.

Apriori extracted 40 associative rules. The procedure outputs 11 rules with strong correlation such as *Price-Rating* and *Calorie-Discount*. From the result, the preferences for low prices and low calories strongly correlate with ratings and discounts. By successfully identifying this factor, we can motivate the customer to decide on this menu by offering more selections with a high rating and discount. The improvement of correlation, i.e., preference p_{10} is shown in Figure 7a,b. After applying our method, the figures show the improved correlation of preference p'_{10}.

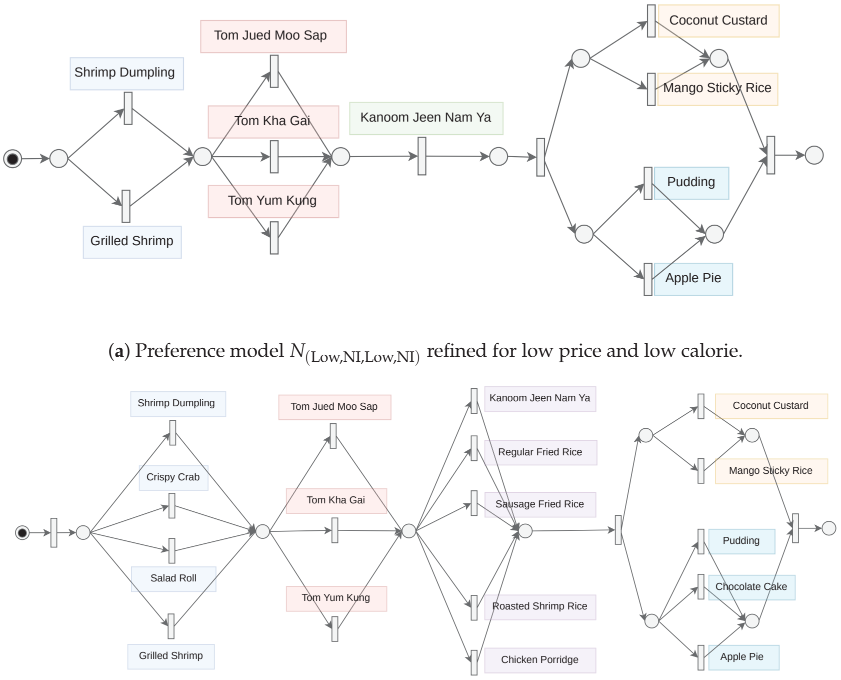


(**a**) Correlation matrix of preference P_{10}.

(**b**) Correlation matrix of preference P'_{10}.

Figure 7. Heat map of correlation matrix before and after applying Procedure 1.

The pruned and optimized preferences result is shown in Table 2. For a given preference with high correlation, 6 preferences (p_0, p_2, p_6, p_{10}, p_{14} and p_{19}) were extracted. The refined preferences is shown as p'_0, p'_2, p'_6, p'_{10}, p'_{14} and p'_{19}. The course menu selection is optimized as shown in Figure 8a,b.



(**a**) Preference model $N_{(\text{Low,NI,Low,NI})}$ refined for low price and low calorie.

(**b**) Preference model $N_{(\text{Low,High,NI,NI})}$ refined for low prices and high rating.

Figure 8. Implicit preference models output by Procedure 1.

Next, we evaluate our approach. First, we calculate the correlation coefficient rule removal based on Cook's distance. From Table A2, the value for the correlation coefficient of rules $p_0, p_2, p_6, p_{10}, p_{14}$ and p_{19} was around 0.54. Figure A1 shows the removal of rules with a distance that exceeds the threshold value for p_{10}. One independent variable is removed from each observation. Around 20% of the total rules were removed. Here, we found that preferences with *No-Interest* (NI) reveal implicit preference with the highest correlation. For example, in $p_{10} = (Low, NI, Low, NI)$, *Rating* and *Discount* was set to *No-Interest*, but $p'_{10} = (Low, NI, Low, High)$ shows the preference have strong relation to high discount.

Table 2. Explicit preference (before) and implicit preferences (after).

