MDPI

Review

Knowledge Graphs: A Practical Review of the Research Landscape

Mayank Kejriwal

Information Sciences Institute, University of Southern California, Los Angeles, CA 90007, USA; kejriwal@isi.edu

Abstract: Knowledge graphs (KGs) have rapidly emerged as an important area in AI over the last ten years. Building on a storied tradition of graphs in the AI community, a KG may be simply defined as a directed, labeled, multi-relational graph with some form of semantics. In part, this has been fueled by increased publication of structured datasets on the Web, and well-publicized successes of large-scale projects such as the Google Knowledge Graph and the Amazon Product Graph. However, another factor that is less discussed, but which has been equally instrumental in the success of KGs, is the cross-disciplinary nature of academic KG research. Arguably, because of the diversity of this research, a synthesis of how different KG research strands all tie together could serve a useful role in enabling more 'moonshot' research and large-scale collaborations. This review of the KG research landscape attempts to provide such a synthesis by first showing what the major strands of research are, and how those strands map to different communities, such as Natural Language Processing, Databases and Semantic Web. A unified framework is suggested in which to view the distinct, but overlapping, foci of KG research within these communities.

Keywords: knowledge graphs; applications; natural language processing; semantic web; data mining; knowledge representation; graph databases

1. Background and Aims

Graphs have always been important in the computational sciences, mathematics and, most notably, artificial intelligence (AI). Graph theory, as it is known today, was originally proposed by Euler in 1735 as a solution to the Königsberg Bridge Problem [1]. More recently, within AI, graphs have found use in sub-fields ranging from planning to probabilistic reasoning and inference [2,3]. In the computational social sciences [4], as well as in complex systems research in the physical sciences, networks have emerged as an important model, with many important findings over the years [5–8], including domain-specific applications in finance [9–12], crisis informatics [13,14], biology [15–17] and even human trafficking [18–21]. Of course, these models are not exclusive to the communities where they are most prominent, although the goals of each community tend to be more distinct. For example, while networks also play an important role in AI, the primary targets of research tend to be pragmatic applications, such as link prediction and community detection [22–25].

With accelerating growth of the Web over the 2000s, and the rise of both e-commerce and social media, knowledge graphs (KGs) have emerged as important models for representing, storing and querying heterogeneous pieces of data that have some relational structure between them, and that typically have real-world semantics [26]. The semantics are closely associated with the domain for which the KG has been designed [27]. A formal way to define such a domain, favored in the Semantic Web (SW) community, is through an ontology [28].

The most common definition of a KG is that it is a directed graph where both edges and nodes have labels. Nodes are considered to be entities, ranging from everyday entities such as people, organizations and locations to highly domain-specific entities such as proteins

and viruses (assuming the domain is a biological one). Edges, also known as properties or predicates, represent either relations between entities (e.g., an 'employed_at' relation between a person and organization entity) or an attribute of an entity (e.g., a person's date of birth), typically represented as a literal. Edges and nodes may also be used to represent an entity's attribute (e.g., the 'date_of_birth' of a person entity) and the attribute's value (e.g., '1970-01-01'), respectively. Even definitionally, diversity is observed in KG research. For example, the SW community makes formal distinctions between the two uses of nodes and edges mentioned above, while others, such as NLP, are less formal. (Within SW, nodes representing entities and attribute values are generally referred to as 'resources' and 'literals', respectively. Similarly, edges representing entity-relations and attributes are, respectively, referred to as 'object properties' and 'datatype properties'.)

An illustrative KG fragment from the tourism domain is visualized in Figure 1. The fragment contains both the actual KG fragment (called the A-Box) and the concepts (nodes shaded in orange) that are part of the T-Box or ontology that models the domain of interest. Put differently, concepts are the types or classes of entities allowable in the domain. Another important aspect of the domain is the set of allowable edge-labels (called properties or predicates) and the constraints associated with them. For example, the 'employed_at' relation can be constrained to only map from an entity of type 'Person' to an entity of type 'Organization'. Formally, 'Person' and 'Organization' would be declared as the allowable domain and range of the predicate 'employed_at', similar to a functional constraint in mathematics. The ontology can also have other axioms and constraints. (An intuitive example is a cardinality constraint, e.g., the requirement can be imposed that a 'married_to' predicate can be linked to at most one entity-object.) A special predicate called rdf:type serves as an explicit bridge between the A-Box and the T-Box by declaring an entity's type (which, by definition, is in the T-Box).
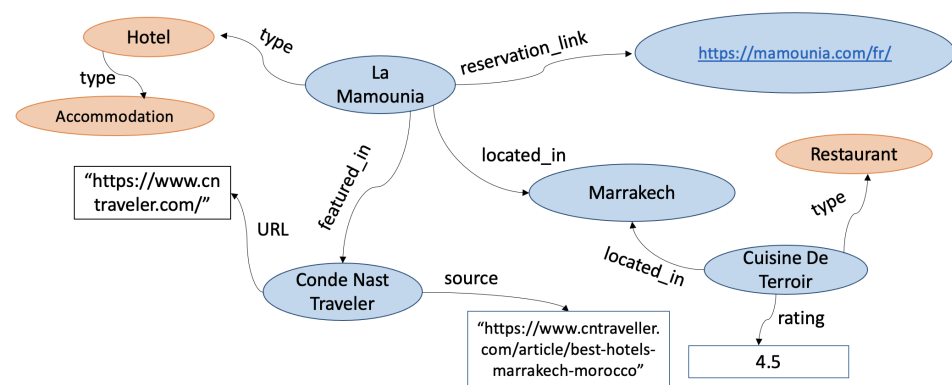

Figure 1. A knowledge graph (KG) fragment. Concepts (that typically belong in the T-Box) are shaded in orange. Links in the figure were accessed on 17 March 2022.

Per the brief formalism above, the semantics of the KG are provided for by the ontology itself, in conjunction with a reasoning engine that (in principle) can detect when the KG is violating the ontology in some way. However, while this formalism is among the most mature in the AI community for expressing, codifying and manipulating the semantics of domain knowledge, it is not the only way. The NLP, knowledge discovery and database communities have much more lightweight and implicit notions of an ontology (usually denoted a 'schema' in the academic work, if mentioned explicitly at all).

Perhaps because the surge of interest in KGs over the last decade can be traced to both the academic community (e.g., as Resource Description Framework knowledge bases in the Semantic Web) and industry (e.g., the Google Knowledge Graph), KG research has witnessed a high degree of conceptual and algorithmic diversity. This diversity can be a double-edged sword. On the one hand, it leads to a dynamic and creative research agenda where progress is continuously being achieved in some specific area that is the current focus of attention within a community. On the other hand, it may lead to the proverbial

reinventing of the wheel, including a profusion of independently developed terminology for the same fundamental phenomenon. This is already evident in at least one research sub-area (namely, entity resolution or ER) that existed before KGs in multiple research areas, including data mining [29], Semantic Web [30,31], and databases [32]. (Each of the citations is an example survey of ER in that specific area. Note that the ER problem goes by different names in different communities but as discussed in a book on the subject [33], they are fundamentally the same AI problem despite community-specific differences. Further complicating the matter is the similarity of problems such as schema matching, type matching and ontology alignment with ER [34–37].) Coincidentally, ER is an important component in any non-trivial KG construction workflow which does not help maintain uniformity [27].

The goal of this article is to provide a review of the research landscape that serves as a concise (but multi-community) synthesis of KG research as it has been shaped over the last decade. As described next, it draws on a rich body of primary sources for this synthesis. The aim herein is not to survey KGs or KG algorithms, since good efforts along those lines already exist, in addition to detailed technical surveys on KG sub-tasks such as information extraction, ER and (more recently) KG representation learning. Rather, in the spirit of similar meta-reviews in both the natural and the social sciences (two examples of which are [38,39]), this article is an attempt to survey the landscape of KG research itself, in the hopes of illustrating the connections between these (superficially disparate) research strands. Although this is not a traditional meta-review in the sense of collating data from multiple studies in a single work (usually with the goal of achieving higher statistical significance or uncovering variability that was not evident in a single study), it does collate together and synthesize output from a collection of papers and sub-fields in AI to present a more unified view of KG research than any individual paper would suggest.

2. Related Work

This synthesis builds on other related work that already reviews the technical details of KGs, and in some cases important aspects of KGs (such as information extraction), in great depth. This section primarily describes and cites other surveys that could serve as further reading material for the reader interested in learning more about these technical aspects. In the section following this one, a community-specific overview of KG research is also provided, wherein seminal research articles on some of the individual topical areas are briefly described and cited.

