



# **A Systematic Study on a Customer's Next-Items Recommendation Techniques**

Qazi Mudassar Ilyas <sup>1,\*</sup>, Abid Mehmood <sup>2</sup>, Ashfaq Ahmad <sup>3</sup> and Muneer Ahmad <sup>4,\*</sup>

- <sup>1</sup> Department of Information Systems, College of Computer Sciences and Information Technology, King Faisal University, Al Ahsa 31982, Saudi Arabia
- <sup>2</sup> Department of Management Information Systems, College of Business Administration, King Faisal University, Al Ahsa 31982, Saudi Arabia; aafzal@kfu.edu.sa
- <sup>3</sup> Department of Computer Science, College of Computer Sciences & Information Technology, Jazan University, Jazan 45142, Saudi Arabia; ashfaq@jazanu.edu.sa
- <sup>4</sup> School of Electrical Engineering and Computer Science (SEECS), National University of Sciences and Technology (NUST), Sector H-12, Islamabad 44000, Pakistan
- \* Correspondence: qilyas@kfu.edu.sa (Q.M.I.); muneer.ahmad@seecs.edu.pk (M.A.)

Abstract: A customer's next-items recommender system (NIRS) can be used to predict the purchase list of a customer in the next visit. The recommendations made by these systems support businesses by increasing their revenue and providing a more personalized shopping experience to customers. The main objective of this paper is to provide a systematic literature review of the domain to analyze the recent techniques and assist future research. The paper examined 90 selected studies to answer the research questions concerning the key aspects of NIRSs. To this end, the main contribution of the paper is that it provides detailed insight into the use of conventional and deep learning techniques, the popular datasets, and specialized metrics for developing and evaluating these systems. The study reveals that conventional machine learning techniques have been quite popular for developing NIRSs in the past. However, more recent works have mainly focused on deep learning techniques in developing NIRSs that need further investigation are related to cold start, data sparsity, and cross-domain recommendations.

**Keywords:** recommender systems; next-items; next basket; personalization; systematic literature review; deep learning

# 1. Introduction

A point-of-sale (POS) system is considered an integral part of a store. A typical POS system supports several core business functions, including inventory control, invoicing, sales management, payment processing, and managerial reporting [1]. A POS system in a typical brick-and-mortar store usually stores the sales data anonymously. However, the stores also use loyalty programs to record individual customers' purchase history.

The recent outbreak of COVID-19 has motivated customers to adopt online shopping modes to limit physical contact and reduce the chances of contracting coronavirus in physical stores [2]. In some cases, such as home isolation, customers were left with no option but to shop online. Online shopping has great potential to collect even more data than brick-and-mortar stores [3]. As online shopping is usually done by logging into the store system, the store can also collect other valuable data such as customer browsing history, interests, and frequently purchased items [4].

Data are considered to be an essential asset by businesses today. It can be used to gain a competitive advantage [5]. A recommender system is one popular application of exploiting sales data to enhance sales [6]. A recommender system can be used for cross-selling, upselling, and down-selling. Businesses commonly use these techniques to enhance sales, build customer loyalty, and increase revenue [7].



Citation: Ilyas, Q.M.; Mehmood, A.; Ahmad, A.; Ahmad, M. A Systematic Study on a Customer's Next-Items Recommendation Techniques. *Sustainability* **2022**, *14*, 7175. https:// doi.org/10.3390/su14127175

Academic Editor: Pedro Guimarães

Received: 10 May 2022 Accepted: 9 June 2022 Published: 11 June 2022

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



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https:// creativecommons.org/licenses/by/ 4.0/). The next-items (or next basket) prediction is a recommender system that can be built using data collected by a POS system. The system predicts the items most likely to be purchased by a customer in the next visit based on the past purchase history. Formally, the next-basket recommendation problem can be stated as follows [8–10]:

Let  $C = \{c_1, c_2, c_3, \dots, c_m\}$  be a set of distinct customers, and  $I = \{i_1, i_2, i_3, \dots, i_n\}$  be a set of items/products. Let  $B_t^u = \{i_1^c, i_2^c, i_3^c, \dots, i_p^c\}$  be a shopping basket at time t and every  $ix_t^c \in I$  is an item purchased by customer c in this basket. The set of all baskets of a customer c sorted by time is represented as  $\mathbb{B}^c = \{B_{t_1}^c, B_{t_2}^c, B_{t_3}^c, \dots, B_{t_k}^c\}$ . The next-items recommendation task attempts to predict  $B_{t_{k+1}}^c = \{i_{k+1}^c, i_{k+1}^c, i_{k+1}^c, \dots, i_{k+1}^c\}$  through a prediction function that recommends top l items in descending order of their probabilities to be purchased by the customer c at time k + 1.

A next-items recommender system (NIRS) can serve businesses and customers in several ways. These recommendations can be used to support targeted marketing and advertisement. One-size-fits-all is not applicable in most businesses today. Customized and user-centric services have become mainstream in the last decade. Such a personalized shopping experience is much more enjoyable for a customer, increasing customer loyalty and revenue. These recommendations also help reduce the possibility of lost revenue because of forgetful customers, as such customers may buy these products from a competitor, affecting future sales. Personalized recommendations greatly enhance customers' shopping experience, resulting in customer satisfaction. Improved customer satisfaction improves customer loyalty, ultimately resulting in an improved reputation and increased revenue for the business. These recommender systems can also be used for cross-selling and up-selling. The most recent application of such recommender systems is to dynamically generate online brochures and websites highlighting recommendations for every customer.

Several techniques have been proposed to develop NIRSs. The main purpose of this study is to provide a thorough and unbiased review of these techniques to help researchers in proposing improved solutions. In order to achieve this goal, we adopted the systematic literature review (SLR) technique proposed by Kitchenham et al. [11]. SLRs provide a framework for defining the scope and objectives of a study, selecting relevant resources, conducting the review, and reporting its results. To the best of our knowledge, NIRSs have not been reviewed using the aforementioned approach. Following the SLR framework, the current study defines a number of research questions related to NIRS and investigates the existing literature to answer them. The main contribution of the paper is to investigate the key aspects of the NIRSs, including the use of conventional and deep learning techniques, the popular datasets, and the specialized metrics for developing and evaluating these systems.

The rest of the paper is organized as follows. Section 2 gives an overview of the related works. Section 3 presents the methodology used for conducting this study. Sections 4 and 5 explore conventional and deep learning techniques for a customer's next-items recommendations, respectively. Section 6 presents commonly used datasets, while Section 7 introduces evaluation measures for next-items recommender systems. Finally, Section 8 highlights challenges and future directions for such systems, and the paper is concluded in Section 9.

# 2. Related Work

As revealed by the results of an exhaustive search of the selected digital sources, no previous SLR has been reported on NIRSs. However, since NIRSs can be viewed as a specialized case of recommender systems, this section describes the SLRs related to the latter type of systems. Table 1 gives a summary of these works.

Study	Focus	Publication Year	Studies Reviewed	Years Covered
Alyari and Jafari [12]	Recommender systems	2018	51	2005-2018
Portugal et al. [13]	Recommender systems	2018	121	2001–2016
Monti et al. [14]	Multicriteria recommender systems	2021	93	2003–2018
Villegas et al. [15]	Context-aware recommenders	2017	87	2004–2016
Murciego et al. [16]	Context-aware recommender systems in the music domain	2021	100	2010–2021
Jesse and Jannach [17]	Digital nudging in recommender systems	2021	63	2009–2021
Da'u and Salim [18]	Deep learning methods in recommender systems	2019	99	2007–2018
Khan et al. [19]	Recommender systems for e-tourism	2021	143	2012-2020
Hamid et al. [20]	Recommender systems for e-tourism	2021	65	2013–2020
Mohammadi et al. [21]	Trust-based recommender systems in Internet of Things	2019	59	2011–2018
Rahayu et al. [22]	Use of ontologies in recommender systems	2022	28	2010-2020

Table 1. Summary of related works.

Alyari and Jafari Navimipour [12] provide a comprehensive review of recommender systems. They used 51 studies and covered various types of recommender systems such as collaborative filtering, content-based filtering, demographic filtering, knowledge-based, and hybrid systems. The study's main limitation is a lack of coverage of the latest trends in deep learning for developing recommender systems. Portugal et al. [13] also conducted an SLR about recommendation systems. The study was also published in 2018 and reviewed 121 papers published from 2001 to 2016. Monti et al. [14] conducted an exhaustive review of multicriteria recommender systems using 93 studies published between 2003 and 2018. A brief introduction to the primary recommendation approaches is followed by an in-depth analysis of multicriteria recommender systems, their applications, evaluation protocols, metrics, and datasets. The authors suggest integrating user-based, item-based, and contextaware approaches based on item characteristics for future studies. Villegas et al. [15] surveyed context-aware recommenders and provided a reference framework characterizing the recommendation processes in recommendation techniques, methods of incorporating context, and the process stages at which integration of context into the systems would be reasonable. The study included 87 papers and classified them as context-based, collaborative filtering, and hybrid approaches. Murciego et al. [16] also studied contextual recommender systems. However, the focus of their study was the music domain.

Jesse and Jannach establish an analogy between automated recommendations and nudging, as recommender systems subtly influence users' decision-making process without forcing a choice [17]. They argue that only a few types of nudges have been explored by other researchers and present a comprehensive taxonomy of 87 mechanisms of digital nudging. The authors propose investigating other forms of nudging, such as developing a feeling of urgency or scarcity in users for quicker decision-making. Personalized nudging is another area that needs further exploration and research. Da'u and Salim [18] provide an exposé of deep-learning-based recommender systems. Using 99 publications from 2007 to 2018 in credible sources, they reviewed several deep learning techniques for automated recommendations. The authors conclude that autoencoder models are the most popular technique for developing recommender systems. CNN and RNN are also widely used by researchers for developing such systems. The review also highlights datasets and evaluation measures used in these systems. The authors highlight the data extraction method as the study's main limitation and argue that some critical attributes might have been ignored from the study's scope because of a limited number of sources considered in the study. Khan et al. [19] explored the use of recommender systems in the e-tourism domain through 143 articles published from 2012 to 2020. In particular, the study focuses on tourists' preferences and location-based contextual recommendations in e-tourism. The authors argue that automated recommendations not only help offer personalized products to the tourists but also support sustainable tourism by reducing travel costs and time. The authors identify diversity in tourism services and personalized and realtime recommendations for travelers as future research directions. Hamid et al. [20] also performed an SLR with a similar scope and objectives. Mohammadi et al. [21] performed an SLR on trust-based recommender systems in the Internet of Things environments. The authors reviewed 59 papers between 2012 and 2018 and highlighted challenges in the trust-based recommendation for various architectures such as cloud computing, fog computing, peer-to-peer, and service-oriented architecture. Rahayu et al. [22] studied the usage of ontologies in developing e-learning recommender systems. The primary focus of the study is learning objects. However, the authors have also reviewed the role of ontologies in assessment and feedback to learners. The authors report a lack of awareness among stakeholders in adopting ontologies in e-learning recommender systems. They also highlight the lack of a systematic approach for ontology evaluation in these systems, which raises questions about the effectiveness of such systems. Although the study considered high-quality papers in their review, searches were performed only in computer science and artificial intelligence domains. A more thorough search strategy, including sources in engineering and multidisciplinary venues, could have resulted in a more thorough review. This argument is also supported by a relatively fewer number of articles selected for the study (28). The study also did not consider some critical factors in the e-learning environment that affect recommender systems, such as devices, pedagogical theories, and technology readiness in students and instructors.

# 3. Methodology

In this study, we have followed the SLR methodology for software engineering proposed by Kitchenham et al. [11] and Wohlin et al. [23]. Figure 1 gives a pictorial representation of the phases and activities in each phase. In the following, we give details of each phase and associated activities as they were performed for this study.

Planning the review Identification of the need Specifying the research questions Developing review protocol

Evaluating review protocol

**Conducting the review** Identification of research Selection of primary studies Study quality assessment Data extraction & synthesis **Reporting the review** Specifying dissemination mechanisms Developing review report Evaluating review report

Figure 1. SLR methodology adopted from Kitchenham et al. [11].