Old Preference (Explicit)		Refined Preference (Implicit)	
p_0	(NI, NI, NI, NI)	p'_0	(Low, NI, NI, Low)
p_2	(NI, NI, Low, NI)	p'_2	(NI, NI, Low, Low)
p_6	(NI, High, Low, NI)	p'_6	(NI, High, Low, Low)
p_{10}	(Low, NI, Low, NI)	p'_{10}	(Low, NI, Low, High)
p_{14}	(Low, High, NI, NI)	p'_{14}	(Low, High, NI, Low)
p_{19}	(Low, High, Low, NI)	p'_{19}	(Low, High, Low, Low)

The mining result shows that *Calorie-Discount* and *Price-Discount* show an increase in correlation from 0.4 to 0.7 and -0.3 to -0.5. Here, the price and calorie attributes have high correlations to discounts. Based on this information, the seller of course menu can focus on discounted items for low prices and low calories. Preference p_{10} can be refined as p'_{10} as $(Low, NI, Low, High)$. In the case of p'_{10}, low-price items and low-calorie item menus usually attract customers that prefer highly discounted menus.

Figure 9 shows the correlation coefficient $\mathcal{R}$ for *Price-Calorie* (PC), *Rating-Calorie* (RC), and *Rating-Discount* (RD). *PR′*, *PC′*, *PD′*, *RC′* and *RD′* is the correlation after applying our method. Most of the attributes' correlation increased. Here, some of the correlation changes from a negative correlation to a positive correlation. The correlation between negativity and positivity does not give much meaning other than the increase and decrease relationship of either attribute. In this paper, we focus on the strength of the correlation regardless of the negativity and positivity of the correlation.

Figure 10a shows the comparison of the correlation coefficient between attributes. The *Calorie-Discount* relation shows the most improvement in preferences P_0, P_2, P_6, and P_{10}. *Rating-Discount* and *Price-Discount* show significant improvement in preference P_{10} and P_0. This is because P_0 do not specify any interest where all attributes were set to no-interest (*N/I*) and P_{10} is almost similar to P_0 because the attribute values were set to *Low* and *No-Interest*. *Price-Calorie* shows significant improvement in preferences P_0, P_2, and P_6. In contrast, *Price-Rating* and *Rating-Calorie* correlation shows an improvement. From here, we can conclude that *Price-Rating* and *Rating-Calorie* are explicitly expressed in the preferences. Therefore, relations such as *Calorie-Discount* should be considered when recommending a new menu. Figure 10b shows the comparison for the average improvement of the coefficient for each preference. From the result, preferences with more no-interest specifications show the most improvement. This shows that our method can extract implicit preferences. However, depending on the pattern of preferences of customers, the preferences with more specifications, such as *High* and *Low*, may have different improvement values. This may be caused by the variations of selections and combinations allowed in the services, i.e., the business rules.

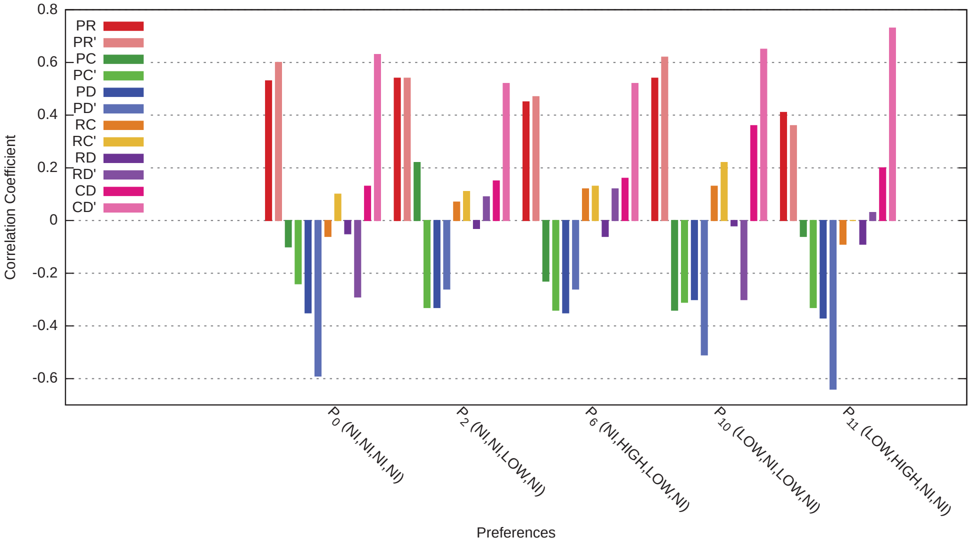
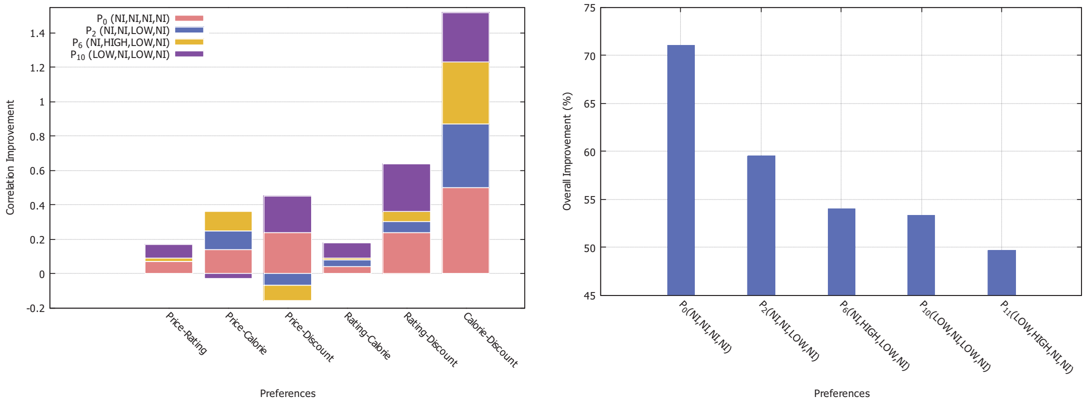