To begin, the term 'knowledge graph' started becoming more popular and standard in the community with the release and success of the Google Knowledge Graph [40], even though graphs have been prominent in AI since its founding days. Since the release of the Google Knowledge Graph, research on KGs has grown tremendously. A very recent and fairly comprehensive work on KG representation, acquisition and applications is the survey by [41]; while an excellent source of primary material, the survey also suggests why the current time is appropriate for a meta-review. In the survey, the authors provide overviews of state-of-the-art systems for the three topical areas mentioned above (i.e., representation, acquisition and applications), but do not attempt to unify them into a common framework. They also do not mention research in communities like the Semantic Web (SW) that have developed rigorous techniques to model and publish KGs on the Web. In contrast, the recent Communications of the Association for Computing Machinery (CACM) article by [28] takes a deeper look at how SW has contributed to KGs, but the review is primarily about the Semantic Web, which does not just research KGs. Even book-level treatments of KGs, of which there are a few (see, e.g., [42–44], and a recent textbook-level treatment [45]), have not provided an explicit mapping between KG research strands and priorities in different communities like NLP and SW. This article attempts to do so by suggesting an initial framework for thinking about these areas in a unified manner.

Earlier work that has attempted to define and provide an overview of KG models include [26,46]. In recent years, more specialized surveys have also been published,

e.g., see [47], and the author's book in 2019 on domain-specific KG construction [27], as well as Abu-Salih's survey on the same subject more recently [48]. Multilingual KGs have become popular as well, with a recent line of research detailing techniques for constructing and working with such KGs. Examples include the survey by [49] on constructing and applying Chinese KGs, the work in [50] describing how to construct a multilingual event-centric temporal KG, and the work by Wu et al. on constructing KGs from multiple non-English online encyclopedias [51]. Even more recently, a survey by Zhu et al. also discusses the construction and applications of multi-modal KGs, which tend to involve many of the challenges of building multilingual KGs [52].

An important point noted at the outset of this article is that the focus herein is on fundamental research, mainly pursued by academic researchers, as opposed to more practitioner-oriented research or applications, although an effort is made to mention and cite applications where applicable. Applications do play a very important role in the KG community, with the Google Knowledge Graph serving as an obvious example [40], but with several others having been implemented in recent years, including (most recently) in the wake of the COVID-19 pandemic [53,54]. Following the success of the Google Knowledge Graph, building and using KGs for domain-specific searches has also gained in prominence in both academia and industry [55,56]. Furthermore, as evidenced recently through efforts such as the Amazon Product Graph [55], domain-specific KGs frequently find application in industry [57–59].

3. Community-Specific Overview of KG Research

Given that different aspects of KG research are prioritized in different communities, an important component of this article is to first review the main research priorities (as pertinent to KGs) within those communities. The treatment herein does not imply exclusivity, e.g., information extraction (IE), which is predominantly researched in NLP, has also witnessed interesting research in knowledge discovery and SW [60,61]. However, an attempt is made to capture the norms and priorities of the overall community to a reasonable extent. One manner in which this attempt was made systematically was to consider the tutorials, workshops and demonstrations published in the top conferences covering these sub-fields over the last 5 years, including the International Semantic Web Conference (ISWC), the Knowledge Discovery and Data Mining (KDD) conference, the Association for Computational Linguistics (ACL), the Web Conference (WebConf; formerly known as the World Wide Web Conference) and core machine learning conferences, such as NeurIPS, International Conference on Learning Representations (ICLR) and International Conference on Machine Learning (ICML). In all of these conferences, there was at least one tutorial, and multiple workshops and demonstrations involving an important aspect of KG research. Some recent (non-exhaustive) examples of such workshops include Heterogeneous Graph Deep Learning and Applications (KDD 2021), Mining Knowledge Graph for Deep Insights (KDD 2020), International Workshop on Semantic Evaluation (ACL 2021) and Workshop on Deep Learning for Knowledge Graphs (ISWC 2021).

In short, only those communities where substantial KG-related research has been published, demonstrated or otherwise promoted (e.g., through tutorials and workshops) to date are considered. A good example of an important AI community that would not meet this condition is Computer Vision. Although some KG research has been published in Computer Vision [62], including the construction of multimodal KGs [52], the number of KG-related publications is still relatively small compared to the other communities that are covered in this section. Finally, it bears noting that, because KG research is rapidly advancing as a field, some of the areas discussed below may become less relevant for presenting advances in KG research, and others (not currently discussed in depth, such as computer vision) may gain in importance. Hence, this selection of areas should be interpreted as being only quasi-objective and subject to change even in the near future.

3.1. Natural Language Processing (NLP)

KG research can trace its origins to at least two different research areas (NLP and the Semantic Web, which is re-visited subsequently). Within NLP, KGs first emerged as a result of progress in the domain of information extraction (IE), starting from the 1990s with the institution of the Message Understanding Conferences [63]. The majority of IE research published over the last three decades has involved either named entity recognition (NER) or relation extraction (RE). Good surveys on the former include work by [64,65] (the second of which focuses on deep learning methods), while [66,67] provide a recent, comprehensive survey on the latter.

Since RE research has almost always involved 2-arity relations (where the relation is assumed to exist between a pair of entities), extracted relations and entities can be modeled as triples and placed into (what has been traditionally denoted as) a knowledge base (KB). Prior to the growth of the Web, there was no reason to model these KBs as graphs. Connections between entities became more apparent and important both when the same entity started getting extracted from multiple documents and (much later) when it was discovered that the structural properties of the KB, such as entity and relation co-occurrence features, could lead to improved performance on related tasks such as entity linking [68]. Entity linking is the problem of automatically linking an extracted entity to its equivalent in an agreed-upon 'canonical' KB like Wikipedia [69]. To take a simple example of the utility of a structural feature like co-occurrence, suppose that both 'V. Williams' and 'Wimbledon' were extracted from a single document. If the entity extraction system attempts to link these two extractions to Wikipedia independently, it becomes difficult to decide whether V. Williams refers to Venus Williams (the tennis player) or Vanessa Williams (the actress), and also whether Wimbledon refers to the tennis grand slam tournament of the same name or Wimbledon, London (where the championships are held, but which is technically different from the event itself). Co-occurrence helps resolve this ambiguity by not linking independently. More complex features help improve performance even further, and a similar philosophy would also apply to related tasks such as co-reference resolution [70], which is the problem of determining when words and phrases (including pronouns) refer to a unique entity.

From the perspective of KG research, IE, entity linking and other problems such as co-reference resolution, all play a vital role because they ultimately lead to a higher-quality initial KG. If two extractions, such as 'V. Williams' and 'Venus Williams', can indeed be linked to the same Wikipedia entry, for example, then they can be modeled as a single node in the KG. Good co-reference resolution can help add more data to the KG (e.g., more facts and relations). For these reasons, and also because of other applications that have arisen over the years (such as question answering [71]), improving performance through the design of more sophisticated algorithms and representation learning techniques has always been an important goal in the community. IE problems such as Open IE and event extraction continue to pose challenges [72,73].

3.2. Semantic Web

Earlier, the concept of the A-Box and the T-Box were briefly introduced. These notions are primarily inspired by description logics, which have heavily influenced KG research in the SW community [74]. For example, [75] describe how description logics serve as ontology languages for the semantic web. However, in the broader community, modeling and representing KGs is only one part of the equation. An equally important goal is to devise better ways of publishing, linking and accessing this data on the Web. According to a seminal article by [76], the Semantic Web is fundamentally an effort to transform the Web by 'augmenting Web pages with data targeted at computers'.

With the advent of a movement called Linked Data [77], KGs modeled in formal graph-friendly languages like Resource Description Framework (RDF) started becoming more common on the Web [78], although they are still dwarfed by the volume of natural language text. The KG fragment that was illustrated earlier in Figure 1 is an RDF graph. Data are

represented as a set of triples of the form (subject, predicate, object), intuitively representing a directed edge in the graph, where the subject and predicate must be uniform resource identifiers or URIs (and are typically just uniform resource locators for actual datasets), while the object may be a URI or a literal. (Technically, they must be internationalized resource identifiers, which subsume URIs.)

Linked Data are defined as a set of four best practices (https://www.w3.org/wiki/LinkedData accessed on 17 March 2022) for publishing 'structured data' (that are, by and large, KGs) on the Web: (i) use URIs as names for things, (ii) use HTTP URIs to enable people to look up those names, (iii) provide useful information when a person looks up a URI and (iv) include links to other URIs to enable greater discoverability [77]. Linked Open Data started in 2007 with only a handful (<10) of datasets that has since grown to hundreds of datasets in recent years [79], spanning domains as varied as social media [80,81], biology and life sciences [82,83], and computational linguistics [84,85]. The fourth principle, in particular, has made this possible, since without it, different datasets obeying the other three Linked Data principles may still have been siloed. Both classic and recent research in the 50-year-old problem of entity resolution (ER) has made automatic linking of equivalent entities in independent datasets to one another (even at the Web scale, e.g., the author's previous work on entity name systems [86]) much more feasible [29,87].