# 3.1. Planning the Review

Careful planning is essential for conducting a successful SLR. This plan provides a foundation for the subsequent phases and tasks of the SLR. This phase comprises the following tasks:

- 1. Defining the need for the review
- 2. Formulating the research questions
- 3. Developing a review protocol

In the following, we provided details of each one of these tasks.

#### 3.1.1. Defining the Need for the Review

As argued above, next-items recommender systems may assist businesses as well as customers in various tasks. As no SLR exists for next-items recommender systems, to the best of our knowledge, we conducted one to help other researchers contribute to this knowledge area. A comprehensive SLR is performed to discuss existing techniques, datasets, and evaluation measures to help other researchers further the research in this domain. 3.1.2. Formulating the Research Questions

Research questions provide the foundation of an SLR as all activities in the subsequent phases must be aligned with and answer these questions. We used the PICOC (Population, Intervention, Comparison, Outcome, Context) framework proposed by Petticrew and Roberts [24] to define the research question for the study, the elements of which are given below:

- **Population**—Shopping baskets of customers;
- Intervention—Customers' next-items recommender systems;
- **Comparison**—None;
- Outcome—Comparative study of next-items recommender systems;
- **Context**—Works related to next-items recommender systems in various domains; The research questions defined for the study are as follows:
- 1. RQ1: How can a next-items recommender system support businesses and individuals in their tasks?
- 2. RQ2: Which conventional machine learning techniques have been proposed for nextitems recommender systems?
- 3. RQ3: Which contemporary deep learning techniques have been employed for nextitems recommender systems?
- 4. RQ4: Which datasets have been used by researchers to implement their proposed solutions?
- 5. RQ5: In addition to the general evaluation measures used for prediction problems, which special measures have been proposed by researchers for evaluating the nextitems recommender systems?
- 6. RQ6: What are some challenges and open problems in the next-items recommender systems?

3.1.3. Developing a Review Protocol

5.

A well-defined review protocol is essential for an unbiased and thorough SLR. A review protocol includes various criteria used to search and select the studies. These studies are synthesized to conduct the SLR. In the following, we describe how we performed this process in this study.

**Resource Selection:** Because of the quality of publications, popularity, and relevance to the research topic as revealed by an analysis of related works, we conducted searches using the following digital libraries:

- 1. IEEE Xplore (https://ieeexplore.ieee.org/, accessed on 7 June 2022);
- 2. Science Direct (https://www.sciencedirect.com/, accessed on 7 June 2022);
- 3. Springer Link (https://link.springer.com/, accessed on 7 June 2022);
- 4. ACM Digital Library (https://dl.acm.org/, accessed on 7 June 2022);
- 5. Web of Science (https://clarivate.com/, accessed on 7 June 2022);
- 6. Scopus (http://scopus.com/, accessed on 7 June 2022).

Keywords and synonyms: We used the following terms for searching for related papers:

- 1. The terms "predict," "prediction," "recommender," and "recommendation systems" for predicting the future shopping basket of a customer;
- 2. The terms "grocery," "e-commerce," and "shopping" for recommendations in the domain of shopping;
- 3. The terms "next-items" and "next basket" for predicting the next-items list of items for customers;
- 4. The terms "collaborative filtering", "deep learning", "kNN", "neural networks", and "CNN" for specialized recommendation systems;

The term "machine learning" for survey papers or general solutions to the problem.

**Query Strings:** The following query strings were formed using various combinations of terms given above:

1. (recommender OR recommendation) AND ((e-commerce OR grocery OR shopping OR (next AND (item OR basket))) OR (predicting AND customer AND shopping);

- 2. (collaborative filtering) AND (Predict\*) AND (e-commerce OR grocery OR shopping OR (next AND (item OR basket)));
- 3. (machine learning) AND (prediction AND customer AND shopping) AND (e-commerce OR (next AND (item OR basket)));
- 4. (deep learning) AND (predict\* AND customer AND shop\*) AND (e-commerce OR grocery OR shopping OR (next AND (item OR basket))).

Note that the search-related features and the exact syntax used vary from one data source to another. Therefore, the search strings above were used to construct strings semantically equivalent to the given string but specific to each data source. The query strings were applied to the title, abstract, and keywords.

**Inclusion Criteria:** the papers satisfying the following inclusion criteria passed the first selection filter before going through the more rigorous quality assessment given below:

- 1. Peer-reviewed papers that have been published;
- 2. Papers published between 2017 and 2022 to ensure the inclusion of recent studies;
- 3. Papers published in journals and conferences only;
- 4. Papers written in the English language;
- 5. Papers related to customers' next-items recommender systems;
- 6. Papers related to one or more research questions of this study;
- 7. Studies presenting recommender systems for customers' next-items recommendations using any conventional machine learning approach;
- 8. Studies presenting recommender systems for customers' next-items recommendations using any deep learning approach.

**Exclusion Criteria:** the papers satisfying the following criteria were excluded from the review:

- 1. Papers not related to next-items recommender systems;
- 2. Papers published in any language other than English;
- 3. Papers published prior to 2017;
- 4. Papers published in journals or conferences not following a peer-review process;
- 5. Papers unrelated to any one of the research questions of this study;
- 6. Any types of documents other than research articles such as thesis, white papers, reports, commentaries, and editorials;
- 7. Repeated papers found in more than one source.

**Quality assessment:** The selected papers were passed through the quality assessment criteria given below for final selection for the review:

- 1. The study has well-defined objectives.
- 2. The study reports the model and findings consistently and coherently.
- 3. The research methods and process are detailed clearly.
- 4. The study applies the proposed model/algorithm to a dataset.
- 5. The evaluation metrics are clearly described and measured.
- 6. The findings of the study are credible.
- 7. The findings of the study are important.
- 8. The study compares its findings with the most suitable and most recent alternatives.
- 9. The study provides sufficient information to replicate its findings.

# 3.2. Conducting the Review

After defining the research questions and developing a review protocol, we adopted a systematic process for study selection, as shown using a PRISMA [25] flow diagram in Figure 2. We executed the search queries on the six sources given above to identify and collect research papers. The initial search resulted in 2733 papers. A total of 2514 papers were selected for screening after removing 219 duplicate papers. The inclusion and exclusion criteria defined above were applied to obtain 260 available studies in this next step. Finally, the quality criteria were applied to exclude 170 studies, and the remaining 90 studies were selected to be included in the review. Thus, by applying the filtering process given above,

9.6% of the search results were included as the primary studies. This means that only a fraction of the studies from initial search results could meet the study's inclusion, exclusion, and quality criteria. It indicates that the selected studies are of high relevance and quality.





The chronological distribution of the selected papers is shown in Figure 3. Most of the included papers (24 studies) were published in 2020, closely followed by 2021, when 22 papers were published. The number of papers published in 2019 and 2018 was 18 and 10, respectively. There were 8 papers published in both 2017 and 2022.



Figure 3. Chronological distribution of selected papers.

# 3.3. Reporting the Review

The last phase of SLR comprises a write-up of the findings and sharing them with the scientific community. We reviewed the selected papers thoroughly to provide answers to the research questions given above. As far as the specific research questions (RQs) are concerned, the RQ1, which relates to highlighting the applications of NIRSs in supporting individuals and businesses in fulfilling their tasks, has been addressed in the Introduction section. Following this, each of Sections 4–8 is aimed at addressing RQ2 to RQ6, respectively. In this way, the following discussion provides details about the proposed conventional and deep learning techniques for predicting future baskets of customers and highlights the strengths and weaknesses of each strategy. Furthermore, we also discuss the most popular datasets and evaluation measures relevant to the problem.

#### 4. Conventional Techniques for Customer's Next-Items Recommendation

This section aims at answering the RQ2, which is related to examining the role of conventional machine learning techniques for the development of NIRSs. The most popular techniques in this category include Markov chains, collaborative filtering, and k-nearest neighbors.

#### 4.1. Markov Chains (MC)

Sequential models [26,27], based on Markov chains, can be used to predict the next purchase based on the last purchase in sequential transaction data. Pattern-based models can refine the next purchase based on the previous purchase in general and the user's preferences in specific cases. Hybrid models combine sequential models with patternbased models. When a Markov model is used for sequential recommendation, future user behaviors are presumed to depend only on the last or last few behaviors [28]. The general model makes recommendations based on a customer's complete purchase history rather than on sequential behavior [29]. This model has a significant advantage because it captures sequential behavior to give better recommendations. However, the MC models, which rely only on the last behavior or several behaviors in a long sequence, are inadequate to capture the intricate dynamics of more complex scenarios by taking account of only the final behavior or some behaviors. Additionally, data sparsity may limit their effectiveness.

Markov models are stochastic models where data are sequentially generated and randomly changing states, presuming that future states will be determined by the present state, not the previous states. We are concerned at each step with sequential measures and the optimization approach to make the best decision out of many options when we use a Markov chain model where we can observe the states directly. The input for these models is a timestamped or ordered list of users' past interactions, and Markov-based RS is sometimes called sequence-aware RS or time-aware RS. As the sequence-aware RS incorporates context in the form of users' short-term actions gauged from their long-term past moves, it can be considered a special case of the context-aware recommender system (CARS) [30]. Table 2 provides a summary of the Markov chain techniques used for a customer's next-items recommender systems.

Study	Objective	Method/Model/ Features	Strength	Weakness	Source
[31]	Considering interactions in the historical sessions and the changing semantics of an item over time	МС	Hybrid representations allow dynamic attention to each factor while maintaining a balance between general preferences and sequential patterns	Sparsity	IEEE/J/ 2021
[32]	Comparing the mixed model with state-of-the-art methods for evaluating the music predictions	Hidden Markov model (HMM)	Developing a large-scale real-world dataset in a Kaggle competition	Poor Markov property for general cases; Missing real-time user feedback	IEEE/CP/ 2019
[33]	Proposing an intelligent recommender system	MC and grouping genres	The merger of two different techniques improves the quality of RS	Data sparsity	SD/J/ 2020
[34]	Utilizing social and temporal information in RS	МС	Addresses user cold-start issue by a two-level model founded on MC at both user-level and user group level to consider user picks dynamically	Data sparsity	SD/J/ 2019
[35]	Developing a context-aware recommendation approach to handle data sparsity issue	MC/CARS	Embedded features selection to handle the data sparsity issue	Scalability	Springer/J/ 2021
[36]	Solving the data sparsity problem of the transition matrix	МС	Integrating interest-forgetting attributes, social trust relation and item similarity in personalized Markov model	Single-user transition matrix data sparseness; cold start	Springer/C/ 2019
[37]	Exploiting the hierarchical hidden Markov model (HHMM) to extract latent context from the data	Hidden Markov model	The proposed model significantly improves performance both in terms of accuracy and the diversity of recommendations	Scalability	Springer/J/ 2019
[38]	Proposing sequential offers to users based on a succession of the user's prior reaction to recommendations	МС	The study proposes the dynamic model learning technique for a time-dependent next-basket recommendation, which together sports the dynamic character of user selections and item relations	Sparsity; scalability	Springer/J/ 2019
[39]	Handling personalization and a lack of semantics among recommended items	МС	The study proposes a model to enhance e-commerce product recommendation using semantic context and sequential chronological acquisitions	Scalability	Springer/J/ 2021
[40]	Interactive sequential basket recommendation, which iteratively predicts next baskets by learning the intrabasket/interbasket couplings between items and both positive and negative user feedback on recommended baskets	МС	Intrabasket/interbasket couplings, incorporating the user selection and nonselection	A large number of items in the next basket lead to unfair comparison, underfitting, and poor performance	ACM/J/ 2021

Table 2. Markov chain techniques used for customer's next-items recommender systems.