Figure 9. Comparison for the attribute's correlation coefficient for each preference.



(**a**) Improvement of correlation coefficient. (**b**) Average improvement.

Figure 10. Improvement result. Calculated based on the correlation of implicit preferences.

7. Discussion

Previous research focuses on finding explicit preferences from existing data such as product specifications, sales logs, reviews, and surveys. Explicit preference can be easily extracted from existing data mining methods such as clustering analysis, machine learning, associative mining, and genetic algorithm. The explicit preference can be gathered from the mining result. However, the gap lies when extracting implicit preference. Implicit preference in our study is regarded as an 'unknown preference' by the user. It means that the customer does not know their true preference unless some relation related to their explicit preferences are given. In this method, we use a combination of process mining and associative mining to extract implicit preferences. The ultimate benefit is that even if they do not give any explicit preference, which is 'no preference' set for all attributes of an item, the method can still extract what they might prefer by referring to the previous preference of the previous customer as a starting point in making a decision on their preferences. Intuitively, the user requires less effort but more flexibility in making decisions.

The proposed method's main characteristic is to partially extract implicit preferences where specific preferences for certain attributes related to a service or product are not the same. For example, in the given menu, course attributes such as *Price*, *Rating*, *Discount*, and *Calories* are four attributes the customer must consider. Therefore, even if they have a certain preference, such as low price and high rating, the approach can find implicit preferences that are most likely to be preferred by the user, such as high discount and low calorie. By converting numerical values in the attributes into abstract values such as category, i.e., *High* and *Low*, we can intuitively present the result to the customer. From the example, most restaurant managers generally refer to the best-selling items to improve their menu. However, the best-selling item is independent of each other, and sometimes there is no reason why such best-selling items perform better than other items. We assume that best-selling items usually is driven by other items but with less frequency. We can extract related items (mostly in sequence) from process mining to be selected by frequency. In addition, associative mining strengthens the correlation with support and lift value.

The main difference with previous works that focus on extracting explicit and implicit differences is in combination with the process mining model. Nguyen et al. [45] focused on integrating explicit and implicit ratings by using latent factors in the recommendation system. They proposed two models, which is for experienced user and inexperienced user. The method is solely based on the user's rating of each product or service item. Vu et al. [46] proposed a method to analyze and reveal implicit references using the probabilistic method. The method depends on exploratory analysis and a large group of samples to accurately extract implicit preferences.

In our process mining model, i.e., Petri net, we can explicitly represent relations between items from process mining and compare the sequence of items with associative rules. In our approach, our method emphasizes relations based on combinations of sequentially connected choices. It is optimized based on the frequency of items identified by process mining and the frequency identified by associative rule.