Other research priorities in SW include the development of efficient KG querying infrastructures, such as triplestores [88]. Recently, such triplestores (along with the related technology of graph databases, which has been a subject of heavy research in the core Database research community, as subsequently detailed) have also started gaining prominence, with at least one major cloud service (Amazon Neptune) available for it [89]. Another paradigm that has recently been proposed for data integration and access is the Virtual Knowledge Graph (VKG) paradigm. This paradigm is inspired by the literature on Ontology-Based Data Access (OBDA), which is a well known problem in the Semantic Web community. The key difference between VKGs and OBDA is that the former replaces rigidly structured tables that are a key feature of the latter with flexible graphs. Similar to OBDA, however, the graphs do not have to be 'materialized' but can be maintained as a virtual layer and used to capture and represent domain knowledge. A comprehensive overview of systems and use-cases for VKGs is provided in [90].

3.3. Core Machine Learning: Representation Learning and Probabilistic Graphical Models

Representation learning and probabilistic graphical models, the best known examples of which are Markov logic networks and Bayesian networks [91–94], have played an equally important role in recent KG research. Representation learning is a more recent phenomenon, with the structured embedding paper by [95], followed by influential architectures such as TransE, ConvE and the neural tensor network. Several surveys of such KG-embedding approaches have been published, examples being [96–98]. The basic purpose of these methods is to 'embed' each node and relation in the KG into a dense, continuous, real-valued vector space. Similar to word embeddings, operations such as link prediction can then be optimized in vector space. In recent years, KG representation learning and refinement have also become popular in other KG communities, such as SW [99], natural language processing [100] and broad AI topics such as commonsense reasoning [101]. More recent surveys on KG embeddings and representation learning include [97,98]. Beyond surveys, in the SW, examples of KG applications and algorithms include [102–104]. Unsurprisingly, the success of these approaches closely mirrors the success of deep learning methods and architectures in related areas. Representation learning has been particularly successful in 'refining' KGs by predicting links, detecting incorrect triples and resolving entities.

As KG embeddings have become more advanced, several authors have sought to use other classes of interesting 'information sets' with which to obtain higher-quality embeddings. One such type of information is temporal information. Since KG facts can be time-sensitive in some domains (e.g., X co-authored a paper with Y in a given year), the goal is to use time-aware embedding models to further improve KG embeddings [105,106]. One

way in which this can be accomplished is by imposing temporal order constraints on time-sensitive relation pairs. Another way is to model the temporal evolution of KGs by using quadruples rather than triples. This kind of representation is especially well suited for medical or sensor-based domains (e.g., Internet of Things). Other kinds of information sets that have inspired similar research in the machine learning community include relation paths, which are designed to help incorporate richer context into the relationship between a pair of entities [107,108], rather than 'single-hop' relations represented using an edge in the KG, and even logical rules. Although the use of such rules, once a staple of expert systems, is more common in communities such as Semantic Web, their use as regularizers when learning better KG embeddings shows the interdisciplinary connections between these fields. Examples of systems that use rules or rule-based constraints to refine KG embeddings include [109–111].

The application of probabilistic graphical models and probabilistic soft logic (PSL) to problems like link prediction predates representation learning by several years [2,112]. PSL is well suited for large-scale KGs because its optimization is convex. A particularly interesting use case is knowledge graph identification (KGI), wherein the confidence-annotated outputs of tasks like IE and ER (the 'initial' KG) are fed into a PSL program, along with ontological constraints [113]. The output of the program is a much cleaner KG. The advantage of PSL is that it is able to incorporate a combination of domain knowledge and probabilistic reasoning to 'identify' the true KG. Results have been promising. The possible synergy of such probabilistic models with representation learning is an interesting avenue for future research.

3.4. Databases, Data Mining, and Knowledge Discovery in Databases (KDD)

Although distinct from the SW or NLP communities, the knowledge discovery in databases (KDD) and data mining communities have also had a significant influence on KG research in the last 5 years. KGs have been used in innovative applications, including recommender systems [114,115]. One reason that KGs can make a difference in recommender systems' performance is their ability to provide useful external knowledge. Combined with deep learning, the external knowledge can make quite a difference. Gao et al. provide a survey on deep learning on KGs for recommender systems [116]. They cite the emergence of graph neural networks (GNNs) as an important recent advance in this space [117]. Using GNNs in tandem with KGs, recommender systems can be adapted to become more knowledge-aware, and in turn, this also helps such systems adapt to problems such as cold-start. In their survey [116], Gao et al. also cite publicly available open-source code and benchmark datasets (examples of which include [118,119]), showing that the ecosystem is starting to mature, making it more likely that these algorithms will be adopted and refined by independent developers (and possibly, smaller companies who may not have a significant research and development budget) in the near future. Although the use of external knowledge and even taxonomies is not novel in this space [120–126], KGs have historically been difficult to work with due to both scale and noise. GNNs present a robust solution to the problem [127].

KGs have also been studied under the umbrella of heterogeneous information networks or HINs [128]. The HIN model resembles a KG and it is also a directed graph, but the schema (called a network schema [129]) is less formal than the ontologies that are commonly found in the SW community. HINs have found applications in many of the domains that KGs have, including social media, healthcare and bibliographic domains [129]. To take healthcare as an example, Ding et al. [130] propose considering a biological system to be a 'complex HIN' that can be used to explore heterogeneous and complicated relationships between biological entities such as molecules to study distinct phenotypes. This treatment of HINs is reminiscent of domain-specific KGs, especially in biology and medicine (including recently proposed KGs for COVID-19) [53,82]. HINs have also been applied to recommender systems [131], as well as for tasks such as sentiment link prediction and learning structure-aware embeddings [132,133].

Last but not least, because efficient querying is an important problem in KG research [134], techniques developed by the database community, especially in query reformulation and graph databases, have also been influential [135,136]. Indeed, as argued in a synthesis lecture series on querying graphs [137], executing queries on modern graph database systems involves a 'complete lifecycle' of processing, with relevant topics of research including graph data models and query languages, graph constraints, query specification and formulation, and query processing. There are many outstanding challenges still in the community, including defining schemas for property graphs, understanding graph representations in a comprehensive and comparative framework, understanding and formalizing advanced graph query optimization techniques, and efficiently evaluating certain classes of queries. These topics are directly relevant to building, maintaining and optimizing KG access (which is fundamentally a graph querying problem), and they continue to be explored in the database community (in particular), with recently published work including [138–141].

4. A Unified Synthesis

While intuitively the different communities seem to have some overlap in their treatment of KGs, the focus is different in each. For example, the NLP community tends to work with inputs that are primarily natural language, and the results of techniques like IE, though more structured in a graph-theoretic sense, are still not 'ontologized' to the same extent as is often expected in (for example) the SW community. Hence, more algorithmic processing, such as the application of entity resolution and collective reasoning methods, is necessary. Depending on the application, the choice of a storage and querying infrastructure, which itself depends on the KG representation model such as RDF, can make the difference between success or failure in a real-world deployment.

Each community, in short, works with different inputs and outputs, and has a different focus. It is not obvious how KG research in each community relates together. Figure 2 attempts such a mapping, wherein a high-level overview is provided of the different research strands that were reviewed earlier and of the manner in which they relate among each other.
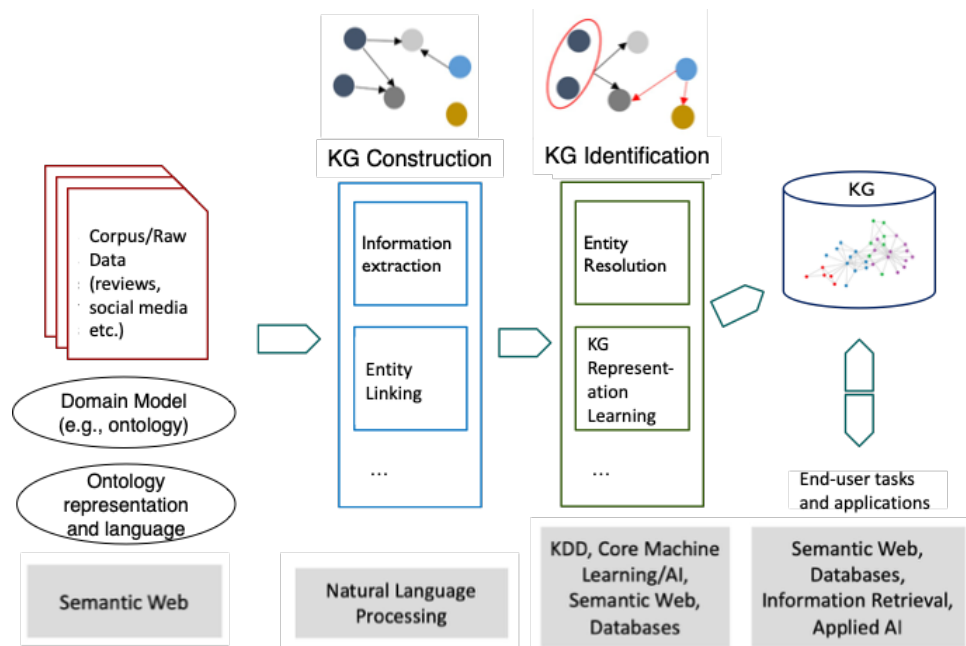

Figure 2. A systems-level overview for unifying KG research areas. Although the 'flow' of different areas is represented as a linear workflow, there are complex systems-level dependencies between many of the steps. For example, as discussed in the main text, the choice of KG access and infrastructure (the final step) is strongly influenced by the ontology representation language decided in the first step of the workflow.