# 4.2. Collaborative Filtering

Generally, collaborative filtering (CF) methods forecast the user's preferences based on previous interactions between users and items. Thus, they collaborate on the ratings of several related users to make recommendations. Additionally, a few approaches employ optimization techniques to devise a training model in the same manner as classifiers create a training model using marked data. The CF, which has become one of the most popular and widely used applied models in recommender systems (RSs), refers to connecting the applier's data with that of similar applications based on purchasing customs to recommend for the next basket [41,42]. When Amazon recommends an item to a client based on their previous buying history and buying activity of those who purchased that same item, it uses CF techniques. A market segmentation method based on customer behavior corresponding to demographic and psychographic metrics is applied to e-commerce platforms. A critical aspect of CF is its scalability and accuracy. Customer satisfaction requires trendy processing space and speed optimization on an international scale to address demand scalability and efficiency.

CF approaches are classified into two groups: memory-based technique [43] and model-based technique [44,45]. The memory-based approach identifies similarities between the applications by computing a similarity function comparable to the cosine formula. The model-based strategy seeks patterns in the datasets and utilizes data from that pattern to construct new models. It also employs some methods such as matrix factorization [46].

#### 4.2.1. User-Based Collaborative Filtering (UBCF)

A client set is chosen by the UBCF algorithm based on clients with similar ratings [47]. Based on others' evaluations in a similar client set, it can anticipate the client's opinion about another item. Finding the best set of clients for the selected client is one of the most challenging tasks of this calculation. A neighbor record is constructed after distinguishing the client's likeness to others based on their closeness and assessing the likeness to other neighbors [47]. To obtain the proposed suggestion results, one can forecast client appraisals of specific things based on the rating history of neighbors.

#### 4.2.2. Item-Based Collaborative Filtering (IBCF)

An IBCF approach is used to resemble the comparability of objects. It anticipates the rating of a related item by considering the user's current rating of the corresponding item [47]. A parameterized model recommends the item having the highest rank to a user [48].

#### 4.2.3. Pros and Cons of CF Approaches

Compared with content-based filtering (CBF), CF methods are often used for RS. CF models offer several advantages over CBF approaches, including the capacity to handle domains where content-related items are scarce and in cases where contents, such as opinions, are hard to process. It also provides serendipitous recommendations, another advantage of the collaborative technique. However, the CF filtering methods have some potential drawbacks, which are listed below.

**Cold-start problem:** occurs when there are insufficient details about an item or user to make precise forecasts [49]. Because of this, the RS's performance reduces drastically. Since the system does not have a rating with the new item or the new user, there will be an empty profile, limiting the ability to recognize their preference.

**Data sparsity:** RS experiences a data sparsity problem when there are few or no users or ratings, causing recommendations to be inaccurate. This results from a lack of information about users and item purchase history [50]. A sparse user–item matrix is created when the number of items rated by users is relatively low [49]. Additionally, it impairs the ability to locate successful neighbors and ultimately deteriorates the recommendation process.

**Scalability:** Scalability is another aspect of RS to take into account. In recommender systems, scalability is a crucial factor. The scalability of a system can be determined by measuring the average response time it takes for it to process and respond to a request.

**Serendipity:** By producing and presenting unexpected and valuable recommendations, serendipity indicates that the RS has been able to amaze the user. In e-commerce, for example, the system can also recommend one or more items that are likely to satisfy the user according to the serendipity theory. The serendipity is determined by whether a recommendation is novel or not and whether or not it positively surprises the user. Consequently, a typical investigation thinks that, in this case, the recommendations will not become monotonous. However, a system might want to attempt to estimate the prospects, where a user can determine a relevant item [51]. Through a serendipitous recommendation, the user can discover a compelling item that might otherwise go unnoticed. Table 3 below summarizes the advantages and disadvantages of collaborative filtering techniques, and Table 4 provides a summary of these techniques.

Techniques	Representative Algorithm	Advantages	Disadvantages
Memory-Based Collaborative Filtering (CF)	User-Based CF; Item-Based CF	Efficient scalability; Easy to implement; Adding data is straightforward; Content is not a factor	Cold start, data sparsity and scalability issues; Relying on the precise recommendations; Biased for massive dataset
Model-Based Collaborative Filtering	Slope one CF	Improves the prediction efficiency; Enhances sparsity and scalability issues	Model is pricey; Loss of details in a factorization matrix
Hybrid Collaborative Filtering	A blend of memory-based and model-based	Controls the sparsity constraints; Enhances the prediction efficiency	Raised complexity; Challenging for implementation

 Table 3. Comparison of collaborative filtering techniques for recommender systems.

Table 4. Collaborative filtering (CF) techniques used for customer's next-items recommender systems.

Study	Objective	Method/Model/ Features	Strength	Weakness	Source
[52]	Proposed innovator-based collaborative filtering (INVBCF) for recommending cold items	Collaborative filtering	Cold items can be managed in the recommendation list via innovators, achieving the balance between serendipity and accuracy	Poor performance on sparse data	IEEE/J/ 2019
[53]	To help customers acquire recommendations on the purchase of desired products	Collaborative filtering, clustering, and association rule mining	The item-to-item collaborative filtering is selected as it furnishes recommendations to all customers regardless of the number of buyers and in the ratings given already on priority	Cold start, data sparsity	IEEE/CP/ 2020
[54]	This article concerns diverse issues faced by recommender systems and proposes solutions to them	Classification, collaborative filtering, association rule mining and sequence rule mining	User classification	Cold start, data sparsity, scalability	IEEE/CP/ 2017
[55]	Users' purchase prediction using clicking behavior features	CF	Evaluation of the proposed system with the dataset provided by Ali Mobile Recommendation Competition held in 2015	Cold start	IEEE/CP/ 2017
[56]	A book recommender system	User-based CF	A user-user similarity matrix	Cold start, data sparsity	IEEE/CP/ 2020
[57]	To make accurate and efficient recommendations	CF	Multiple algorithms are used to generate user-based recommendations and item-based recommendations	Cold start, data sparsity	IEEE/CP/ 2020
[58]	AI-based recommender system	CF	A thorough review of CF-based recommender systems	Data sparsity	IEEE/CP/ 2020
[59]	Transition-based cross-domain collaborative filtering	CF	Overcomes the cold-start problem	Cross-domain review analysis, data sparsity	IEEE/CP/ 2020

# Table 4. Cont.

Study	Objective	Method/Model/ Features	Strength	Weakness	Source
[60]	Improvement of item-based CF algorithm	CF	Proposed group weighted rating method to improve item-based CF	Cold start, data sparsity, scalability	IEEE/CP/ 2020
[61]	Enhancement of item-based CF recommendation	CF	Proposed similarity-based algorithm using modified cosine similarity algorithm	Cold start	IEEE/CP/ 2020
[62]	Predicting the ratings for cold-start items by exploiting items' textual descriptions	CF	The study focuses on the item cold-start problem	User cold-start problem, data sparsity, scalability	SD/J/ 2018
[63]	Suggesting fashion goods for clients by enhancing the existing CF procedure to consider the features of fashion products	CF	Both online and offline purchase data used to generate recommendations	Experimental results cannot demonstrate the effects of using online and offline data, as well as the effects of a decline in preference over time	SD/J/ 2018
[64]	Curtailing the size of the recommendation list	CF	Used item weight generator block to rank recommended items at the appropriate position in the list	Cold start, data sparsity	SD/J/ 2020
[65]	Considering users' retrieval intentions for more personalized recommendations	CF	Incremental strategy for the collaborative filtering recommendation algorithm	Cold start, data sparsity	Springer/J/ 2020
[66]	To overcome the data sparsity problem of the CF algorithm	CF	Optimizing biclustering and information entropy	Scalability, cold start	Springer/J/ 2019
[67]	Serendipity-oriented location-based recommender system	Association rule mining and CF	A personalized recommender assistant which offers both precise and spontaneous points of interest (POIs)	Cold start	Springer/CP/ 2018
[68]	Striking a balance between accuracy and serendipity	CF	The proposed strategy employed user records and feedback to optimize and balance serendipity and accuracy	Cold start	Springer/CP/ 2017
[69]	Improving search query analysis by using collaborative filtering and naive Bayes algorithms	CF, naïve Bayes	Improved F1 measure by 14% compared to simple naïve Bayes technique	Cold start, data sparsity	Springer/CP/ 2017
[70]	Improving the quality of the user-item matrix by normalizing the frequency of item purchase	Clickstream-based CF recommender system	Improved quality of ratings	Poor integration between sequential patterns of the user-item matrix.	Springer/J/ 2018
[71]	Developing a hybrid recommender system	CF	Improved recommendations through better customer behavior modeling and enhanced user–item matrix	Inability to incorporate multiple data source-based sequential patterns; no provision for infrequent users	Springer/J/ 2019
[72]	Proposing a neural network-based collaborative filtering framework	CF	The proposed model merges the strength of the linearity of MF and the nonlinearity of multilayer perceptron for modeling the user-item latent structures	Data sparsity	ACM/J/ 2017
[73]	Implementing social, collaborative mutual learning for recommender system	CF	Combining item-based CF and social CF to improve the recommendation performance	Scalability	ACM/J/ 2020

# 4.3. kNN

The k-nearest neighbors (kNN) algorithm classifies a given set of instances based on their similarity with k-closest training examples in a given dataset. Thus, a typical application of kNNs for recommender systems consists in finding clusters of similar users based on common item ratings and making predictions based on the average rating of top-k nearest neighbors. In this way, kNNs have been adopted to find users or items similar to their neighborhood. The existing methods use different approaches to apply kNN algorithm-based filtering, e.g., by applying filtering to each criterion separately or jointly considering all criteria [14].

Kuo et al. [74] adopted a perturbation-based kNN with an imputation technique to manage the data sparsity problem. The main idea of the technique is to search for optimal weights among similarities with the help of metaheuristics to obtain better recommendation performance. Hu et al. [75] based their proposal on the argument that RNN-based methods do not directly capture important patterns related to item frequency and thus result in lower performance in next-basket recommendation scenarios. Their kNN-based model utilizes two patterns, i.e., repeated purchase pattern and collaborative purchase pattern in relation to the personalized item frequency (PIF), by incorporating them into the temporal dynamics of the repeated purchase. Given the model's simplicity, the model performs well for the NBR.

Table 5 provides a summary of KNN techniques used for customers' next-items recommender systems.

Study	Objective	Method/Model/ Features	Strength	Weakness	Source
[74]	Addressing the problems of data sparsity and similarities selection	kNN with the densest imputation	Model simplicity	High computational time to build the model; Low scalability; Limited validation	SD/J/ 2021
[75]	Enhancing NBR recommendation by capturing two important patterns associated with personalized item frequency (PIF)	kNN	Effective use of personalized item frequency (PIF) with a simple method	Low scalability	ACM/C/ 2020

Table 5. The kNN techniques used for customer's next-items recommender systems.

#### 5. Deep Learning Techniques for Customer's Next-Items Recommendation

This section aims at answering the RQ3, which is related to reviewing the role of deep learning techniques for the development of NIRSs. Deep-learning-based recommender systems [18,76] possess some critical advantages over traditional content-based and collaborative filtering methods, as briefly described in the following:

- (i) Deep learning (DL) enables modeling of nonlinear interactions found in data using nonlinear activations (e.g., ReLU and Sigmoid). Unlike conventional methods that are linear in nature, nonlinear modeling of interactions allows capturing of more complex user-item interaction patterns. Consequently, users' preferences can be more precisely reflected.
- (ii) DL techniques can learn the descriptive information about items and users efficiently and thus enhance the recommender system's understanding of items and users.
- (iii) DL techniques have been shown to be a perfect fit for sequential modeling tasks (e.g., natural language processing) and thus work well for the temporal dynamics associated with user and item behavior.
- (iv) DL is highly flexible and allows combining different neural structures to construct more powerful hybrid models. This capability can be exploited to develop hybrid recommendation models that can capture and process varying characteristics simultaneously.

In the following, we discuss how deep learning has been employed in the context of recommender systems.