In extracting highly correlated associative rules, we utilized Cook's distance. The Cook's distance can be given in two versions; the L_1-norm (Manhattan distance-based) shown in Equation (3) and L_2-norm (Euclidean distance-based) version shown in Equation (5). According to Aggarwal et al. [47], the difference in L_p-norms is that due to the high dimensional data, L_1-norm is constantly preferable. The larger the value of p, the less qualitative meaning a distance metric holds. In our research, we perform multivariate data mining where the scale for each attribute in data variables differs. Therefore, Cook's distance in Manhattan-based form is preferable compared to Euclidean-based form due to sensitivity and scale in terms of the value of observation y.

We highlight the advantages of our method in extracting implicit preferences. At first, a customer intuitively set their preferences. In our method, we regard the preferences as explicit preferences. Implicit preference is a preference that the customer does not know. In general, even a service user only knows their preference once they experience using the service. This is most likely to happen to first-time users. Therefore, by extracting previous users' experiences, we can extract the options of services most likely to be selected by the first-time user. This is done by filtering our weakly correlated selections from the service processes in the form of associative rules.

As a result, we can obtain optimized preferences for the first-time user. The extracted preferences also apply to the experienced user so they can experience a better service. In the example given in this paper, a Thai restaurant's course menu was taken as an example. The result is the optimized menu course for a user that prefers p_0, p_2, p_6, p_{10}, p_{14}, and p_{19} as shown in Table 2. The method is effective for applications such as travel planning services, learning courses, and interior design or fashion coordination services.

8. Conclusions

In this paper, we proposed a method to extract implicit preference based on process mining and data mining (associative mining). The model removes a refined preference model represented by Petri net and implicit preferences with a stronger correlation than the given explicit model. The proposed procedure was able to extract various associative rules and filter the rules to implicit output preferences based on Cook's distance and Pearson correlation coefficient. The process model supports associative mining by giving confidence in filtering up highly correlated rules and representing the combination of associative rules as a process model. The model serves as multiple options for items with multi-variate attributes. The proposed approach was evaluated and showed more than 70% of improvements even if the customer did not specify any interests towards any attributes for the item selection. The method is suitable for recommending a set of options rather than a single option. It is effective when used with services that offer many variations of combinations for the customer. For example, for the problem where more choices cause harder decision-making due to various possible combinations such as travel route planning services, meal ordering services, learning courses, and interior design or fashion coordination services. In future work, we will use the proposed method to identify sentiments in selecting options in a service. The proposed method is useful for supporting user experience by extracting customized preferences.

In future work, we will use the proposed method to identify sentiments in selecting options in a service. The proposed method is useful for supporting user experience by extracting customized preferences. We plan to extract the sentiments of previous customers to support the decision-making of new customers. Even though we can find the preference of *'no preference'* set for any attributes in the recommendation, the customer should be able to understand why the preference is recommended. We plan to provide sentiment recommendations for new customers so that the reason for the recommendation can be further understood and trusted. We will also consider correlation distances such as Mahalanobis distance and cosine similarity to compare the proposed method's effectiveness.

Author Contributions: Conceptualization, M.A.B.A. and S.Y.; methodology, M.A.B.A. and S.Y.; validation, M.A.B.A.; formal analysis, M.A.B.A.; investigation, M.A.B.A.; resources, M.A.B.A.; data curation, M.A.B.A.; writing—original draft preparation, M.A.B.A.; writing—review and editing, M.A.B.A., S.Y., A.K.M. and S.S.; visualization, M.A.B.A.; supervision, S.Y.; project administration, S.Y.; funding acquisition, S.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by Interface Corporation and Faculty of Engineering, Yamaguchi University, Japan.