Step 1 (Domain Modeling and Data Acquisition): Assuming that an ontology, even if simple, is the mechanism for modeling the domain and adding semantics to the (eventually constructed) KG, there are two important first steps: data acquisition and domain modeling. The latter has been best researched in SW, while the former may not be necessary if the data are already available. In some cases, however, it is necessary to crawl the data from the Web, or to use mechanisms like the Twitter Application Programming Interface (API) to acquire social media data. However, domain modeling is more difficult than it appears. Designing an ontology from scratch can be a complex process, requiring input from many stakeholders (often, non-technical). One option is to re-use or re-purpose standard ontologies, especially in fields for which such ontologies are well developed. Two good examples are the Conceptual Reference Model (CRM) for cultural heritage (https://cidoc-crm.org/) and the Gene Ontology (http://geneontology.org/) for biomedical domains.

If the user still decides to design the ontology from scratch, a balance has to be maintained between having concepts and relations that are too fine-grained versus a shallow schema with minimal semantics. NLP pipelines are best suited for the latter; however, the true power of a reasoning engine can only be unlocked with the former. A similar philosophy applies to the declaration and enforcement of constraints in the ontology. In general, ontology representation is known to be an important problem in the Semantic Web community, and a representation that successfully negotiates the needs of the different stakeholders mentioned above can go a long way towards avoiding problems in later steps, including the important step of knowledge graph access (discussed below) [142–144].

Step 2 (Knowledge Graph Construction): Next is a component for KG construction. Depending on whether the data are webpages, social media or natural language documents, the correct set of NLP techniques must be applied, the most important of which is IE. Other steps tend to be optional, but if extracted entities can be linked to a canonical KG like Wikidata or GeoNames, even moderately precise entity linking can raise the quality of the initial KG. (For example, co-reference resolution is almost never applicable to web documents or to social media, but if the documents are proper articles (e.g., news) then it may be essential.) Earlier, all of these steps had been described briefly: both historically and currently, they are primarily researched within NLP. Some problems, such as relation extraction, continue to lag in performance (of state-of-the-art systems on general datasets) compared to problems that have been studied for longer, such as named entity recognition [66].

Step 3 (Knowledge Graph Identification): This KG that is output by the KG construction step is fed into a component that is denoted as KG identification, in the spirit of [113]. In [113], Pujara et al. define KG identification as the task of 'removing noise, inferring missing information, and determining which candidates should be included into a knowledge graph'. They refer to the graph that is output by the KG construction step as an 'extracted graph'. Therefore, according to their terminology, the KG identification step is performed on the extracted graph, following which the actual (or 'identified') KG is obtained for purposes of querying and deploying in downstream applications.

It must be noted, however, that many communities are involved in the KG identification stage, as covered earlier. Important tasks (which tend to draw on similar techniques) include entity resolution [29,31,145], link prediction [23,132,146], triples classification [147,148] and representation learning [95,96]. To the researchers working on these problems, the provenance of the initial KG is not important; rather, their interest is in assuming it as input and yielding outputs that potentially improve it (e.g., clusters of resolved entities). As conveyed in Section 5, an important future direction is to better understand how noise in KG construction might affect these outputs and further downstream processing. Another important line of work is to reconcile the traditional KG identification methods, such as those drawing on graphical models (prominent examples of which are Bayesian networks and Markov logic networks) [91,93], with more recent advances in neural networks research, such as representation learning [99]. While some recent research has been proposed in the emerging area of neural Markov logic networks [92], successfully

applying it to KG-specific application areas such as entity resolution and link prediction is still relatively unexplored.

Step 4 (Knowledge Graph Access): Finally, to enable access, the identified KG has to be placed in a querying and storage architecture, which can range from a graph database to a more traditional triplestore (or even a document store such as Elasticsearch, which can be used to store KGs that were modeled using simple ontologies [149]). Triplestores are best suited for RDF graphs that need to be queried using a formal graph pattern-matching language such as SPARQL, while their scalability has been improving, they tend to be less mature and slower than graph-oriented databases, which are themselves less efficient or scalable than relational databases, certain 'Big Data' NoSQL databases, and document stores, such as Elasticsearch and MongoDB. Examples of commercial graph-oriented (or graph, for short) databases that have made an impact over the last decade include Neo4j and TigerGraph, along with databases proposed in academic research [136,150]. Unlike the RDF triplestores, graph databases tend to have their own representation model and querying language. For instance, Neo4j databases are queried using a language called Cypher [151].

RDF is not the only model that can be used to represent a KG. Earlier, the document store was mentioned, which takes as input key-value documents. One way to represent a KG this way is for each entity to be its own 'document' (with an identifier) and for the entity's attributes and attribute-values to be keys and values, respectively. However, ordinary graph pattern matching is not compatible with such document stores, and specialized query reformulation engines are required. On the other hand, most industrially developed graph databases can work with a variety of graph formats, including ordinary triples (that do not have URIs or other RDF elements). Many have their own querying languages. More recently, large-scale KGs like Wikidata have been represented using alternative, often simpler (but less formal), models [152].

From an engineering standpoint, decisions also have to be made whether to set up the infrastructure on a local server or to opt for a cloud service such as Amazon Neptune. Such decisions are intimately connected to the applications that the KG will be used for; while the research community has looked at many applications, the best examples of KGs in practice come from industry. Notable examples include the Google Knowledge Graph [40], the Amazon Product Graph (https://www.amazon.science/blog/building-product-graphs-automatically accessed on 17 March 2022), the LinkedIn graph (https://engineering.linkedin.com/blog/2016/10/building-the-linkedin-knowledge-graph accessed on 17 March 2022) and the eBay knowledge graph (https://tech.ebayinc.com/research/relation-embedding-with-dihedral-group-in-knowledge-graph/ accessed on 17 March 2022), to only name a few.

5. Future Directions

Fully instantiating the framework in Figure 2 for a particular domain continues to be a challenging goal, requiring considerable manpower and investment. This may be why successful examples of fully end-to-end, scalable deployments of this pipeline have only been observed in large technological organizations and research groups.

It is equally important to note that the individual components in the figure do not represent solved problems. However, in addition to the reductionist goal of improving performance of individual components (e.g., through algorithmic innovations), some problems require more systems-level thinking. An example of such a research question is: how does noise in the KG construction step affect both KG identification and querying? Part of what makes this question challenging to answer is that noise in actual information extraction algorithms is not random. By contrast, most researchers studying querying (even when trying to explicitly understand the dependencies of querying performance on noise) work with benchmarks that are relatively 'clean' compared to real-world outputs produced by the KG construction and KG identification steps. If noise is introduced, it is 'simulated' using some type of noise model (e.g., by randomly removing or corrupting facts

and entities). What is needed at present is a set of studies that attempt to understand such dependencies by using the outputs of actual execution runs from previous steps.

Another future direction is to apply KGs to commonsense reasoning problems that have recently witnessed an upsurge of attention in the AI community [153,154], perhaps due to the success of large-scale language representation models like the Bidirectional Encoder Representations from Transformers (BERT) [155]. To the best of the author's knowledge, only two major KG efforts in commonsense AI currently exist: ConceptNet and Cyc [156,157]. However, the advantages of these KGs over the new-age, large-scale language models are still not fully understood.

Visualizing KGs (especially interactively) is another area of research that could benefit from the expertise of allied research areas. Currently, there are few ways of intuitively visualizing even small KGs, and deploying visualization tools for custom KGs still requires expertise. Good examples of platforms that help with direct visualization and KG manipulation (including collaborative knowledge modeling and knowledge generation) include metaphacts (https://metaphacts.com/), but they tend to be commercial. For large KGs, popular tools that are both freely available and widely used seem to be presently lacking in the community; they represent a fruitful direction for future research, especially for system developers.

In the next half-decade, a growing body of research may start addressing some of these questions. Achieving performance improvements on individual steps continues, of course, to be a worthy goal. However, with impressive recent improvements in transformers and other neural models, the state-of-the-art is continuously being pushed for AI problems such as entity resolution, information extraction and question answering. For the last of these in particular, there have been enormous gains even since the publication and release of the original BERT architecture [158–160]. In several cases, performance is good enough that they can be transitioned to real applications. However, for a technology of similar scale and calibre as the Google Knowledge Graph to be replicated in other industries (such as manufacturing and pharmaceuticals), a more complex-systems view of Figure 2 may be necessary.