#### 5.1. CNN

Convolutional neural networks (CNNs) are deep architectures that enable pattern classification via a discriminative function. A CNN typically uses a perceptron to handle high-dimensional data. The modules within a CNN comprise stacked convolutional and pooling layers. A convolutional layer applies some filter to an input that results in an

activation. A map of repeated activations achieved in this way indicates the detection of a specific feature and is called a feature map. Alongside convolutions, various pooling operations such as max pooling and average pooling are applied to subsample the output of the convolutional layers. In general, CNNs are highly capable in tasks that involve the detection of highly specific features in input data [77]. So far as recommender systems are concerned, CNNs have been used to extract distinguishing deep features from textual review and thus enhance the representation of user and item.

Haihan et al. [78] developed a data crawler to obtain user reviews and product information from the database of a mall. A CNN-based model then extracts salient features to learn and predict matching rates between users and commodities. Khan et al. [79] used a word2vector to capture the semantics of the text in user and item embeddings and a CNN to extract the contextual details. The model handles both user and item metadata concurrently to deal with the sparsity problem. Some studies have developed end-to-end systems for recommending products for a specific context. Kavitha et al. [80] proposed a simple system that takes text or images as input and recommends choices of fashion outfits or accessories. The recommendation model is trained with a CNN using images and related text data. In a similar study, Reyes et al. [81] augmented the product recommendations using scores pertaining to preferences from past user interactions with the system. Yuan et al. [82] proposed a convolutional generative model based on a stack of holed conv layers to efficiently increase the receptive fields without depending on pooling operation. A residual block structure is employed to optimize the network depth. The model optimizes the training time and demonstrates the efficacy, particularly when long sequences of user feedback are involved. Table 6 provides a summary of CNN techniques used for customers' next-items recommender systems.

Table 6. CNN techniques used for customer's next-items recommender systems.

Study	Objective	Method/Model/ Features	Strength	Weakness	Source
[78]	Improving recommendations using salient visual features of products	CNN	Use of real-user and commodity dataset	Low scalability; High operation cost; Limited parameters considered	IEEE/C/ 2021
[79]	Addressing the sparsity problem	CNN, Word2Vec	Concurrent handling of semantics and contexts of user and item data	Temporal effect not addressed; Data noise issue may seriously affect performance	SD/J/ 2021
[80]	Offering a recommender system based on text and image inputs	CNN	Simplicity of the system.	Low scalability; Limited features considered; Limited validation	SD/C/ 2020
[81]	Improving recommendations using the score from past interactions	CNN	Simplicity of the system.	Low scalability; Limited features considered; Limited validation	SD/C/ 2020
[82]	To keep an ordered collection of all items the user interacts in a session	Generative CNN	Effectively addresses the problems with a typical session-based CNN recommender	Low scalability; Limited use for long sequences of user feedback	ACM/C/ 2019

# 5.2. RNN

Recurrent neural networks (RNNs) are a class of deep learning models specifically suitable for modeling sequential data. In their purest form, RNNs use temporal layers to capture sequential data. Therefore, they naturally become an effective way of dealing with the temporal dynamics involved in user behavior and item evolution [83]. RNNs contain loops and memories to retain former computations and changes. As the primary form of RNNs suffers from vanishing gradient issues, which means decreased performance in modeling long-term activities and temporal dependencies, its variants, including long short-term memory (LSTM) and gated recurrent unit (GRU), are often deployed to address such problems.

Choe et al. [84] incorporated item usage sequence and related time-series data from user history using a hierarchical structured RNN to improve recommendations. Chen et al. [31] aimed to infer the user's general preference based on the most relevant items over historical sessions and used this information to recommend the next items in the context. Rabiu et al. [85] enhance recommendation accuracy by quantifying the sentiment bias to extract the user's true opinion and thus deal with data sparsity and imbalance problems. Their adaptive LSTM models the drifting of users and item features dynamically, enabling more accurate rating predictions. Wang et al. [86] performed a rigorous projection of users' individual preferences with all user interactions. The former was modeled using tensor product operation, while the latter used the quantum many-body wave function (QMWF). Lo et al. [87] proposed a two-stage RNN model that combines information from the user, item context, and several shopping signals to optimize relevance and conversion. Han et al. [88] developed an on-device deep learning framework that provides sequential recommendations while uploading no raw data or intermediate results. Table 7 provides a summary of RNN techniques used for customers' next-items recommender systems.

 Table 7. RNN techniques used for customer's next-items recommender systems.

Study	Objective	Method/Model/ Features	Strength	Weakness	Source
[84]	Enhancing the use of temporal properties of the user history	Multilayered RNN, long- and short-term time series	Effectively uses temporal properties in time-series data to improve recommendation quality	Review content and ratings not considered; Low scalability	IEEE/J/ 2021
[31]	Balancing the combination of general preference and the sequential patterns factors in recommendations	LSTM	Deducing high-level preferences of users based on the observed sessions	High operation cost; Limited validation	IEEE/J/ 2021
[85]	Dealing with problems of data unbalance and sparsity	LSTM	Dynamic modeling of user and item features	High operation cost	SD/J/ 2022
[86]	Learning user preferences by leveraging the mathematical formalisms of quantum theory	RNN, quantum many-body wave function (QMWF)	The mathematical formalism of QT is exploited	Low scalability; Limited validation	ACM/C/ 2019
[87]	Enhancing recommendations for relevant items on the items details page and increasing discoverability of relevant items	Two-stage RNN	Formulates page-level optimization problem as a sequential ranking problem and accommodates heterogeneous feature types	Low scalability; Limited validation	ACM/C/ 2021
[88]	Enhancing sequential recommendation while ensuring privacy by sending no raw data out of the mobile device	RNN, Automated Gradual Pruner (AGP)	Addresses privacy as the primary concern and addresses issues specific to mobile devices	Low scalability; Limited validation	ACM/C/ 2021

#### 5.3. Graph Neural Networks

Graph neural networks (GNNs) use graph data to aggregate features from neighbors iteratively. Next, they employ a propagation process consisting of stacks of layers to integrate the aggregated feature information with the current central node representation [89]. The data commonly used by recommender systems, such as user interaction and item evolution data, can essentially be represented in the form of a graph. Therefore, GNNs provide a unified perspective for effectively modeling heterogeneous data found in recommender systems. Additionally, instead of only capturing the collaborative signals in recommender systems, GNNs enable the encoding of the complete topological structure of the user and item representations.

The underlying idea of Liu et al. [90] is the understanding that an adequate representation of relationships among products as product graphs can incorporate the strengths of the graph's topological structure and improve relationship prediction. They utilized an item relationship graph neural network to learn multiple complex connections in items. Wong et al. [91] used information about users, items, and conversations using a conversation knowledge graph to make click-through rate prediction models aware of the states associated with users, dialogues, and items. Tao et al. [92] aimed to improve the recommendation accuracy by integrating item trend information obtained from the user's interaction history with the user's short-term preference. They used a gated graph neural network jointly with a self-attention layer to enhance the representation of item trends. Zhang et al. [93] argued that the similarity resulted in node-embeddings because over-smoothing of graph convolutions leads to a decrease in recommendation performance. To solve this issue, their approach enhances the graph convolutional network by unifying it with the label propagation algorithm in a way that the former focuses on the basic recommendation model while the latter consistently regularizes the training edge weights. An attention network is used to retain the information of each user-item pair. In order to improve the click-through rate prediction, Zhao et al. [94] have attempted to alleviate the difficulties related to the sparseness of user behaviors. To this end, they developed a model for jointly learning both search and recommendation scenarios by sharing information between the two scenarios using a unified graph. The information from the two heterogeneous scenarios is aggregated using a dedicated layer in the graph neural network model. Table 8 provides a summary of GNN techniques used for customers' next-items recommender systems.

Study	7 Objective Method/Model/ Features		Strength	Weakness	Source
[90]	Improving product relationship representation by exploiting the graph's topological structures	Graph neural network, edge relational network, multihop dependencies	Improves inference of complex dependencies in item relationships	Limited validation	IEEE/J/ 2021
[91]	Solving the problems of data scarcity and sparseness related to click-through rate prediction	Conversation knowledge graph, deep convolution network	Utilizes various types of information from user states and dialogue interactions	Low scalability; Entity descriptions and knowledge schema are not considered	IEEE/C/ 2021
[92]	Improving the modeling of the current item by incorporating the item trend information	Gated graph neural network, self-attention layer	Item trend information from the implicit history of the user incorporated into subsequent recommendation tasks	Low scalability; Cold-start problem	Springer/J/ 2021
[93]	Solving the problem of over-smoothing of graph convolutional networks to improve the recommendation performance	Graph convolutional network, label propagation algorithm, attention network	Improved recommendation performance by the unification of GCN and LPA	Static preference modeling; Low training performance	Springer/J/ 2022
[94]	Alleviate the sparseness of user behaviors by jointly learning from search and recommendation scenarios	Graph neural network with the aggregation layer	Search and recommendation scenarios are jointly learned	Low scalability; Limited validation	ACM/C/ 2022

Table 8. GNN techniques used for customer's next-items recommender systems.

# 5.4. Other Deep Networks (DNN)

In addition to the more common deep learning techniques described above, other deep networks are also often implemented [18]. The main idea is to develop a deep network architecture containing stacked layers of nonlinear transformation. Some of the techniques include: (i) a feed-forward neural network having multiple hidden layers between the input and output layer (e.g., a multilayer perceptron), (ii) stacks of two-layer neural networks containing a hidden layer and a visible layer (e.g., restricted Boltzmann machines), and (iii) an autoencoder model that attempts to recreate its input data in the output layer. In the context of recommender systems, these models generally view a recommendation as a two-way interaction between users' preferences and items' features and model it by constructing a dual neural network.

Wang et al. [95] aimed to solve the user cold-start problem by modeling users based on a mechanism that takes a common set of users from online shopping and ads domains and utilizes it to initialize recommendations. Similarly, Ahmed et al. [96] addressed the cold-start problem by learning features common to users from different domains and thereby identifying similar users. Some studies have developed end-to-end systems for recommending products for a specific context. Abinaya and Devi [97] address the conflicts found in rating scores and user sentiment in reviews by incorporating both into a context. The item splitting method is used to model context by creating fictitious items based on the context. In Qin et al. [98], a denoising generator is proposed to decide whether each item in the historical basket is relevant to the target item or not. It creates positive and negative sub-baskets for each basket of each user.

Further, a context encoder determines if it is a relevant preference or noise. An anchorguided contrastive learning process is used. Ngaffo and Choukair [99] aimed to manage the problems of data sparsity and cold start by incorporating an enhanced matrix factorization technique used to extract the user's and item's features within a deep learning structure employed for the prediction. The technique effectively alleviates the limitations of the matrix factorization process. Liu and He [100] improve the recommendation performance by two techniques. First, they improve the initialization of latent feature vectors by learning the initial values of users and items using a deep autoencoder. Next, a users' socialtrust learning model is proposed to augment the recommendation by considering the recommendations of trusted friends and related communities. Table 9 provides a summary of DNN techniques used for customers' next-items recommender systems.

Study	Objective	Method/Model/ Features	Strength	Weakness	Source
[95]	Alleviating the cold-start problem	DNN, Stacking model, Word2Vec	Enhances user modeling	High operation cost; Limited validation	IEEE/J/ 2020
[96]	Alleviating the cold-start problem	DNN, latent factor model, multilayer perceptron	Models relationships in cross-domain scenarios; Generates useful recommendations in high sparsity scenarios	Implementation complexity	SD/J/ 2021
[97]	Deriving sentiment by merging ratings and reviews to enhance top-n recommendations	Context-specific sentiment-based stacked encoder (CSSAE), item splitting	Computing concrete preferences of the user by integrating rating and review for a given context	Low scalability	Springer/J/ 2021
[99]	Solving problems of data sparseness and cold start to increase recommendation quality	Matrix factorization, combination of multilayer and single-layer perceptron	Effectively solves issues with latent factors	Low scalability; Limited validation	Springer/J/ 2022
[100]	Solving the problem of random initialization of latent features, which play an important role in obtaining a good local minimum	Deep autoencoder, social-trust holistic learning model, item classification	Effective integration of users and items in latent feature vectors	High operation cost; Limited validation	Springer/J/ 2022
[98]	Enhancing NBR by denoising the baskets and extracting credibly relevant items	Context encoder, contrastive learning	Item-level denoising for a basket	Low scalability; Limited validation	ACM/C/ 2021

Table 9. DNN techniques used for customer's next-items recommender systems.