Data Availability Statement: The dataset used in this study is partially and openly available in Kaggle. https://www.kaggle.com/datasets/anslab/thai-restaurant-dataset, accessed on 23 November 2022.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

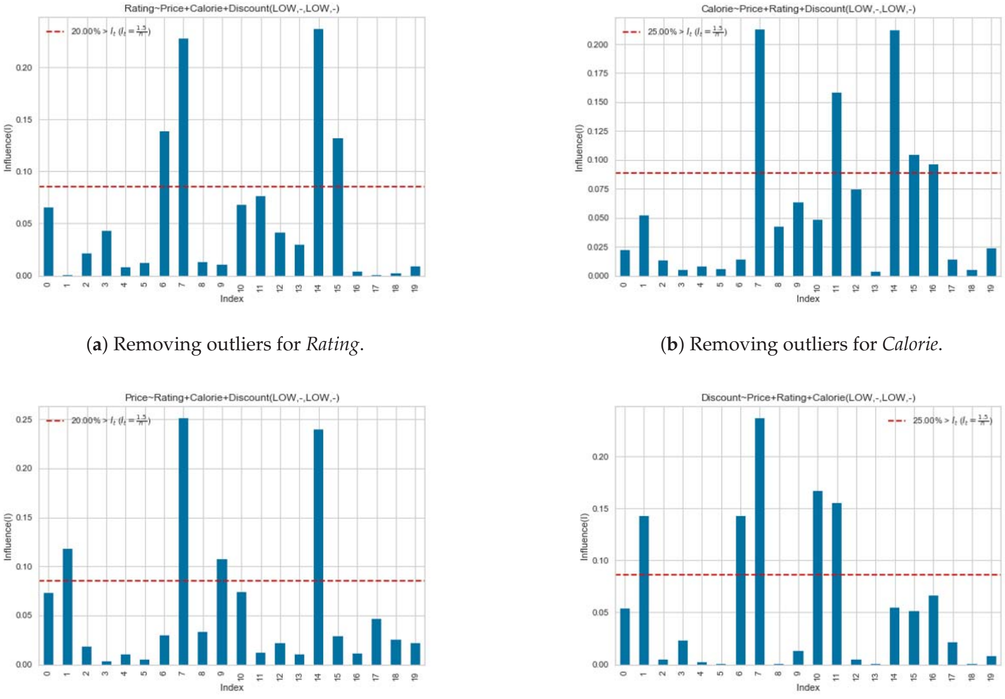
Tables and figures used in the manuscript.

Table A1. Example for rules discovered by Apriori and averaged attribute values.

Rules	Price	Rating	Calorie	Discount
r_1:('Appetizer2 = MiangKham', 90, 5, 280, 20) ⇒ ('Dessert1 = CoconutCustard', 40, 4, 540, 10)	65	4.5	410	15
r_2:('Appetizer2 = GrilledShrimp', 83, 2.7, 125, 10) ⇒ ('Dessert1 = CoconutCustard', 40, 4, 540, 10)	61.5	3.35	332.5	10
r_3:('Soup = TomYumKung', 90, 4.3, 229, 0) ⇒ ('Dessert1 = CoconutCustard', 40, 4, 540, 10)	65	4.15	384.5	5
r_4:('MainDish2 = ChickenPorridge', 50, 4.2, 228, 25) ⇒ 'Dessert1 = CoconutCustard', 40, 4, 540, 10)	45	4.1	384	17.5
r_5:('MainDish2 = KanomJeenNamYa', 45, 2, 81, 5) ⇒ ('Dessert1 = MangoStickyRice', 87, 5, 270, 0)	66	3.5	175.5	2.5
r_6:('Dessert1 = CoconutCustard', 40, 4, 540, 10) ⇒ ('Appetizer2 = MiangKham', 90, 5, 280, 20)	65	4.5	410	15

Table A2. Pattern of preference and its correlation PR, PC, PD, RC, RD, CD between attributes.

Pref.	P	R	C	D	Rules	PR	PC	PD	RC	RD	CD
p_0	**NI**	**NI**	**NI**	**NI**	**52**	**0.53**	**−0.1**	**−0.35**	**−0.06**	**−0.05**	**0.13**
p_2	**NI**	**NI**	**Low**	**NI**	**46**	**0.54**	**0.22**	**−0.33**	**0.07**	**−0.03**	**0.15**
p_4	NI	NI	High	NI	6	0.88	−0.9	−0.99	−0.58	−0.8	0.95
p_6	**NI**	**High**	**Low**	**NI**	**42**	**0.45**	**−0.23**	**−0.35**	**0.12**	**−0.06**	**0.16**
p_8	NI	High	High	NI	6	0.88	−0.9	−0.99	−0.58	−0.8	0.95
p_{10}	**Low**	**NI**	**Low**	**NI**	**40**	**0.54**	**−0.34**	**−0.3**	**0.13**	**−0.02**	**0.36**
p_{12}	Low	NI	High	NI	6	0.88	−0.9	−0.99	−0.58	−0.8	0.95
p_{14}	**Low**	**High**	**NI**	**NI**	**42**	**0.41**	**−0.06**	**−0.37**	**−0.09**	**−0.09**	**0.2**
p_{17}	Low	High	High	NI	6	0.88	−0.9	−0.99	−0.58	−0.8	0.95
p_{19}	**Low**	**High**	**Low**	**NI**	**36**	**0.45**	**−0.36**	**−0.33**	**0.2**	**−0.07**	**0.38**