6. Conclusions

Since the Google Knowledge Graph was published, research on KGs entered a prolific era, with contributions spanning both applications and fundamental algorithms. Many advances have been achieved in tasks such as link prediction, entity resolution and search, and ecosystems that use KGs in some significant way range from large companies like Amazon and eBay to large scientific communities such as geology and biology. In this decade, applications that are even more moonshot may become feasible with more systems-level research. This review of the KG research landscape briefly summarized how KG research has developed in individual communities, and how one could view the research within a unified framework.

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. Trudeau, R.J. *Introduction to Graph Theory*; Courier Corporation: North Chelmsford, MA, USA, 2013.
2. Koller, D.; Friedman, N. *Probabilistic Graphical Models: Principles and Techniques*; MIT Press: Cambridge, MA, USA, 2009.
3. Asai, M.; Fukunaga, A. Classical planning in deep latent space: Bridging the subsymbolic-symbolic boundary. In Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018 ; Volume 32.

4. Lazer, D.M.; Pentland, A.; Watts, D.J.; Aral, S.; Athey, S.; Contractor, N.; Freelon, D.; Gonzalez-Bailon, S.; King, G.; Margetts, H.; et al. Computational social science: Obstacles and opportunities. *Science* **2020**, *369*, 1060–1062. [CrossRef]

5. Newman, M.E.; Barabási, A.L.E.; Watts, D.J. *The Structure and Dynamics of Networks*; Princeton University Press: Princeton, NJ, USA, 2006.

6. Vespignani, A. Twenty Years of Network Science. 2018. Available online: https://www.nature.com/articles/d41586-018-05444-y (accessed on 6 February 2022).

7. Kivelä, M.; Arenas, A.; Barthelemy, M.; Gleeson, J.P.; Moreno, Y.; Porter, M.A. Multilayer networks. *J. Complex Netw.* **2014**, *2*, 203–271. [CrossRef]

8. Bianconi, G. *Multilayer Networks: Structure and Function*; Oxford University Press: Oxford, UK, 2018.

9. Kejriwal, M.; Dang, A. Structural studies of the global networks exposed in the Panama papers. *Appl. Netw. Sci.* **2020**, *5*, 1–24. [CrossRef]

10. Battiston, S.; Glattfelder, J.B.; Garlaschelli, D.; Lillo, F.; Caldarelli, G. The structure of financial networks. In *Network Science*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 131–163.

11. Kejriwal, M. On using centrality to understand importance of entities in the Panama Papers. *PLoS ONE* **2021**, *16*, e0248573. [CrossRef] [PubMed]

12. Winecoff, W.K. Structural power and the global financial crisis: A network analytical approach. *Bus. Polit.* **2015**, *17*, 495–525. [CrossRef]

13. Sadri, A.M.; Ukkusuri, S.V.; Ahmed, M.A. Review of social influence in crisis communications and evacuation decision-making. *Transp. Res. Interdiscip. Perspect.* **2021**, *9*, 100325. [CrossRef]

14. Kejriwal, M.; Zhou, P. On detecting urgency in short crisis messages using minimal supervision and transfer learning. *Soc. Netw. Anal. Min.* **2020**, *10*, 58. [CrossRef]

15. Gosak, M.; Markovič, R.; Dolenšek, J.; Rupnik, M.S.; Marhl, M.; Stožer, A.; Perc, M. Network science of biological systems at different scales: A review. *Phys. Life Rev.* **2018**, *24*, 118–135. [CrossRef]

16. Koh, G.C.; Porras, P.; Aranda, B.; Hermjakob, H.; Orchard, S.E. Analyzing protein–protein interaction networks. *J. Proteome Res.* **2012**, *11*, 2014–2031. [CrossRef]

17. Szklarczyk, D.; Gable, A.L.; Nastou, K.C.; Lyon, D.; Kirsch, R.; Pyysalo, S.; Doncheva, N.T.; Legeay, M.; Fang, T.; Bork, P.; et al. The STRING database in 2021: Customizable protein–protein networks, and functional characterization of user-uploaded gene/measurement sets. *Nucleic Acids Res.* **2021**, *49*, D605–D612. [CrossRef]

18. Mendel, J.; Sharapov, K. Human trafficking and online networks: Policy, analysis, and ignorance. *Antipode* **2016**, *48*, 665–684. [CrossRef]

19. Kejriwal, M.; Kapoor, R. Network-theoretic information extraction quality assessment in the human trafficking domain. *Appl. Netw. Sci.* **2019**, *4*, 44. [CrossRef]

20. Cockbain, E. *Offender and Victim Networks in Human Trafficking*; Routledge: London, UK, 2018.

21. Kejriwal, M.; Gu, Y. Network-theoretic modeling of complex activity using UK online sex advertisements. *Appl. Netw. Sci.* **2020**, *5*, 1–23. [CrossRef]

22. Martínez, V.; Berzal, F.; Cubero, J.C. A survey of link prediction in complex networks. *ACM Comput. Surv. (CSUR)* **2016**, *49*, 1–33. [CrossRef]

23. Kumar, A.; Singh, S.S.; Singh, K.; Biswas, B. Link prediction techniques, applications, and performance: A survey. *Phys. A Stat. Mech. Its Appl.* **2020**, *553*, 124289. [CrossRef]

24. Kim, J.; Lee, J.G. Community detection in multi-layer graphs: A survey. *ACM SIGMOD Rec.* **2015**, *44*, 37–48. [CrossRef]

25. Jin, D.; Yu, Z.; Jiao, P.; Pan, S.; He, D.; Wu, J.; Yu, P.; Zhang, W. A survey of community detection approaches: From statistical modeling to deep learning. *IEEE Trans. Knowl. Data Eng.* **2021**. [CrossRef]

26. Ehrlinger, L.; Wöß, W. Towards a Definition of Knowledge Graphs. *SEMANTiCS (Posters Demos SuCCESS)* **2016**, *48*, 1–4.

27. Kejriwal, M. *Domain-Specific Knowledge Graph Construction*; Springer: Berlin/Heidelberg, Germany, 2019.

28. Hitzler, P. A review of the semantic web field. *Commun. ACM* **2021**, *64*, 76–83. [CrossRef]

29. Getoor, L.; Machanavajjhala, A. Entity resolution for big data. In Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Chicago, IL, USA, 11 August 2013 ; pp. 1527.

30. Christophides, V.; Efthymiou, V.; Palpanas, T.; Papadakis, G.; Stefanidis, K. An Overview of End-to-End Entity Resolution for Big Data. *ACM Comput. Surv. (CSUR)* **2020**, *53*, 1–42. [CrossRef]

31. Kejriwal, M. *Populating a Linked Data Entity Name System: A Big Data Solution to Unsupervised Instance Matching*; IOS Press: Amsterdam, The Netherlands, 2016; Volume 27.

32. Winkler, W.E. Matching and record linkage. *Wiley Interdiscip. Rev. Comput. Stat.* **2014**, *6*, 313–325. [CrossRef]

33. Christen, P. The data matching process. In *Data Matching*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 23–35.

34. Kolyvakis, P.; Kalousis, A.; Kiritsis, D. Deepalignment: Unsupervised ontology matching with refined word vectors. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), New Orleans, LA, USA, 1–6 June 2018; pp. 787–798.

35. Tian, A.; Kejriwal, M.; Miranker, D.P. Schema matching over relations, attributes, and data values. In Proceedings of the 26th International Conference on Scientific and Statistical Database Management, Aalborg, Denmark, 30 June–2 July 2014; pp. 1–12.

36.　Ngo, D.; Bellahsene, Z. YAM++: A multi-strategy based approach for ontology matching task. In *International Conference on Knowledge Engineering and Knowledge Management*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 421–425.

37.　Kejriwal, M.; Miranker, D.P. Experience: Type alignment on DBpedia and Freebase. *arXiv* **2016**, arXiv:1608.04442.

38.　Niedhammer, I.; Bertrais, S.; Witt, K. Psychosocial work exposures and health outcomes: A meta-review of 72 literature reviews with meta-analysis. *Scand. J. Work. Environ. Health* **2021**, *47*, 489. [CrossRef] [PubMed]

39.　Paulet, R.; Holland, P.; Morgan, D. A meta-review of 10 years of green human resource management: Is Green HRM headed towards a roadblock or a revitalisation? *Asia Pac. J. Hum. Resour.* **2021**, *59*, 159–183. [CrossRef]

40.　Singhal, A. Introducing the knowledge graph: Things, not strings. *Off. Google Blog* **2012**, *5*, 16.

41.　Ji, S.; Pan, S.; Cambria, E.; Marttinen, P.; Philip, S.Y. A survey on knowledge graphs: Representation, acquisition, and applications. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *33*, 494–514. [CrossRef] [PubMed]

42.　Fensel, D.; Şimşek, U.; Angele, K.; Huaman, E.; Kärle, E.; Panasiuk, O.; Toma, I.; Umbrich, J.; Wahler, A. *Knowledge Graphs*; Springer: Berlin/Heidelberg, Germany, 2020.