#### 5.5. Hybrid Networks (Attention+)

The flexibility of deep neural networks is often exploited to integrate different neural building blocks to construct models that are much more powerful and expressive for specific tasks. To this end, one of the most commonly used models in recommender systems is the attention model [101]. Motivated by human visual attention, these models provide differential architectures to filter the informative features from input data. Recommendation systems leverage the attention mechanism to select the most expressive items from inputs such as user reviews. Zhang et al. [102] incorporated aggregations of sequential behavior records and personalized tastes of users within an adaptive attention mechanism to handle sparse interaction data effectively. The model improves parallelism in handling correlations without increasing its complexity. Thaipisutikul and Shih [103] aimed to capture the hierarchies of relationships between context and items affecting the users' preferences considering long-term representation. To this end, they adopted an attention network to combine the information from users' short-term sessions with the relevant long-term representations. An attentive neural network was then proposed to identify the highly relevant items to the recent session.

Furthermore, there are different ways to integrate attention mechanisms with other mainstream techniques to develop a hybrid method. A majority of the recent techniques have combined RNN with the attention mechanism. Che et al. [104] improve the recommendation performance by incorporating the historical information from all historical baskets of the user with the item-level inference from the most recent basket. An adaptive attention mechanism has been combined with an RNN. Similarly, Liu et al. [105] focus on the attributes of items within a basket and exploit their relationships within a single basket and the past baskets of the user to improve the recommendations. Dau and Salim [106] have utilized a topic model to extract the domain-specific aspect of the product and associated sentiment lexicons fed into an LSTM encoder via an interactive neural attention mechanism. Additionally, they used neural coattention to improve the learning of finer interactions. Ouyang and Lawlor [107] developed a character-level model based on LSTM with attention to generate personalized reviews to manage common problems (e.g., noise and unrelated content) in reviews. Cui et al. [108] aimed to alleviate the difficulties of RNNs in acquiring the short-term interest of a user based on recent engagements. They extend an RNN to form a context by capturing and modeling multiple adjacent factors. An attention mechanism is then used to find the most likely items contributing to short-term interest in that particular context. Zhang et al. [109] argued that the accuracy of next-item recommendations often suffers because users' long-term behavior is processed only sequentially, and their attention to individual items is not considered. To solve these problems, their method equips the combination of a bidirectional LSTM and a gated recurrent unit (GRU) module with a userbased attention mechanism. Thus, the Bi-LSTM keeps track of users' long-term behavior, whereas the GRU captures their latest intent.

Some techniques have also included a CNN with the combination of RNN and attention. Li and Xu [110] extract aspects from textual reviews and build two separate paths to deal with user and item reviews. Each path comprises a CNN, an LSTM, and the attention mechanism to process local and global features. Similarly, some other techniques have adopted a CNN with the attention mechanism. Du et al. [111] used two parallel networks to enhance user and item feature representation. The two networks model users and items in the review text. The two models are incorporated into a hierarchical attention mechanism. The network layer corresponding to item representation is further enhanced by introducing a mechanism to produce highly relevant features. A CNN was used for feature extraction. Table 10 provides a summary of hybrid techniques used for customers' next-items recommender systems.

Study	Objective	Method/Model/ Features	Strength	Weakness	Source
[102]	Utilizing knowledge from aggregations of users' behavior and taste records to deal with sparsity and improve recommendations	Attention layers with position embeddings	Effective time-sensitive next-item recommendation; Low model complexity with high parallelism.	Low scalability	SD/J/ 2021
[103]	Improving the extraction of sequential behaviors from contextual info and representation of users' short-term preferences	Hierarchical attention network, neural attentive bidirectional GRU with sequential residual connection	Gradual refining of user's preferences via a hierarchical learning process	High operation cost; Limited context information	Springer/J/ 2020
[104]	Addressing the problems resulting from static use of attention-based RNNs	Adaptive attention-based RNN	Acts on all historical baskets as well as at item level in the most recent basket	High operation cost; Low scalability	IEEE/J/ 2019
[106]	Enhancing the use of user sentiments in reviews to improve the performance of recommendation	Attention-based RNN, neural coattention	Incorporation of sentiments with consideration of various aspects of a product	High operation cost	IEEE/J/ 2019
[105]	Incorporating the context (short-time interests and long-term preferences) to improve recommendations	Multilevel attention, item attributes	Enhanced basket-level recommendation performance	Low scalability; Minimal attributes information	IEEE/J/ 2020
[108]	Alleviating the limitations of RNNs in the acquisition of the short-term interest	Attention-based RNN, contextual hidden states	Contextual modeling of interest-based on recent factors	Does not explicitly model long-term interest	SD/J/ 2019
[109]	Capturing the user's general preferences as well as his/her latest intent	Hybrid network with Bi-LSTM and GRU modules	Combining the usual preference with the latest intent	High operation cost; Low training efficiency	Springer/J/ 2021
[110]	Improving the local and global aspect representations obtained from user reviews	Context embedding, local and global aspects FE	Incorporates preferences in different aspects; Determines local and global importance of each word in review	Computational expense; Synonymity problems	IEEE/J/ 2020
[107]	Generating compelling review-based explanations for input to reviewer-aware recommenders	Attention-based RNN	High prediction performance; Deals with item sparsity problems	Synonymity problems; Limited validation	IEEE/J/ 2021
[111]	Enhancing user and item feature representation	Hierarchical attention, CNN	High prediction performance on real-world datasets	High operation cost	SD/J/ 2021

Table 10. Hybrid techniques used for customer's next-items recommender systems.

# 6. Datasets

This section answers the RQ4, which is related to elaborating the commonly adopted datasets for the development of NIRSs. Table 11 provides an overview of the commonly used datasets within the problem domain. As evident from the table, several datasets have been developed by researchers in the problem domain based on sales records from popular online stores such as eBay, Amazon, or Taobao. The performance of machine learning techniques is directly proportional to the size of a dataset. Hence, researchers prefer to use the largest possible dataset available. There are three key features of datasets in the customer's

next-items recommender system: number of records, number of users (customers), and number of items (products). With over 100 million records, the datasets Taobao18 and UserBehavior have the highest number of records. AliExpress has the maximum number of users (>1.5 million), followed by Taobao18 with about 0.9 million users. The maximum number of products is gathered by the dataset Taobao20 (>0.9 million). All datasets are publicly accessible except AliExpress, which is available on request. Amazon, Movielens, JingDong, and Tafeng are the most popular datasets among the researchers' community.

Name	Used By	No. of Users (1000/s)	No. of Items (1000/s)	No. of Records (Millions)	Availability	URL	Year
Taobao18	[93]	988	4162	100	Public	https://tianchi.aliyun.com/dataset/dataDetail? dataId=649, (accessed on 30 May 2022)	2018
UserBehavior	[112]	970	4158	100	Public	https://tianchi.aliyun.com/dataset/dataDetail? dataId=649&userId=1, (accessed on 30 May 2022)	2017
Taobao20	[94]	83	9621	75	On request	taobao.com, (accessed on 30 May 2022)	2020
MovieLens	[113]	280	58	27	Public	https://files.grouplens.org/datasets/movielens/ ml-latest.zip, (accessed on 30 May 2022)	2018
eBay	[114]	40	5375	12	Public	github.com/urielsinger/Trans2D, (accessed on 30 May 2022)	2021
ValuedShopper	[98]	10	8	5	Public	https://www.kaggle.com/competitions/acquire- valued-shoppers-challenge/data, (accessed on 30 May 2022)	2014
Instacart	[98]	200	8	3	Public	https://www.instacart.com/datasets/grocery- shopping-2017, (accessed on 30 May 2022)	2017
Ali Express	[96]	1507	49	2	On request	https://doi.org/10.1016/j.eswa.2021.114757, (accessed on 30 May 2022)	2021
Yelp	[106]	169	63	2	Public	https://www.yelp.com/dataset, (accessed on 30 May 2022)	2021
Outbrain	[108]	66	69	0.83	Public	https: //www.kaggle.com/c/outbrain-click-prediction, (accessed on 30 May 2022)	2017
Epinions	[100]	49	140	0.66	Public	trustlet.org/downloaded_epinions.html, (accessed on 30 May 2022)	2003
Dunnhumby	[115]	36	5	0.52	Public	https://www.dunnhumby.com/source-files/, (accessed on 30 May 2022)	2014
Ta-Feng	[104]	14	12	0.50	Public	https://www.kaggle.com/datasets/ chiranjivdas09/ta-feng-grocery-dataset, (accessed on 30 May 2022)	2001
Amazon	[116]	58	50	0.32	Public	http://jmcauley.ucsd.edu/data/amazon/, (accessed on 30 May 2022)	2018
JingDong	[104]	10	9	0.21	Public	https://www.datafountain.cn/competitions/24 7/datasets, (accessed on 30 May 2022)	2018
Book-crossing	[93]	18	15	0.14	Public	http://www2.informatik.uni-freiburg.de/ ~cziegler/BX/, (accessed on 30 May 2022)	2004
Last.fm	[93]	2	18	0.09	Public	https://grouplens.org/datasets/hetrec-2011/, (accessed on 30 May 2022)	2011

Table 11. Most popular datasets in the problem domain.

Note that some datasets are not named by their authors (such as TaoBao18 and TaoBao20); these names are given in this table for identification purposes only.

# 7. Evaluation Metrics

This section aims at answering the RQ5, which is related to investigating the specialized evaluation metrics used for NIRSs. Researchers have proposed several measures to evaluate the quality and performance of recommender systems. These measures can be classified into rating prediction metrics, classification accuracy metrics, and ranking metrics [18]. We briefly describe the proposed measures under each of these categories in the following.

# 7.1. Rating Prediction Measures

These measures determine how correctly the recommender system relates the user to the items. The three commonly used measures in this category are mean squared error (*MSE*), root mean squared error (RMSE), and mean absolute error (MAE). Mean squared error can be expressed by the following formula:

$$MSE = \frac{1}{P} \sum_{i=1}^{P} (x_i - Y_i)^2$$
(1)

where P,  $x_i$ , and  $\hat{Y}_i$  are the total number of ratings, predicted ratings, and actual ratings, respectively.

As these measures are commonly used in machine learning, we refer a reader to literature such as Chai and Draxler [115] for further information regarding RMSE and MAE.

#### 7.2. Classification Accuracy Metrics

These metrics are used to determine the accuracy of classification, i.e., how correctly the recommender system identifies items of interest for a particular user. The most commonly used metrics in this category include precision, recall, F1 measure, and area under the curve (AUC). Additionally, we are usually interested in top-K recommendations for a particular user; hence, these measures are modified as prcision@K and recall@K, etc., to consider evaluation for top-K recommendations. For brevity, we refer a reader to literature for these measures, such as Powers [114], and focus on the specific metrics unique to the multilabel classification problem of next-items recommendation [117].

**Subset accuracy** evaluates the fractions of correctly predicted items, i.e., the predicted item set in a customer basket is identical to the ground-truth item set. It can be expressed mathematically as:

$$SubsetAcc(h) = \frac{1}{P} \sum_{i=1}^{P} \llbracket h(x_i) = Y_i \rrbracket$$
(2)

where h(.) is a classifier for the next-items recommender system,  $x_i$  is the ith item in a customer's basket x as predicted by this classifier, and  $Y_i$  is the ith item in the basket (ground truth). P denotes the total number of items in this basket.