(**a**) Removing outliers for *Rating*.

(**b**) Removing outliers for *Calorie*.

(**c**) Removing outliers for *Price*.

(**d**) Removing outliers for *Discount*.

Figure A1. Outlier detection based on Cook's distance. The red lines represent the cut-off threshold.

References

1. Abdi, A.; Idris, N.; Maitama, J.; Shuib, L.; Fauzi, R. A Systematic Review on Implicit and Explicit Aspect Extraction in Sentiment Analysis. *IEEE Access* **2020**, *8*, 194166–194191. [CrossRef]
2. He, G.; Li, J.; Zhao, W.X.; Liu, P.; Rong Wen, J. Mining Implicit Entity Preference from User-Item Interaction Data for Knowledge Graph Completion via Adversarial Learning. In Proceedings of The Web Conference, Taipei, Taiwan, 20–24 April 2020.
3. Van der Aalst, W. Data Science in Action. In *Process Mining: Data Science in Action*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 3–23. [CrossRef]

4. Yamaguchi, S.; Ahmadon, M.A.B.; Ge, Q.W. Introduction of Petri Nets: Its Applications and Security Challenges. In *Handbook of Research on Modern Cryptographic Solutions for Computer and Cyber Security*; IGI Global: Hershey, PA, USA, 2016; pp. 145–179. [CrossRef]
5. ProM Tools. Available online: https://www.promtools.org/ (accessed on 18 October 2022).
6. Fluxicon Disco. Available online: https://fluxicon.com/disco/ (accessed on 18 October 2022).
7. RapidProm. Available online: http://www.rapidprom.org/ (accessed on 18 October 2022).
8. Leemans, S.J.J.; Fahland, D.; van der Aalst, W.M.P. Discovering Block-Structured Process Models from Event Logs—A Constructive Approach. In Proceedings of the Application and Theory of Petri Nets and Concurrency, Milan, Italy, 24–28 June 2013; Colom, J.M., Desel, J., Eds.; Springer: Berlin/Heidelberg, Germany, 2013; pp. 311–329.
9. Ahmadon, M.A.B.; Yamaguchi, S. State Number Calculation Problem of Workflow Nets. *IEICE Trans. Inf. Syst.* **2015**, *98-D*, 1128–1136. [CrossRef]
10. Hegland, M. The apriori algorithm—A tutorial. In *Mathematics and Computation in Imaging Science and Information Processing*; World Scientific Publishing Co. Pte. Ltd.: Singapore, 2007; 209–262. [CrossRef]
11. Shayegan Fard, M.J.; Namin, P.A. Review of Apriori based Frequent Itemset Mining Solutions on Big Data. In Proceedings of the 2020 6th International Conference on Web Research (ICWR), Tehran, Iran, 22–23 April 2020; pp. 157–164. [CrossRef]
12. Kotu, V.; Deshpande, B. Chapter 6—Association Analysis. In *Data Science*, 2nd ed.; Kotu, V., Deshpande, B., Eds.; Morgan Kaufmann: Burlington, MA, USA, 2019; pp. 199–220. [CrossRef]
13. Díaz-García, J.A.; González-Farías, G. A note on the Cook's distance. *J. Stat. Plan. Inference* **2004**, *120*, 119–136. [CrossRef]
14. Szabo, F.E. *The Linear Algebra Survival Guide*; Szabo, F.E., Ed.; Academic Press: Boston, MA, USA, 2015; pp. 219–233. [CrossRef]
15. Lu, B.; Charlton, M.; Brunsdon, C.; Harris, P. The Minkowski approach for choosing the distance metric in geographically weighted regression. *Int. J. Geogr. Inf. Sci.* **2015**, *30*, 351–368. [CrossRef]
16. Chai, T.; Draxler, R.R. Root mean square error (RMSE) or mean absolute error (MAE)—Arguments against avoiding RMSE in the literature. *Geosci. Model Dev.* **2014**, *7*, 1247–1250. [CrossRef]
17. Cullinane, M.J. Metric axioms and distance. *Math. Gaz.* **2011**, *95*, 414–419. [CrossRef]