43.　Janev, V.; Graux, D.; Jabeen, H.; Sallinger, E. *Knowledge Graphs and Big Data Processing*; Springer Nature: Berlin/Heidelberg, Germany, 2020.

44.　Pan, J.Z.; Vetere, G.; Gomez-Perez, J.M.; Wu, H. *Exploiting Linked Data and Knowledge Graphs in Large Organisations*; Springer: Berlin/Heidelberg, Germany, 2017.

45.　Kejriwal, M.; Knoblock, C.A.; Szekely, P. *Knowledge Graphs: Fundamentals, Techniques, and Applications*; MIT Press: Cambridge, MA, USA, 2021.

46.　Yan, J.; Wang, C.; Cheng, W.; Gao, M.; Zhou, A. A retrospective of knowledge graphs. *Front. Comput. Sci.* **2018**, *12*, 55–74. [CrossRef]

47.　Sikos, L.F.; Philp, D. Provenance-aware knowledge representation: A survey of data models and contextualized knowledge graphs. *Data Sci. Eng.* **2020**, *5*, 293–316. [CrossRef]

48.　Abu-Salih, B. Domain-specific knowledge graphs: A survey. *J. Netw. Comput. Appl.* **2021**, *185*, 103076. [CrossRef]

49.　Wu, T.; Qi, G.; Li, C.; Wang, M. A survey of techniques for constructing Chinese knowledge graphs and their applications. *Sustainability* **2018**, *10*, 3245. [CrossRef]

50.　Gottschalk, S.; Demidova, E. EventKG: A multilingual event-centric temporal knowledge graph. In *European Semantic Web Conference*; Springer: Berlin/Heidelberg, Germany, 2018; pp. 272–287.

51.　Wu, T.; Wang, H.; Li, C.; Qi, G.; Niu, X.; Wang, M.; Li, L.; Shi, C. Knowledge graph construction from multiple online encyclopedias. *World Wide Web* **2020**, *23*, 2671–2698. [CrossRef]

52.　Zhu, X.; Li, Z.; Wang, X.; Jiang, X.; Sun, P.; Wang, X.; Xiao, Y.; Yuan, N.J. Multi-Modal Knowledge Graph Construction and Application: A Survey. *arXiv* **2022**, arXiv:2202.05786.

53.　Domingo-Fernández, D.; Baksi, S.; Schultz, B.; Gadiya, Y.; Karki, R.; Raschka, T.; Ebeling, C.; Hofmann-Apitius, M.; Kodamullil, A.T. COVID-19 Knowledge Graph: A computable, multi-modal, cause-and-effect knowledge model of COVID-19 pathophysiology. *Bioinformatics* **2021**, *37*, 1332–1334. [CrossRef] [PubMed]

54.　Kejriwal, M. Knowledge Graphs and COVID-19: Opportunities, Challenges, and Implementation. *Harv. Data Sci. Rev.* **2020**. [CrossRef]

55.　Dong, X.L. Challenges and innovations in building a product knowledge graph. In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, London, UK, 19–23 August 2018; pp. 2869.

56.　Kejriwal, M. A meta-engine for building domain-specific search engines. *Softw. Impacts* **2021**, *7*, 100052. j.simpa.2020.100052. [CrossRef]

57.　Hubauer, T.; Lamparter, S.; Haase, P.; Herzig, D.M. Use Cases of the Industrial Knowledge Graph at Siemens. In Proceedings of the International Semantic Web Conference (P&D/Industry/BlueSky), Monterey, CA, USA, 8–12 October 2018.

58.　Kejriwal, M.; Liu, Q.; Jacob, F.; Javed, F. A pipeline for extracting and deduplicating domain-specific knowledge bases. In Proceedings of the 2015 IEEE International Conference on Big Data (IEEE BigData 2015), Santa Clara, CA, USA, 29 October–1 November 2015; IEEE Computer Society: Washington, DC, USA, 2015; pp. 1144–1153. [CrossRef]

59.　Abu-Rasheed, H.; Weber, C.; Zenkert, J.; Krumm, R.; Fathi, M. Explainable Graph-Based Search for Lessons-Learned Documents in the Semiconductor Industry. In *Intelligent Computing*; Springer: Berlin/Heidelberg, Germany, 2022; pp. 1097–1106.

60.　Kejriwal, M.; Szekely, P. Information extraction in illicit web domains. In Proceedings of the 26th International Conference on World Wide Web, Perth, Australia, 3–7 April 2017; pp. 997–1006.

61.　Lin, B.Y.; Sheng, Y.; Vo, N.; Tata, S. Freedom: A transferable neural architecture for structured information extraction on web documents. In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Virtual Event, CA, USA, 6–10 July 2020; pp. 1092–1102.

62.　Zhu, Y.; Zhang, C.; Ré, C.; Fei-Fei, L. Building a large-scale multimodal knowledge base system for answering visual queries. *arXiv* **2015**, arXiv:1507.05670.

63.　Grishman, R.; Sundheim, B.M. Message understanding conference-6: A brief history. In Proceedings of the COLING 1996 Volume 1: The 16th International Conference on Computational Linguistics, Copenhagen, Denmark, 10–14 August 1996.

64.　Nadeau, D.; Sekine, S. A survey of named entity recognition and classification. *Lingvist. Investig.* **2007**, *30*, 3–26. [CrossRef]

65.　Li, J.; Sun, A.; Han, J.; Li, C. A survey on deep learning for named entity recognition. *IEEE Trans. Knowl. Data Eng.* **2020**, *34*, 50–70. [CrossRef]

66. Smirnova, A.; Cudré-Mauroux, P. Relation extraction using distant supervision: A survey. *ACM Comput. Surv. (CSUR)* **2018**, *51*, 1–35. [CrossRef]

67. Kumar, S. A survey of deep learning methods for relation extraction. *arXiv* **2017**, arXiv:1705.03645.

68. Cai, Z.; Zhao, K.; Zhu, K.Q.; Wang, H. Wikification via link co-occurrence. In Proceedings of the 22nd ACM International Conference on Information & Knowledge Management, San Francisco, CA, USA, 27 October–1 November 2013; pp. 1087–1096.

69. Shen, W.; Wang, J.; Han, J. Entity linking with a knowledge base: Issues, techniques, and solutions. *IEEE Trans. Knowl. Data Eng.* **2014**, *27*, 443–460. [CrossRef]

70. Sukthanker, R.; Poria, S.; Cambria, E.; Thirunavukarasu, R. Anaphora and coreference resolution: A review. *Inf. Fusion* **2020**, *59*, 139–162. [CrossRef]

71. Clark, J.H.; Choi, E.; Collins, M.; Garrette, D.; Kwiatkowski, T.; Nikolaev, V.; Palomaki, J. TyDi QA: A Benchmark for Information-Seeking Question Answering in Ty pologically Di verse Languages. *Trans. Assoc. Comput. Linguist.* **2020**, *8*, 454–470. [CrossRef]

72. Niklaus, C.; Cetto, M.; Freitas, A.; Handschuh, S. A survey on open information extraction. *arXiv* **2018**, arXiv:1806.05599.

73. Nguyen, T.H.; Cho, K.; Grishman, R. Joint event extraction via recurrent neural networks. In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego, CA, USA, 12–17 June 2016; pp. 300–309.

74. Nardi, D.; Brachman, R.J. An introduction to description logics. *Descr. Log. Handb.* **2003**, *1*, 40.

75. Baader, F.; Horrocks, I.; Sattler, U. Description logics as ontology languages for the semantic web. In *Mechanizing Mathematical Reasoning*; Springer: Berlin/Heidelberg, Germany, 2005; pp. 228–248.

76. Berners-Lee, T.; Hendler, J.; Lassila, O. The semantic web. *Sci. Am.* **2001**, *284*, 34–43. [CrossRef]

77. Bizer, C.; Heath, T.; Berners-Lee, T. Linked data: The story so far. In *Semantic Services, Interoperability and Web Applications: Emerging Concepts*; IGI Global: Pennsylvania, PA, USA, 2011; pp. 205–227.

78. Miller, E. An introduction to the resource description framework. *Bull. Am. Soc. Inf. Sci. Technol.* **1998**, *25*, 15–19. [CrossRef]

79. Bauer, F.; Kaltenböck, M. Linked open data: The essentials. *Ed. Mono Monochrom Vienna* **2011**, *710*. Available online: https://africa-toolkit.reeep.org/sites/default/files/LOD-the-Essentials_0.pdf (accessed on 17 March 2022).