**Hamming Loss** measures the fraction of misclassified in the customer basket, i.e., a relevant item is missed from the recommended basket, or an irrelevant item is recommended. Mathematically,

$$HLoss(h) = \frac{1}{P} \sum_{i=1}^{P} |h(x_i) \Delta Y_i|$$
(3)

where  $\Delta$  denotes the symmetric difference between both sets.

#### 7.3. Ranking Metrics

As usually top-K items are recommended to a user by the next-items recommender systems, the order of recommendations becomes important in predictions. The most common measures in this category include hit ratio, normalized-discounted-cumulative gains (NDCG), mean reciprocals ranks (MRR), mean average precision (MAP), and one-error.

**Hit ratio** measures whether an item under consideration appears in top-K recommendations to a user. It is defined as follows:

$$HR@K = \frac{Number \ of \ Hits@K}{|T|} \tag{4}$$

where *T* denotes all interactions of the customer with the recommended items.

**Normalized-discounted-cumulative gains (NDCG)** also considers the position of items in the recommendation list and assigns higher scores to interactions with top-ranked items. Mathematically,

$$NDCG@K = \sum_{i=1}^{K} \frac{2^{r_i} - 1}{\log_2(i+1)}$$
(5)

where  $r_i = 1$  if the target item is ranked at the position *i*, 0 otherwise.

**Coverage** is a metric used to measure the homogeneity of recommendations. It measures the average number of steps required to move from the top of the ranked list and cover all relevant items in the target customer basket. It can be expressed as follows:

$$Coverage@K = \frac{|U_{u \in U}R^u|}{|I|}$$
(6)

where U represents the user set,  $R^u$  represents top-K items recommended to the user u, and I represents the set of all items.

**Mean reciprocal rank (MRR)** represents the reciprocal average of the number of items in the recommended basket the user interacts with:

$$MRR@K = \frac{1}{|U|} \sum_{u \in U} \max(\frac{1}{R_u, G_u}, 0)$$
(7)

where  $G_u$  represents the ground truth; the function returns  $1/R_u$ ,  $G_u$  if  $G_u$  exits in top-K recommendations, 0 otherwise.

**Mean average precision (MAP)** is used to calculate the precision of top-*K* recommendation to a user. Average precision for one user (*AP*) can be given as follows:

$$AP@K = \frac{\sum_{n=1}^{K} Precision@n \times rel(n)}{\min(K, R_u)}$$
(8)

where Precision@n is  $G_u$ 's precision value at n in top-K, the function rel(n) = 1 if the item at rank n is recommended, 0 otherwise.

*MAP*@*K* can be calculated by calculating the average of all users' *AP* values.

**One-error** measures the fraction of recommended baskets whose top-ranked item is not in the ground truth.

$$OneError@K = \frac{1}{|U|} \sum_{u \in U} (R_{u1} \notin G_u)$$
(9)

where  $R_{u1}$  denotes the *K* top-ranked items for the user *u*.

# 7.4. Other Metrics

Despite the popularity of these evaluation metrics, the complexity of a customer's next-items recommendation task calls for novel measures. Jannach et al. advocate the need for devising new evaluation metrics such as serendipity, diversity, and domain-specific metrics [118]. These measures may improve the effectiveness of recommendations in the form of increased revenues for the businesses or higher customer satisfaction. Castells et al. have provided a comprehensive discussion on the use of diversity and novelty in recommender systems [119]. Another evaluation metric, exponential decay score, was introduced by Breese et al. [120], which considers the position of items in the recommendation list, similar to NDCG.

# 8. Challenges and Future Directions

This section addresses the RQ6 highlighting the key challenges and open issues in the problem domain of NIRSs. A customer's next-items recommendations is an exciting research area with many challenges and potential for innovative and valuable enhancements. Cold start has been plaguing the recommender systems for a long time. Recently, however, advanced techniques such as deep learning have reduced the severity of the problem, although there is still room for improvement. Another common problem with these recommender systems is data sparsity, when the number of items in a single basket is negligible compared to the total number of items available for recommendations (e.g., number of products in a store). Some deep learning techniques have made significant improvements, but the problem persists.

Currently, the systems make recommendations in a single domain. However, crossdomain recommendations can be an exciting extension of these systems. This can be achieved through analogical or transfer learning techniques, where a system applies the learning in one system to make recommendations in another domain. This can significantly assist in cold-start issues in the recommender systems too. Another possible enhancement in the recommender system is to augment the learning techniques with user metadata such as past purchase behavior and click-through rate. Lastly, sentiment analysis techniques can be applied to recommendations presented to the user for further analysis and refinement of future predictions.

# 9. Conclusions

This paper presented a systematic literature review of next-items recommender systems with the aim of providing an overview of the domain. The scope of the SLR was limited to studies published between 2017 and 2022 to target the most recent works. The search strategy included the major digital sources of publications including ACM Digital Library, IEEE Xplore, Science Direct, Scopus, Springer Link, and Web of Science. After applying the filtering criteria, a total of 90 studies were selected for review. We defined and addressed research questions related to the key aspects of the NIRSs including: (i) motivation and applications of NIRSs, (ii) use of conventional ML techniques for developing NIRSs, (iii) use of deep learning techniques in NIRSs, (iv) the commonly used datasets, (v) specialized metrics used for evaluating NIRSs, and (vi) the challenges and open problems related to NIRSs. Recommender systems involve a different set of challenges pertaining to the development and deployment phases. The scope of the current study was limited to investigating mainly the former type of challenges. Future SLRs may address the issues involved in the latter type such as reliability, scalability, and performance of NIRSs.

**Author Contributions:** Conceptualization, Q.M.I.; methodology, Q.M.I., A.M. and A.A.; formal analysis, Q.M.I., A.M. and A.A.; investigation, Q.M.I., A.M. and A.A.; resources, M.A.; data curation, Q.M.I., A.M. and A.A.; writing—original draft preparation, Q.M.I., A.M., A.A. and M.A.; project administration, Q.M.I.; funding acquisition, Q.M.I. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported through the Annual Funding track by the Deanship of Scientific Research, Vice Presidency for Graduate Studies and Scientific Research, King Faisal University, Saudi Arabia (Project No. AN000258).

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. Andry, J.F.; Riwanto, R.E.; Wijaya, R.L.; Prawoto, A.A.; Prayogo, T. Development Point of Sales Using SCRUM Framework. J. Syst. Integr. 2019, 10, 1804–2724.
- Koch, J.; Frommeyer, B.; Schewe, G. Online Shopping Motives during the COVID-19 Pandemic—Lessons from the Crisis. Sustainability 2020, 12, 10247. [CrossRef]
- Gorgoglione, M.; Panniello, U.; Tuzhilin, A. Recommendation strategies in personalization applications. *Inf. Manag.* 2019, 56, 103143. [CrossRef]

- 4. Akter, T. Online shopping behavior: An in-depth study on motivating and restraining factors. *Glob. J. Manag. Bus. Res.* **2018**, *18*, 19.
- Anwar, M.; Khan, S.Z.; Shah, S.Z.A. Big Data Capabilities and Firm's Performance: A Mediating Role of Competitive Advantage. J. Inf. Knowl. Manag. 2018, 17, 1850045. [CrossRef]
- Rosa, R.L.; Schwartz, G.M.; Ruggiero, W.V.; Rodriguez, D.Z. A Knowledge-Based Recommendation System That Includes Sentiment Analysis and Deep Learning. *IEEE Trans. Ind. Inform.* 2018, 15, 2124–2135. [CrossRef]
- Yochum, P.; Chang, L.; Gu, T.; Zhu, M. Linked Open Data in Location-Based Recommendation System on Tourism Domain: A Survey. *IEEE Access* 2020, *8*, 16409–16439. [CrossRef]
- Yu, F.; Liu, Q.; Wu, S.; Wang, L.; Tan, T. A dynamic recurrent model for next basket recommendation. In Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retriev-al, Pisa, Italy, 17–21 July 2016; pp. 729–732.
- Lee, A.J.; Wang, C.-S.; Weng, W.-Y.; Chen, Y.-A.; Wu, H.-W. An efficient algorithm for mining closed inter-transaction itemsets. Data Knowl. Eng. 2008, 66, 68–91. [CrossRef]
- 10. Peker, S.; Kocyigit, A.; Eren, P.E. A hybrid approach for predicting customers' individual purchase be-havior. *Kybernetes* **2017**, *46*. [CrossRef]
- Kitchenham, B.; Charters, S. Guidelines for Performing Systematic Literature Reviews in Software Engineering; EBSE Technical Report EBSE-2007-01; Keele University: Keele, UK, 2007.
- 12. Alyari, F.; Navimipour, N.J. Recommender systems. Kybernetes 2018, 47, 985–1017. [CrossRef]
- 13. Portugal, I.; Alencar, P.; Cowan, D. The use of machine learning algorithms in recommender systems: A systematic review. *Expert Syst. Appl.* **2018**, *97*, 205–227. [CrossRef]
- 14. Monti, D.; Rizzo, G.; Morisio, M. A systematic literature review of multicriteria recommender systems. *Artif. Intell. Rev.* 2020, 54, 427–468. [CrossRef]
- 15. 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]
- Murciego, L.; Jiménez-Bravo, D.; Román, A.V.; Santana, J.D.P.; Moreno-García, M. Context-Aware Recommender Systems in the Music Domain: A Systematic Literature Review. *Electronics* 2021, 10, 1555. [CrossRef]
- 17. Jesse, M.; Jannach, D. Digital nudging with recommender systems: Survey and future directions. *Comput. Hum. Behav. Rep.* **2021**, *3*, 100052. [CrossRef]
- 18. Da'U, A.; Salim, N. Recommendation system based on deep learning methods: A systematic review and new directions. *Artif. Intell. Rev.* **2019**, *53*, 2709–2748. [CrossRef]
- 19. Khan, H.R.; Lim, C.; Ahmed, M.; Tan, K.; Bin Mokhtar, M. Systematic Review of Contextual Suggestion and Recommendation Systems for Sustainable e-Tourism. *Sustainability* **2021**, *13*, 8141. [CrossRef]
- Hamid, R.A.; Albahri, A.; Alwan, J.K.; Al-Qaysi, Z.; Zaidan, A.; Alnoor, A.; Alamoodi, A.; Zaidan, B. How smart is e-tourism? A systematic review of smart tourism recommendation system applying data management. *Comput. Sci. Rev.* 2020, 39, 100337. [CrossRef]
- Mohammadi, V.; Rahmani, A.M.; Darwesh, A.M.; Sahafi, A. Trust-based recommendation systems in Internet of Things: A systematic literature review. *Hum.-Cent. Comput. Inf. Sci.* 2019, 9, 21. [CrossRef]
- 22. Rahayu, N.W.; Ferdiana, R.; Kusumawardani, S.S. A systematic review of ontology use in E-Learning recommender system. *Comput. Educ. Artif. Intell.* 2022, *3*, 100047. [CrossRef]
- 23. Wohlin, C.; Runeson, P.; Höst, M.; Ohlsson, M.C.; Regnell, B.; Wesslén, A. *Experimentation in Software Engineering*; Springer: Berlin/Heidelberg, Germany, 2012. [CrossRef]
- 24. Petticrew, M.; Roberts, H. Systematic Reviews in the Social Sciences: A Practical Guide; John Wiley & Sons: Hoboken, NJ, USA, 2008.
- Page, M.J.; McKenzie, J.E.; Bossuyt, P.M.; Boutron, I.; Hoffmann, T.C.; Mulrow, C.D.; Shamseer, L.; Tetzlaff, J.M.; Akl, E.A.; Brennan, S.E.; et al. The PRISMA 2020 statement: An updated guideline for reporting systematic reviews. *Int. J. Surg.* 2021, *88*, 105906. [CrossRef] [PubMed]
- Zhao, C.; You, J.; Wen, X.; Li, X. Deep Bi-LSTM Networks for Sequential Recommendation. *Entropy* 2020, 22, 870. [CrossRef] [PubMed]
- He, R.; Kang, W.-C.; McAuley, J. Translation-based Recommendation: A Scalable Method for Modeling Sequential Behavior. In Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence Best Sister Conferences, Stockholm, Sweden, 13–19 July 2018; pp. 5264–5268. [CrossRef]
- He, R.; Kang, W.-C.; McAuley, J. Translation-based Recommendation. In Proceedings of the Eleventh ACM Conference on Recommender Systems, Como, Italy, 27–31 August 2017; pp. 161–169. [CrossRef]
- 29. Huang, T.; Zhang, D.; Bi, L. Neural embedding collaborative filtering for recommender systems. *Neural Comput. Appl.* **2020**, *32*, 17043–17057. [CrossRef]
- Quadrana, M.; Cremonesi, P.; Jannach, D. Sequence-aware Recommender Systems. In Proceedings of the 26th Conference on User Modeling, Adaptation and Personalization, Singapore, 8–11 July 2018; pp. 373–374. [CrossRef]
- Chen, D.; Hu, W.; Yuan, B.; Zhang, R.; Wang, J. Next-Item Recommendation with Deep Adaptable Co-Embedding Neural Networks. *IEEE Signal Process. Lett.* 2021, 28, 1220–1224. [CrossRef]