18. Zhang, J.; Lin, P.; Simeone, A. Information mining of customers preferences for product specifications determination using big sales data. *Procedia CIRP* **2022**, *109*, 101–106. [CrossRef]
19. Chong, F.S.; O'Sullivan, M.G.; Kerry, J.P.; Moloney, A.P.; Methven, L.; Gordon, A.W.; Hagan, T.D.; Farmer, L.J. Understanding consumer liking of beef using hierarchical cluster analysis and external preference mapping. *J. Sci. Food Agric.* **2020**, *100*, 245–257. [CrossRef] [PubMed]
20. Seo, Y.D.; Kim, Y.G.; Lee, E.; Kim, H. Group recommender system based on genre preference focusing on reducing the clustering cost. *Expert Syst. Appl.* **2021**, *183*, 115396. [CrossRef]
21. Osama, S.; Alfonse, M.; Salem, A.B.M. Mining Temporal Patterns to Discover Inter-Appliance Associations Using Smart Meter Data. *Big Data Cogn. Comput.* **2019**, *3*, 20. [CrossRef]
22. Wang, Y.; Zhou, J.T.; Li, X.; Song, X. Effective User Preference Clustering in Web Service Applications. *Comput. J.* **2019**, *63*, 1633–1643. [CrossRef]
23. Xiao, Y.; Li, C.; Thürer, M.; Liu, Y.; Qu, T. User preference mining based on fine-grained sentiment analysis. *J. Retail. Consum. Serv.* **2022**, *68*, 103013. [CrossRef]
24. Zheng, Q.; Ding, Q. Exploration of consumer preference based on deep learning neural network model in the immersive marketing environment. *PLoS ONE* **2022**, *17*, e0268007. [CrossRef]
25. Sun, Q.; Feng, X.; Zhao, S.; Cao, H.; Li, S.; Yao, Y. Deep Learning Based Customer Preferences Analysis in Industry 4.0 Environment. *Mob. Netw. Appl.* **2021**, *26*, 2329–2340. [CrossRef]
26. Aldayel, M.; Ykhlef, M.; Al-Nafjan, A. Deep Learning for EEG-Based Preference Classification in Neuromarketing. *Appl. Sci.* **2020**, *10*, 1525. [CrossRef]
27. Bi, K.; Qiu, T.; Huang, Y. A Deep Learning Method for Yogurt Preferences Prediction Using Sensory Attributes. *Processes* **2020**, *8*, 518. [CrossRef]
28. Gkikas, D.C.; Theodoridis, P.K.; Beligiannis, G.N. Enhanced Marketing Decision Making for Consumer Behaviour Classification Using Binary Decision Trees and a Genetic Algorithm Wrapper. *Informatics* **2022**, *9*, 45.. [CrossRef]
29. Das, S.; Nayak, J.; Nayak, S.; Dey, S. Prediction of Life Insurance Premium during Pre-and Post-Covid-19: A Higher-Order Neural Network Approach. *J. Inst. Eng. (India) Ser. B* **2022**, *103*, 1747–1773. [CrossRef]
30. Jiang, H.; Kwong, C.; Okudan Kremer, G.; Park, W.Y. Dynamic modelling of customer preferences for product design using DENFIS and opinion mining. *Adv. Eng. Inform.* **2019**, *42*, 100969. [CrossRef]
31. Petiot, J.F.; Blumenthal, D.; Poirson, E. Interactive Genetic Algorithm to Collect User Perceptions. Application to the Design of Stemmed Glasses. In *Nature-Inspired Methods for Metaheuristics Optimization*; Modeling and Optimization in Science and Technologies; Bennis, F., Bhattacharjya, R.K., Eds.; Springer: Berlin/Heidelberg, Germany, 2020; Volume 16, pp. 35–51. [CrossRef]
32. Alhijawi, B.; Kilani, Y. A collaborative filtering recommender system using genetic algorithm. *Inf. Process. Manag.* **2020**, *57*, 102310. [CrossRef]
33. Liu, C.; Kong, X.; Li, X.; Zhang, T. Collaborative Filtering Recommendation Algorithm Based on User Attributes and Item Score. *Sci. Program.* **2022**, *2022*, 4544152. [CrossRef]