80. Nechaev, Y.; Corcoglioniti, F.; Giuliano, C. Type prediction combining linked open data and social media. In Proceedings of the 27th ACM International Conference on Information and Knowledge Management, Torino, Italy, 22–26 October 2018; pp. 1033–1042.

81. 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]

82. Jupp, S.; Malone, J.; Bolleman, J.; Brandizi, M.; Davies, M.; Garcia, L.; Gaulton, A.; Gehant, S.; Laibe, C.; Redaschi, N.; et al. The EBI RDF platform: Linked open data for the life sciences. *Bioinformatics* **2014**, *30*, 1338–1339. [CrossRef]

83. Kamdar, M.R.; Fernández, J.D.; Polleres, A.; Tudorache, T.; Musen, M.A. Enabling web-scale data integration in biomedicine through linked open data. *NPJ Digit. Med.* **2019**, *2*, 1–14. [CrossRef] [PubMed]

84. Chiarcos, C.; McCrae, J.; Cimiano, P.; Fellbaum, C. Towards open data for linguistics: Linguistic linked data. In *New Trends of Research in Ontologies and Lexical Resources*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 7–25.

85. Bond, F.; Foster, R. Linking and extending an open multilingual wordnet. In Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Sofia, Bulgaria, 4–9 August 2013; pp. 1352–1362.

86. Kejriwal, M. Populating entity name systems for big data integration. In *International Semantic Web Conference*; Springer: Berlin/Heidelberg, Germany, 2014; pp. 521–528.

87. Kejriwal, M.; Miranker, D.P. An unsupervised instance matcher for schema-free RDF data. *J. Web Semant.* **2015**, *35*, 102–123. [CrossRef]

88. Sakr, S.; Al-Naymat, G. Relational processing of RDF queries: A survey. *ACM SIGMOD Rec.* **2010**, *38*, 23–28. [CrossRef]

89. Bebee, B.R.; Choi, D.; Gupta, A.; Gutmans, A.; Khandelwal, A.; Kiran, Y.; Mallidi, S.; McGaughy, B.; Personick, M.; Rajan, K.; et al. Amazon Neptune: Graph Data Management in the Cloud. In Proceedings of the International Semantic Web Conference (P&D/Industry/BlueSky), Monterey, CA, USA, 8–12 October 2018.

90. Xiao, G.; Ding, L.; Cogrel, B.; Calvanese, D. Virtual knowledge graphs: An overview of systems and use cases. *Data Intell.* **2019**, *1*, 201–223. [CrossRef]

91. Richardson, M.; Domingos, P. Markov logic networks. *Mach. Learn.* **2006**, *62*, 107–136. [CrossRef]

92. Marra, G.; Kuželka, O. Neural markov logic networks. In Proceedings of the Uncertainty in Artificial Intelligence, Online, 27–30 July 2021; pp. 908–917.

93. Pearl, J. Bayesian Networks. 2011. Available online: https://ftp.cs.ucla.edu/pub/stat_ser/r277.pdf (accessed on 17 March 2022).

94. Koski, T.; Noble, J. *Bayesian Networks: An Introduction*; John Wiley & Sons: Hoboken, NJ, USA, 2011; Volume 924.

95. Bordes, A.; Weston, J.; Collobert, R.; Bengio, Y. Learning structured embeddings of knowledge bases. In Proceedings of the AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 7–11 August 2011; Volume 25.

96. Wang, Q.; Mao, Z.; Wang, B.; Guo, L. Knowledge graph embedding: A survey of approaches and applications. *IEEE Trans. Knowl. Data Eng.* **2017**, *29*, 2724–2743. [CrossRef]

97. Choudhary, S.; Luthra, T.; Mittal, A.; Singh, R. A survey of knowledge graph embedding and their applications. *arXiv* **2021**, arXiv:2107.07842.

98.　Dai, Y.; Wang, S.; Xiong, N.N.; Guo, W. A survey on knowledge graph embedding: Approaches, applications and benchmarks. *Electronics* **2020**, *9*, 750. [CrossRef]

99.　Paulheim, H. Knowledge graph refinement: A survey of approaches and evaluation methods. *Semant. Web* **2017**, *8*, 489–508. [CrossRef]

100.　Huang, X.; Zhang, J.; Li, D.; Li, P. Knowledge graph embedding based question answering. In Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining, Melbourne, Australia, 11–15 February 2019; pp. 105–113.

101.　Guan, N.; Song, D.; Liao, L. Knowledge graph embedding with concepts. *Knowl.-Based Syst.* **2019**, *164*, 38–44. [CrossRef]

102.　Kejriwal, M.; Szekely, P. Neural embeddings for populated geonames locations. In *International Semantic Web Conference*; Springer: Berlin/Heidelberg, Germany, 2017; pp. 139–146.

103.　Ristoski, P.; Paulheim, H. Rdf2vec: Rdf graph embeddings for data mining. In *International Semantic Web Conference*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 498–514.

104.　Nayyeri, M.; Vahdati, S.; Zhou, X.; Shariat Yazdi, H.; Lehmann, J. Embedding-based recommendations on scholarly knowledge graphs. In *European Semantic Web Conference*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 255–270.

105.　Liu, Y.; Hua, W.; Xin, K.; Zhou, X. Context-aware temporal knowledge graph embedding. In *International Conference on Web Information Systems Engineering*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 583–598.

106.　Xu, C.; Nayyeri, M.; Alkhoury, F.; Yazdi, H.S.; Lehmann, J. TeRo: A time-aware knowledge graph embedding via temporal rotation. *arXiv* **2020**, arXiv:2010.01029.

107.　Jia, N.; Cheng, X.; Su, S. Improving knowledge graph embedding using locally and globally attentive relation paths. In *European Conference on Information Retrieval*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 17–32.

108.　Xiong, S.; Huang, W.; Duan, P. Knowledge graph embedding via relation paths and dynamic mapping matrix. In *International Conference on Conceptual Modeling*; Springer: Berlin/Heidelberg, Germany, 2018; pp. 106–118.

109.　Guo, S.; Wang, Q.; Wang, L.; Wang, B.; Guo, L. Jointly embedding knowledge graphs and logical rules. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Austin, TX, USA, 1–5 November 2016; pp. 192–202.

110.　Wang, P.; Dou, D.; Wu, F.; de Silva, N.; Jin, L. Logic rules powered knowledge graph embedding. *arXiv* **2019**, arXiv:1903.03772.

111.　Guo, S.; Wang, Q.; Wang, L.; Wang, B.; Guo, L. Knowledge graph embedding with iterative guidance from soft rules. In Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018; Volume 32.

112.　Kimmig, A.; Bach, S.; Broecheler, M.; Huang, B.; Getoor, L. A short introduction to probabilistic soft logic. In Proceedings of the NIPS Workshop on Probabilistic Programming: Foundations and Applications, Lake Tahoe, NV, USA, 3–6 December 2012; pp. 1–4.

113.　Pujara, J.; Miao, H.; Getoor, L.; Cohen, W. Knowledge graph identification. In *International Semantic Web Conference*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 542–557.

114.　Wang, H.; Zhang, F.; Zhang, M.; Leskovec, J.; Zhao, M.; Li, W.; Wang, Z. Knowledge-aware graph neural networks with label smoothness regularization for recommender systems. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Anchorage, AK, USA, 4–8 August 2019; pp. 968–977.

115.　Palumbo, E.; Rizzo, G.; Troncy, R. Entity2rec: Learning user-item relatedness from knowledge graphs for top-n item recommendation. In Proceedings of the Eleventh ACM Conference on Recommender Systems, Como, Italy, 27–31 August 2017; pp. 32–36.

116.　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.

117.　Yin, R.; Li, K.; Zhang, G.; Lu, J. A deeper graph neural network for recommender systems. *Knowl.-Based Syst.* **2019**, *185*, 105020. [CrossRef]

118.　Zhang, J.; Shi, X.; Zhao, S.; King, I. Star-gcn: Stacked and reconstructed graph convolutional networks for recommender systems. *arXiv* **2019**, arXiv:1905.13129.

119.　Song, W.; Xiao, Z.; Wang, Y.; Charlin, L.; Zhang, M.; Tang, J. Session-based social recommendation via dynamic graph attention networks. In Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining, Melbourne, Australia, 11–15 February 2019; pp. 555–563.

120.　Friedrich, G.; Zanker, M. A taxonomy for generating explanations in recommender systems. *AI Mag.* **2011**, *32*, 90–98. [CrossRef]

121.　Kejriwal, M.; Shen, K.; Ni, C.C.; Torzec, N. Transfer-based taxonomy induction over concept labels. *Eng. Appl. Artif. Intell.* **2022**, *108*, 104548. [CrossRef]

122.　Kejriwal, M.; Shen, K.; Ni, C.C.; Torzec, N. An evaluation and annotation methodology for product category matching in e-commerce. *Comput. Ind.* **2021**, *131*, 103497. [CrossRef]

123.　Kanagal, B.; Ahmed, A.; Pandey, S.; Josifovski, V.; Yuan, J.; Garcia-Pueyo, L. Supercharging recommender systems using taxonomies for learning user purchase behavior. *arXiv* **2012**, arXiv:1207.0136.