- 32. Li, T.; Choi, M.; Fu, K.; Lin, L. Music Sequence Prediction with Mixture Hidden Markov Models. In Proceedings of the 2019 IEEE International Conference on Big Data, Los Angeles, CA, USA, 9–12 December 2019; pp. 6128–6132. [CrossRef]
- Mlika, F.; Karoui, W. Proposed Model to Intelligent Recommendation System based on Markov Chains and Grouping of Genres. Procedia Comput. Sci. 2020, 176, 868–877. [CrossRef]
- Zhang, Y.; Shi, Z.; Zuo, W.; Yue, L.; Liang, S.; Li, X. Joint Personalized Markov Chains with social network embedding for cold-start recommendation. *Neurocomputing* 2019, 386, 208–220. [CrossRef]
- 35. Chen, L.; Xia, M. A context-aware recommendation approach based on feature selection. Appl. Intell. 2020, 51, 865–875. [CrossRef]
- Zhu, J.; Ma, X.; Yue, C.; Wang, C. Interest-Forgetting Markov Model for Next-Basket Recommendation. *Commun. Comput. Inf. Sci.* 2019, 1058, 20–31. [CrossRef]
- Aghdam, M.H. Context-aware recommender systems using hierarchical hidden Markov model. *Phys. A Stat. Mech. Appl.* 2018, 518, 89–98. [CrossRef]
- Wang, P.; Zhang, Y.; Niu, S.; Guo, J. Modeling Temporal Dynamics of Users' Purchase Behaviors for Next Basket Prediction. J. Comput. Sci. Technol. 2019, 34, 1230–1240. [CrossRef]
- 39. Nasir, M.; Ezeife, C.I.; Gidado, A. Improving e-commerce product recommendation using semantic context and sequential historical purchases. *Soc. Netw. Anal. Min.* **2021**, *11*, 82. [CrossRef]
- 40. Wang, W.; Cao, L. Interactive Sequential Basket Recommendation by Learning Basket Couplings and Positive/Negative Feedback. *ACM Trans. Inf. Syst.* **2021**, *39*, 1–26. [CrossRef]
- Yağcı, A.M.; Aytekin, T.; Gürgen, F.S. Parallel pairwise learning to rank for collaborative filtering. *Concurr. Comput. Pract. Exp.* 2018, 31, e5141. [CrossRef]
- 42. De Maio, C.; Fenza, G.; Gallo, M.; Loia, V.; Parente, M. Social media marketing through time-aware collaborative filtering. *Concurr. Comput. Pract. Exp.* **2017**, *30*, e4098. [CrossRef]
- Sharma, R.; Gopalani, D.; Meena, Y. Collaborative filtering-based recommender system: Approaches and research challenges. In Proceedings of the 2017 3rd International Conference on Computational Intelligence & Communication Technology (CICT), Ghaziabad, India, 9–10 February 2017; pp. 1–6. [CrossRef]
- Zheng, L.; Noroozi, V.; Yu, P.S. Joint Deep Modeling of Users and Items Using Reviews for Recommendation. In Proceedings of the Tenth ACM International Conference on Web Search and Data Mining, Cambridge, UK, 6–10 February 2017. [CrossRef]
- Sharma, S.; Mahajan, A. Suggestive Approaches to Create a Recommender System for GitHub. Int. J. Inf. Technol. Comput. Sci. 2017, 9, 48–55. [CrossRef]
- 46. Raja, D.K.; Pushpa, S.; Raja, K. Novelty-driven recommendation by using integrated matrix factorization and temporal-aware clustering optimization. *Int. J. Commun. Syst.* **2018**, *33*, e3851. [CrossRef]
- 47. Issues in various recommender system in E-commerce—A survey. J. Crit. Rev. 2020, 7, 604–608. [CrossRef]
- Çano, E.; Morisio, M. Hybrid recommender systems: A systematic literature review. Intell. Data Anal. 2017, 21, 1487–1524. [CrossRef]
- 49. Kunaver, M.; Požrl, T. Diversity in recommender systems—A survey. Knowl.-Based Syst. 2017, 123, 154–162. [CrossRef]
- Hiriyannaiah, S.; Siddesh, G.M.; Srinivasa, K.G. DeepLSGR: Neural collaborative filtering for recommendation systems in smart community. *Multimed. Tools Appl.* 2022, 1–20. [CrossRef]
- Soliman, K.; Mahmood, M.A.; El Azab, A.; Hefny, H. A Survey of Recommender Systems and Geographical Recommendation Techniques. In GIS Applications in the Tourism and Hospitality Industry; IGI Global: Hershey, PA, USA, 2018; pp. 249–274. [CrossRef]
- 52. Wang, C.-D.; Deng, Z.-H.; Lai, J.-H.; Yu, P.S. Serendipitous Recommendation in E-Commerce Using Innovator-Based Collaborative Filtering. *IEEE Trans. Cybern.* **2018**, *49*, 2678–2692. [CrossRef] [PubMed]
- Satheesan, P.; Haddela, P.S.; Alosius, J. Product Recommendation System for Supermarket. In Proceedings of the 2020 19th IEEE International Conference on Machine Learning and Applications (ICMLA), Miami, FL, USA, 14–17 December 2020; pp. 930–935. [CrossRef]
- Usmani, Z.A.; Manchekar, S.; Malim, T.; Mir, A. A predictive approach for improving the sales of products in e-commerce. In Proceedings of the 3rd IEEE International Conference on Advances in Electrical and Electronics, Information, Communication and Bio-Informatics, AEEICB 2017, Chennai, India, 27–28 February 2017; pp. 188–192. [CrossRef]
- Jia, R.; Li, R.; Yu, M.; Wang, S. E-commerce purchase prediction approach by user behavior data. In Proceedings of the IEEE CITS 2017—2017 International Conference on Computer, Information and Telecommunication Systems, Dalian, China, 21–23 July 2017; pp. 1–5. [CrossRef]
- Kommineni, M.; Alekhya, P.; Vyshnavi, T.M.; Aparna, V.; Swetha, K.; Mounika, V. Machine Learning based Efficient Recommendation System for Book Selection using User based Collaborative Filtering Algorithm. In Proceedings of the 4th International Conference on Inventive Systems and Control, ICISC 2020, Coimbatore, India, 8–10 January 2020; pp. 66–71. [CrossRef]
- Biswas, A.; Vineeth, K.S.; Jain, A. Mohana Development of Product Recommendation Engine By Collaborative Filtering and Association Rule Mining Using Machine Learning Algorithms. In Proceedings of the 4th International Conference on Inventive Systems and Control, ICISC 2020, Coimbatore, India, 8–10 January 2020; pp. 272–277. [CrossRef]
- Verma, P.; Sharma, S. Artificial Intelligence based Recommendation System. In Proceedings of the IEEE 2020 2nd International Conference on Advances in Computing, Communication Control and Networking, ICACCCN 2020, Greater Noida, India, 18–19 December 2020; pp. 669–673. [CrossRef]

- Doan, T.-N.; Sahebi, S. TransCrossCF: Transition-based Cross-Domain Collaborative Filtering. In Proceedings of the 19th IEEE International Conference on Machine Learning and Applications, ICMLA 2020, Miami, FL, USA, 14–17 December 2020; pp. 320–327. [CrossRef]
- Li, C.; Ma, L. Item-based Collaborative Filtering Algorithm Based on Group Weighted Rating. In Proceedings of the 2020 13th International Symposium on Computational Intelligence and Design, ISCID 2020, Hangzhou, China, 12–13 December 2020; pp. 114–117. [CrossRef]
- Xiao, S.; Ling, H.; Lu, Y.; Tang, Z. Study on collaborative filtering recommendation algorithm based on prediction for item rating. In Proceedings of the 2020 International Conference on Information Science, Parallel and Distributed Systems, ISPDS 2020, Xi'an, China, 14–16 August 2020; pp. 352–355. [CrossRef]
- 62. Anwaar, F.; Iltaf, N.; Afzal, H.; Nawaz, R. HRS-CE: A hybrid framework to integrate content embeddings in recommender systems for cold start items. *J. Comput. Sci.* 2018, 29, 9–18. [CrossRef]
- Hwangbo, H.; Kim, Y.S.; Cha, K.J. Recommendation system development for fashion retail e-commerce. *Electron. Commer. Res. Appl.* 2018, 28, 94–101. [CrossRef]
- 64. Tewari, A.S. Generating Items Recommendations by Fusing Content and User-Item based Collaborative Filtering. *Procedia Comput. Sci.* **2020**, *167*, 1934–1940. [CrossRef]
- 65. Tang, Y.; Guo, K.; Zhang, R.; Xu, T.; Ma, J.; Chi, T. ICFR: An effective incremental collaborative filtering based recommendation architecture for personalized websites. *World Wide Web* **2019**, *23*, 1319–1340. [CrossRef]
- Jiang, M.; Zhang, Z.; Jiang, J.; Wang, Q.; Pei, Z. A collaborative filtering recommendation algorithm based on information theory and bi-clustering. *Neural Comput. Appl.* 2019, 31, 8279–8287. [CrossRef]
- Khoshahval, S.; Farnaghi, M.; Taleai, M.; Mansourian, A. A Personalized Location-Based and Serendipity-Oriented Point of Interest Recommender Assistant Based on Behavioral Patterns. In *Lecture Notes in Geoinformation and Cartography*; Springer: Berlin/Heidelberg, Germany, 2018; pp. 271–289. [CrossRef]
- 68. Yu, H.; Wang, Y.; Fan, Y.; Meng, S.; Huang, R. Accuracy Is Not Enough: Serendipity Should Be Considered More. In *Innovative Mobile and Internet Services in Ubiquitous Computing*; Springer: Berlin/Heidelberg, Germany, 2017; pp. 231–241. [CrossRef]
- 69. Gaikwad, R.S.; Udmale, S.S.; Sambhe, V.K. E-commerce Recommendation System Using Improved Probabilistic Model. In *Information and Communication Technology for Sustainable Development*; Springer: Singapore, 2017; pp. 277–284. [CrossRef]
- 70. Xiao, Y.; Ezeife, C.I. E-Commerce Product Recommendation Using Historical Purchases and Clickstream Data. In *Big Data Analytics and Knowledge Discovery*; Springer: Berlin/Heidelberg, Germany, 2018; pp. 70–82. [CrossRef]
- Bhatta, R.; Ezeife, C.I.; Butt, M.N. Mining Sequential Patterns of Historical Purchases for E-commerce Recommendation. In *Big* Data Analytics and Knowledge Discovery; Springer: Berlin/Heidelberg, Germany, 2019; pp. 57–72. [CrossRef]
- 72. 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, Perth, Australia, 3–7 April 2017; pp. 173–182.
- Zhu, T.; Liu, G.; Chen, G. Social Collaborative Mutual Learning for Item Recommendation. ACM Trans. Knowl. Discov. Data 2020, 14, 1–19. [CrossRef]
- 74. Kuo, R.; Chen, C.-K.; Keng, S.-H. Application of hybrid metaheuristic with perturbation-based K-nearest neighbors algorithm and densest imputation to collaborative filtering in recommender systems. *Inf. Sci.* **2021**, 575, 90–115. [CrossRef]
- Hu, H.; He, X.; Gao, J.; Zhang, Z.-L. Modeling Personalized Item Frequency Information for Next-basket Recommendation. In Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, 11–15 July 2020; pp. 1071–1080. [CrossRef]
- Gupta, G.; Katarya, R. A Study of Deep Reinforcement Learning Based Recommender Systems. In Proceedings of the 2021 2nd International Conference on Secure Cyber Computing and Communications (ICSCCC), Jalandhar, India, 21–23 May 2021; pp. 218–220. [CrossRef]
- Deldjoo, Y.; Di Noia, T.; Malitesta, D.; Merra, F.A. A Study on the Relative Importance of Convolutional Neural Networks in Visually-Aware Recommender Systems. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Nashville, TN, USA, 19–25 June 2021; pp. 3956–3962. [CrossRef]
- Haihan, L.; Guanglei, Q.; Nana, H.; Xinri, D. Shopping Recommendation System Design Based On Deep Learning. In Proceedings of the 2021 IEEE 6th International Conference on Intelligent Computing and Signal Processing, ICSP 2021, Xi'an, China, 9–11 April 2021; pp. 998–1001. [CrossRef]
- 79. Khan, Z.; Hussain, M.I.; Iltaf, N.; Kim, J.; Jeon, M. Contextual recommender system for E-commerce applications. *Appl. Soft Comput.* **2021**, *109*, 107552. [CrossRef]
- 80. Kavitha, K.; Kumar, S.L.; Pravalika, P.; Sruthi, K.; Lalitha, R.; Rao, N.K. Fashion compatibility using convolutional neural networks. *Mater. Today Proc.* 2020. [CrossRef]
- 81. Reyes, L.J.P.; Oviedo, N.B.; Camacho, E.C.; Calderon, J.M. Adaptable Recommendation System for Outfit Selection with Deep Learning Approach. *IFAC-PapersOnLine* **2021**, *54*, 605–610. [CrossRef]
- Yuan, F.; Karatzoglou, A.; Arapakis, I.; Jose, J.M.; He, X. A Simple Convolutional Generative Network for Next Item Recommendation. In Proceedings of the 12th ACM International Conference on Web Search and Data Mining, Melbourne, Australia, 11–15 February 2019; pp. 582–590. [CrossRef]
- Hidasi, B.; Karatzoglou, A.; Baltrunas, L.; Tikk, D. Session-based Recommendations with Recurrent Neu-ral Networks. *arXiv* 2015, arXiv:1511.06939.