34. Liang, G.; Wen, J.; Zhou, W. Individual Diversity Preference Aware Neural Collaborative Filtering. *Knowl.-Based Syst.* **2022**, *258*, 109730. [CrossRef]
35. Valera, A.; Lozano Murciego, A.; Moreno-Garcia, M.N. Context-Aware Music Recommender Systems for Groups: A Comparative Study. *Information* **2021**, *12*, 506. [CrossRef]
36. Fkih, F. Similarity measures for Collaborative Filtering-based Recommender Systems: Review and experimental comparison. *J. King Saud Univ. Comput. Inf. Sci.* **2021**, *34*, 7645–7669. [CrossRef]
37. Davis, K.M., III; Spapé, M.; Ruotsalo, T. Collaborative Filtering with Preferences Inferred from Brain Signals. In Proceedings of the Web Conference 2021, Ljubljana, Slovenia, 19–23 April 2021; Association for Computing Machinery: New York, NY, USA, 2021; pp. 602–611. [CrossRef]
38. Qi, J.; Mou, X.; Li, Y.; Chu, X.; Mu, W. A novel consumer preference mining method based on improved weclat algorithm. *J. Enterprising Communities People Places Glob. Econ.* **2022**, *16*, 74–92. [CrossRef]
39. Tan, Z.; Yu, H.; Wei, W.; Liu, J. Top-K interesting preference rules mining based on MaxClique. *Expert Syst. Appl.* **2020**, *143*, 113043. [CrossRef]
40. Chen, G.; Li, Z. A New Method Combining Pattern Prediction and Preference Prediction for Next Basket Recommendation. *Entropy* **2021**, *23*, 1430. [CrossRef]
41. Ait-Mlouk, A.; Agouti, T.; Gharnati, F. Mining and prioritization of association rules for big data: multi-criteria decision analysis approach. *J. Big Data* **2017**, *4*, 42. [CrossRef]
42. Kaur, M.; Kang, S. Market Basket Analysis: Identify the Changing Trends of Market Data Using Association Rule Mining. *Procedia Comput. Sci.* **2016**, *85*, 78–85. [CrossRef]
43. Pearson, K. Note on Regression and Inheritance in the Case of Two Parents. *Proc. R. Soc. Lond. Ser. I* **1895**, *58*, 240–242.
44. Han, J.; Pei, J.; Kamber, M. *Data Mining: Concepts and Techniques*; The Morgan Kaufmann Series in Data Management Systems; Elsevier Science: Amsterdam, The Netherlands, 2011.
45. Nguyen Hoai Nam, L. Latent factor recommendation models for integrating explicit and implicit preferences in a multi-step decision-making process. *Expert Syst. Appl.* **2021**, *174*, 114772. [CrossRef]
46. Vu, H.Q.; Li, G.; Law, R. Discovering implicit activity preferences in travel itineraries by topic modeling. *Tour. Manag.* **2019**, *75*, 435–446. [CrossRef]
47. Aggarwal, C.C.; Hinneburg, A.; Keim, D.A. On the Surprising Behavior of Distance Metrics in High Dimensional Space. In Proceedings of the Database Theory—ICDT 2001, London, UK, 4–6 January 2021; Van den Bussche, J., Vianu, V., Eds.; Springer: Berlin/Heidelberg, Germany, 2001; pp. 420–434.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

MDPI
St. Alban-Anlage 66
4052 Basel
Switzerland
Tel. +41 61 683 77 34
Fax +41 61 302 89 18
www.mdpi.com

Big Data and Cognitive Computing Editorial Office
E-mail: bdcc@mdpi.com
www.mdpi.com/journal/bdcc

MDPI
St. Alban-Anlage 66
4052 Basel
Switzerland

Tel: +41 61 683 77 34

www.mdpi.com

ISBN 978-3-0365-7211-6