124.　Kejriwal, M.; Selvam, R.K.; Ni, C.C.; Torzec, N. Locally constructing product taxonomies from scratch using representation learning. In Proceedings of the 2020 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), The Hague, The Netherlands, 7–10 December 2020; pp. 507–514.

125.　Kejriwal, M.; Shen, K. Unsupervised real-time induction and interactive visualization of taxonomies over domain-specific concepts. In Proceedings of the 2021 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, Virtual, The Netherlands, 8–11 November 2021; pp. 301–304.

126. Liang, H.; Xu, Y.; Li, Y.; Nayak, R.; Weng, L.T. Personalized recommender systems integrating social tags and item taxonomy. In Proceedings of the 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology, Milan, Italy, 15–18 September 2009; Volume 1; pp. 540–547.

127. Wu, Z.; Pan, S.; Chen, F.; Long, G.; Zhang, C.; Philip, S.Y. A comprehensive survey on graph neural networks. *IEEE Trans. Neural Netw. Learn. Syst.* **2020**, *32*, 4–24. [CrossRef]

128. Shi, C.; Li, Y.; Zhang, J.; Sun, Y.; Philip, S.Y. A survey of heterogeneous information network analysis. *IEEE Trans. Knowl. Data Eng.* **2016**, *29*, 17–37. [CrossRef]

129. Sun, Y.; Han, J. Mining heterogeneous information networks: Principles and methodologies. *Synth. Lect. Data Min. Knowl. Discov.* **2012**, *3*, 1–159. [CrossRef]

130. Ding, P.; Ouyang, W.; Luo, J.; Kwoh, C.K. Heterogeneous information network and its application to human health and disease. *Brief. Bioinform.* **2020**, *21*, 1327–1346. [CrossRef] [PubMed]

131. Shi, C.; Hu, B.; Zhao, W.X.; Philip, S.Y. Heterogeneous information network embedding for recommendation. *IEEE Trans. Knowl. Data Eng.* **2018**, *31*, 357–370. [CrossRef]

132. Wang, H.; Zhang, F.; Hou, M.; Xie, X.; Guo, M.; Liu, Q. Shine: Signed heterogeneous information network embedding for sentiment link prediction. In Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining, Marina Del Rey, CA, USA, 5–9 February 2018; pp. 592–600.

133. Lu, Y.; Shi, C.; Hu, L.; Liu, Z. Relation structure-aware heterogeneous information network embedding. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; Volume 33; pp. 4456–4463.

134. Goasdoué, F.; Manolescu, I.; Roatiş, A. Efficient query answering against dynamic RDF databases. In Proceedings of the 16th International Conference on Extending Database Technology, Genoa, Italy, 18–22 March 2013; pp. 299–310.

135. Tatarinov, I.; Halevy, A. Efficient query reformulation in peer data management systems. In Proceedings of the 2004 ACM SIGMOD International Conference on Management of Data, Paris, France, 13–18 June 2004; pp. 539–550.

136. Angles, R.; Gutierrez, C. Survey of graph database models. *ACM Comput. Surv. (CSUR)* **2008**, *40*, 1–39. [CrossRef]

137. Bonifati, A.; Fletcher, G.; Voigt, H.; Yakovets, N. Querying graphs. *Synth. Lect. Data Manag.* **2018**, *10*, 1–184. [CrossRef]

138. Sakr, S.; Bonifati, A.; Voigt, H.; Iosup, A.; Ammar, K.; Angles, R.; Aref, W.; Arenas, M.; Besta, M.; Boncz, P.A.; et al. The future is big graphs: A community view on graph processing systems. *Commun. ACM* **2021**, *64*, 62–71. [CrossRef]

139. Angles, R.; Bonifati, A.; Dumbrava, S.; Fletcher, G.; Hare, K.W.; Hidders, J.; Lee, V.E.; Li, B.; Libkin, L.; Martens, W.; et al. Pg-keys: Keys for property graphs. In Proceedings of the 2021 International Conference on Management of Data, Virtual Event, Xi'an, China, 20–25 June 2021; pp. 2423–2436.

140. Klijn, E.L.; Mannhardt, F.; Fahland, D. Classifying and Detecting Task Executions and Routines in Processes Using Event Graphs. In *International Conference on Business Process Management*; Springer: Berlin/Heidelberg, Germany, 2021; pp. 212–229.

141. Lbath, H.; Bonifati, A.; Harmer, R. Schema inference for property graphs. In Proceedings of the EDBT 2021-24th International Conference on Extending Database Technology, Nicosia, Cyprus, 23–26 March 2021; pp. 499–504.

142. Hoekstra, R. Ontology Representation: Design Patterns and Ontologies That Make Sense. 2009. Available online: https://pure.uva.nl/ws/files/763154/68623_thesis.pdf (accessed on 6 February 2022).

143. Guizzardi, G. The role of foundational ontologies for conceptual modeling and domain ontology representation. In Proceedings of the 2006 7th International Baltic Conference on Databases and Information Systems, Vilnius, Lithuania, 3–6 July 2006; pp. 17–25.

144. Lenzerini, M.; Milano, D.; Poggi, A. *Ontology Representation & Reasoning*; Universita di Roma La Sapienza: Roma, Italy, 2004. Available online: http://wwwusers.di.uniroma1.it/~estrinfo/1%20Ontology%20representation.pdf (accessed on 6 February 2022).

145. Kejriwal, M. Unsupervised DNF Blocking for Efficient Linking of Knowledge Graphs and Tables. *Information* **2021**, *12*, 134. [CrossRef]

146. Kejriwal, M. Link Prediction Between Structured Geopolitical Events: Models and Experiments. *Front. Big Data* **2021**, *4*, 779792. [CrossRef]

147. Lin, Y.; Liu, Z.; Sun, M.; Liu, Y.; Zhu, X. Learning entity and relation embeddings for knowledge graph completion. In Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, Austin, TX, USA, 25–30 January 2015.

148. Feng, J.; Huang, M.; Wang, M.; Zhou, M.; Hao, Y.; Zhu, X. Knowledge graph embedding by flexible translation. In Proceedings of the Fifteenth International Conference on the Principles of Knowledge Representation and Reasoning, Cape Town, South Africa, 25–29 April 2016.

149. Kejriwal, M.; Szekely, P. Knowledge graphs for social good: An entity-centric search engine for the human trafficking domain. *IEEE Trans. Big Data* **2017**. [CrossRef]

150. Bonifati, A.; Furniss, P.; Green, A.; Harmer, R.; Oshurko, E.; Voigt, H. Schema validation and evolution for graph databases. In *International Conference on Conceptual Modeling*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 448–456.

151. Francis, N.; Green, A.; Guagliardo, P.; Libkin, L.; Lindaaker, T.; Marsault, V.; Plantikow, S.; Rydberg, M.; Selmer, P.; Taylor, A. Cypher: An evolving query language for property graphs. In Proceedings of the 2018 International Conference on Management of Data, Houston, TX, USA, 10–15 June 2018; pp. 1433–1445.

152. Vrandečić, D.; Krötzsch, M. Wikidata: A free collaborative knowledgebase. *Commun. ACM* **2014**, *57*, 78–85. [CrossRef]

153. Zellers, R.; Bisk, Y.; Farhadi, A.; Choi, Y. From recognition to cognition: Visual commonsense reasoning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 6720–6731.

154. Lin, B.Y.; Chen, X.; Chen, J.; Ren, X. Kagnet: Knowledge-aware graph networks for commonsense reasoning. *arXiv* **2019**, arXiv:1909.02151.

155. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv* **2018**, arXiv:1810.04805.

156. Liu, H.; Singh, P. ConceptNet—a practical commonsense reasoning tool-kit. *BT Technol. J.* **2004**, *22*, 211–226. [CrossRef]

157. Lenat, D.B. CYC: A large-scale investment in knowledge infrastructure. *Commun. ACM* **1995**, *38*, 33–38. [CrossRef]

158. Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language models are few-shot learners. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 1877–1901.

159. Petroni, F.; Rocktäschel, T.; Lewis, P.; Bakhtin, A.; Wu, Y.; Miller, A.H.; Riedel, S. Language models as knowledge bases? *arXiv* **2019**, arXiv:1909.01066.

160. Yang, Z.; Garcia, N.; Chu, C.; Otani, M.; Nakashima, Y.; Takemura, H. A comparative study of language transformers for video question answering. *Neurocomputing* **2021**, *445*, 121–133. [CrossRef]