- 84. Choe, B.; Kang, T.; Jung, K. Recommendation System with Hierarchical Recurrent Neural Network for Long-Term Time Series. *IEEE Access* 2021, *9*, 72033–72039. [CrossRef]
- 85. Rabiu, I.; Salim, N.; Da'U, A.; Nasser, M. Modeling sentimental bias and temporal dynamics for adaptive deep recommendation system. *Expert Syst. Appl.* **2021**, *191*, 116262. [CrossRef]
- Wang, P.; Li, Z.; Zhang, Y.; Hou, Y.; Ge, L. QPIN: A Quantum-inspired Preference Interactive Network for E-commerce Recommendation. In Proceedings of the International Conference on Information and Knowledge Management, Beijing, China, 3–7 November 2019; pp. 2329–2332. [CrossRef]
- Lo, C.; Yu, H.; Yin, X.; Shetty, K.; He, C.; Hu, K.; Platz, J.M.; Ilardi, A.; Madhvanath, S. Page-level Optimization of e-Commerce Item Recommendations. In Proceedings of the RecSys 2021—15th ACM Conference on Recommender Systems, Amsterdam, The Netherlands, 27 September–1 October 2021; pp. 495–504. [CrossRef]
- Han, J.; Ma, Y.; Mei, Q.; Liu, X. DeepRec: On-device Deep Learning for Privacy-Preserving Sequential Recommendation in Mobile Commerce. In Proceedings of the World Wide Web Conference, WWW 2021, Ljubljana, Slovenia, 19–23 April 2021; pp. 900–911. [CrossRef]
- Louis, M.; Azad, Z.; Delhadtehrani, L.; Gupta, S.L.; Warden, P.; Reddi, V.; Joshi, A. Towards Deep Learning using TensorFlow Lite on RISC-V. In Proceedings of the Third Workshop on Computer Architecture Research with RISC-V (CARRV 2019), Phoenix, AZ, USA, 22 June 2019. [CrossRef]
- 90. Liu, W.; Zhang, Y.; Wang, J.; He, Y.; Caverlee, J.; Chan, P.P.K.; Yeung, D.S.; Heng, P.-A. Item Relationship Graph Neural Networks for E-Commerce. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, 1–15. [CrossRef]
- Wong, C.-M.; Feng, F.; Zhang, W.; Chen, H.; Zhang, Y.; He, P.; Chen, H.; Zhao, K. Improving Conversational Recommender System by Pretraining Billion-scale Knowledge Graph. In Proceedings of the 2021 IEEE 37th International Conference on Data Engineering (ICDE), Chania, Greece, 19–22 April 2021; pp. 2607–2612. [CrossRef]
- Tao, Y.; Wang, C.; Yao, L.; Li, W.; Yu, Y. Item trend learning for sequential recommendation system using gated graph neural network. *Neural Comput. Appl.* 2021, 1–16. [CrossRef]
- 93. Zhang, Y.; Yuan, M.; Zhao, C.; Chen, M.; Liu, X. Integrating label propagation with graph convolutional networks for recommendation. *Neural Comput. Appl.* 2022, 34, 8211–8225. [CrossRef]
- Zhao, K.; Zheng, Y.; Zhuang, T.; Li, X.; Zeng, X. Joint Learning of E-commerce Search and Recommendation with a Unified Graph Neural Network. In Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining, Melbourne, Australia, 11–15 February 2022; pp. 1461–1469. [CrossRef]
- 95. Wang, H.; Amagata, D.; Makeawa, T.; Hara, T.; Hao, N.; Yonekawa, K.; Kurokawa, M. A DNN-Based Cross-Domain Recommender System for Alleviating Cold-Start Problem in E-Commerce. *IEEE Open J. Ind. Electron. Soc.* 2020, 1, 194–206. [CrossRef]
- 96. Ahmed, A.; Saleem, K.; Khalid, O.; Rashid, U. On deep neural network for trust aware cross domain recommendations in E-commerce. *Expert Syst. Appl.* **2021**, *174*, 114757. [CrossRef]
- Abinaya, S.; Devi, M.K.K. Enhancing Top-N Recommendation Using Stacked Autoencoder in Context-Aware Recommender System. *Neural Process. Lett.* 2021, 53, 1865–1888. [CrossRef]
- Qin, Y.; Wang, P.; Li, C. The World is Binary: Contrastive Learning for Denoising Next Basket Recommendation. In Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Montreal, QC, Canada, 12–15 July 2021; pp. 859–868. [CrossRef]
- 99. Ngaffo, A.N.; Choukair, Z. A deep neural network-based collaborative filtering using a matrix factorization with a twofold regularization. *Neural Comput. Appl.* 2022, 34, 6991–7003. [CrossRef]
- 100. Liu, T.; He, Z. DLIR: A deep learning-based initialization recommendation algorithm for trust-aware recommendation. *Appl. Intell.* **2022**, 1–12. [CrossRef]
- Chorowski, J.; Bahdanau, D.; Serdyuk, D.; Cho, K.; Bengio, Y. Attention-Based Models for Speech Recognition. In Proceedings of the 29th Annual Conference on Neural Information Processing Systems 2015, Montreal, QC, Canada, 7–12 December 2015.
- 102. Zhang, J.; Wang, D.; Yu, D. TLSAN: Time-aware long- and short-term attention network for next-item recommendation. *Neurocomputing* **2021**, *441*, 179–191. [CrossRef]
- Thaipisutikul, T.; Shih, T.K. A novel context-aware recommender system based on a deep sequential learning approach (CReS). Neural Comput. Appl. 2021, 33, 11067–11090. [CrossRef]
- 104. Che, B.; Zhao, P.; Fang, J.; Zhao, L.; Sheng, V.S.; Cui, Z.; Sheng, V. Inter-Basket and Intra-Basket Adaptive Attention Network for Next Basket Recommendation. *IEEE Access* 2019, 7, 80644–80650. [CrossRef]
- 105. Liu, T.; Yin, X.; Ni, W. Next Basket Recommendation Model Based on Attribute-Aware Multi-Level Attention. *IEEE Access* 2020, *8*, 153872–153880. [CrossRef]
- 106. Da'U, A.; Salim, N. Sentiment-Aware Deep Recommender System With Neural Attention Networks. *IEEE Access* 2019, 7, 45472–45484. [CrossRef]
- 107. Ouyang, S.; Lawlor, A. Improving Explainable Recommendations by Deep Review-Based Explanations. *IEEE Access* 2021, *9*, 67444–67455. [CrossRef]
- Cui, Q.; Wu, S.; Huang, Y.; Wang, L. A hierarchical contextual attention-based network for sequential recommendation. *Neurocomputing* 2019, 358, 141–149. [CrossRef]
- Zhang, L.; Wang, P.; Li, J.; Xiao, Z.; Shi, H. Attentive Hybrid Recurrent Neural Networks for sequential recommendation. *Neural Comput. Appl.* 2021, 33, 11091–11105. [CrossRef]

- 110. Li, W.; Xu, B. Aspect-Based Fashion Recommendation With Attention Mechanism. IEEE Access 2020, 8, 141814–141823. [CrossRef]
- 111. Du, Y.; Wang, L.; Peng, Z.; Guo, W. Review-based hierarchical attention cooperative neural networks for recommendation. *Neurocomputing* **2021**, 447, 38–47. [CrossRef]
- Zhu, H.; Li, X.; Zhang, P.; Li, G.; He, J.; Li, H.; Gai, K. Learning Tree-based Deep Model for Recommender Systems. In Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, London, UK, 19–23 August 2018; pp. 1079–1088. [CrossRef]
- 113. Harper, F.M.; Konstan, J.A. The MovieLens Datasets: History and context. *ACM Trans. Interact. Intell. Syst.* 2015, *5*, 1–19. [CrossRef]
- 114. Singer, U.; Roitman, H.; Eshel, Y.; Nus, A.; Guy, I.; Levi, O.; Hasson, I.; Kiperwasser, E. Sequential Modeling with Multiple Attributes for Watchlist Recommendation in E-Commerce. In Proceedings of the 15th ACM International Conference on Web Search and Data Mining, Virtual, 21–25 February 2022; pp. 937–946. [CrossRef]
- 115. 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]
- 116. Ni, J.; Li, J.; McAuley, J. Justifying Recommendations using Distantly-Labeled Reviews and Fine-Grained Aspects. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Hong Kong, China, 3–7 November 2019; pp. 188–197.
- 117. Zhang, M.-L.; Zhou, Z.-H. A review on multi-label learning algorithms. *IEEE Trans. Knowl. Data Eng.* **2013**, *26*, 1819–1837. [CrossRef]
- Jannach, D.; Lerche, L.; Jugovac, M. Adaptation and Evaluation of Recommendations for Short-term Shopping Goals. In Proceedings of the 9th ACM Conference on Recommender Systems, Vienna, Austria, 16–20 September 2015; pp. 211–218. [CrossRef]
- 119. Castells, P.; Vargas, S.; Wang, J. Novelty and Diversity Metrics for Recommender Systems: Choice, Discovery and Relevance; Universidad Autónoma de Madrid: Madrid, Spain, 2011.
- 120. Breese, J.S.; Heckerman, D.; Kadie, C. Empirical Analysis of Predictive Algorithms for Collaborative Fil-tering. *arXiv* 2013, arXiv:1301.7363.