Education and Information Technologies (2020) 25:2635–2664
https://doi.org/10.1007/s10639-019-10063-9

A systematic review: machine learning based recommendation systems for e-learning

Shristi Shakya Khanal[1] · P.W.C. Prasad[1] · Abeer Alsadoon[1] · Angelika Maag[1]

Received: 15 April 2019 / Accepted: 12 November 2019 / Published online: 14 December 2019

Abstract

The constantly growing offering of online learning materials to students is making it more difficult to locate specific information from data pools. Personalization systems attempt to reduce this complexity through adaptive e-learning and recommendation systems. The latter are, generally, based on machine learning techniques and algorithms and there has been progress. However, challenges remain in the form of data-scarcity, cold-start, scalability, time consumption and accuracy. In this article, we provide an overview of recommendation systems in the e-learning context following four strands: Content-Based, Collaborative Filtering, Knowledge-Based and Hybrid Systems. We developed a taxonomy that accounts for components required to develop an effective recommendation system. It was found that machine learning techniques, algorithms, datasets, evaluation, valuation and output are necessary components. This paper makes a significant contribution to the field by providing a much-needed overview of the current state of research and remaining challenges.

Keywords Recommendation system · Recommender · E-learning · Content-based · Collaborative filtering · Hybrid

1 Introduction

Recommendation systems (RS) belong to the category of information filtering systems designed to predict the preference users would give to a topic. They mimic the social action of relying on recommendations from others and augment the process further (Resnick and Varian 1997). Recommender systems. *Communications of the ACM, 40*(3), 56–59.Initially, they were used as a 'digital' bookshelf in research (Karlgren, in Gupta and Pandey 2019). Later, a 'collaborative filtering' engine, *Tapestry,* was

✉ P.W.C. Prasad
cwithana@studygroup.com

[1] Charles Sturt University, Sydney Campus, 63, Level 1, Oxford Street, Darlinghurst, Sydney, NSW 2010, Australia

developed in the early 1990s by Goldberg et al. (1992), mainly for commercial applications. The name suggested that a collaboration exists between recommenders and requesters, which rejected later on the grounds that this causes confusion as the parties are not known to each other (among others, Gupta and Pandey 2019). RS has significantly evolved since then, being now also extensively used to adapt learning experiences to the goals of individual learner (Klašnja-Milićević et al. 2017; Marinho et al. 2012; Herlocker et al. 2004).

In an e-learning context, RS can provide suggestions for familiarization activities prior to attempting learning modules. RS delivers personalized recommendations for learning materials based on individuals' learning interests and learning path in online learning systems and forms 'user rating resource matrices' (Xiao et al. 2017; Marinho et al. 2012; Herlocker et al. 2004; Klašnja-Milićević et al. 2017; Marinho et al. 2012; Herlocker et al. 2004).

This paper provides a systematic review of current recommendation approaches used in e-learning along with machine learning algorithms utilized in RS. The focus is on generating classifications of recommendation techniques, machine learning techniques and algorithms used in RS, methods applied, areas of application, datasets, validation and evaluation approaches, and input/output data in order to identify research contributions and limitations.

To identify relevant papers, we used the following search terms: "e-learning recommendation system", "e-learning recommender", adaptive e-learning", "recommendation system", and "e-learning". We identified 35 papers as relevant between 2016 and 2018 from Q1 and Q2 journals. Criterion for this first stage of the selection process was simply the existence of a recommendation system. For stage two, the criteria were: 'e-learning', 'novel machine learning algorithm', and 'aims' that address one of the remaining challenges for RS (i.e. cold-start etc.). 25 papers were excluded because they did not meet any one of the stage-2 criteria. The remaining ten papers had proposed an RS system for e-learning based using one of the RS techniques – CB, CF, Hybrid systems or those that are Knowledge-Based, although they did not meet the other two criteria. The rationale for this 2-stage selection approach had been that RS from other domains would still have been of interest had they proposed novel algorithms or addressed one of the remaining challenges.

Inclusion and Exclusion criteria: The inclusion criteria considered for selecting articles for review are:

1. Papers proposing recommendation techniques for e-learning platforms.
2. Papers proposing RS using machine learning techniques.
3. Papers presenting evaluation for the proposed technique.
4. Papers from Q1 and Q2 journals and their state-of-the-art sources.
5. Papers focusing on personalization and adaptive e-learning.

Exclusion criteria were: 1. Papers that were not written in English;
2. Papers to which there was no full access;
3. Papers which lacked detailed explanation and evaluation of the topic.

In Sect. 2, the literature on e-learning recommendation systems is reviewed along with background on related topics. Section 3 covers classification, discussion and future challenges. Section 4 provides the conclusion.

2 Literature Review

2.1 Background of terminologies discussed

We provide an overview of five main research areas related to the topic: recommendation techniques, data used in RS, machine learning techniques in RS, recommendation system evaluation and architecture of Hybrid recommendation system.

2.1.1 Recommendation techniques

There are three recommendation approaches – *Collaborative filtering (*CF*), Content-based recommendation (CB) and Hybrid recommendation system*. (Adomavicius and Tuzhilin 2005).

Content based (CB) Content-Based recommendation is based on identifying characteristics that are like those a user has preferred in the past and make recommendations accordingly.

Figure 1 below shows the architecture of CB RS which has three components – a Content analyzer, a Profile Learner (Generator) and a Filtering Component (Narducci et al. 2015) The Content Analyzer collects the content of items from different information sources and extracts item representations using feature extraction techniques. The Profile Learner applies machine learning techniques to the item representation and generalizes the user data to create the user profile based on past likes and dislikes. In a last step, the filtering component matches the user profile with items to be recommended (Narducci et al. 2015, pp. 81–82).

Collaborative filtering (CF) Collaborative filtering recommendation is based on user behavior or user ratings of recommended items. It recommends items liked by similar users and explores diverse possible content (Han et al. 2016; Kim et al. 2013; Smys and Bala 2012). By accessing a learner profile, RS can access information about age, country, previous learning activities, educational background, etc. With the help of this information, RS can find learners with similar learning preferences and suggest learning materials accordingly (Yera and Martínez 2017). The CF algorithm finds either prediction ratings or recommends a list of top-N items as shown in Fig. 2.

As shown in Fig. 2, (a), the primary actor of a CF system is the active user who requires a rating prediction. Here, correlation is deduced by using past preferences among users. Based on this, CF recommender will generate recommendations to (a) based on the preferences of congruent users. Basically, a CF system will have a list of (m) users $U = \{u1, u2, \ldots, um\}$ and (n) items $P = \{p1, p2, \ldots, pn\}$. The system formulates an mXn user-item matrix incorporating user ratings per item where user

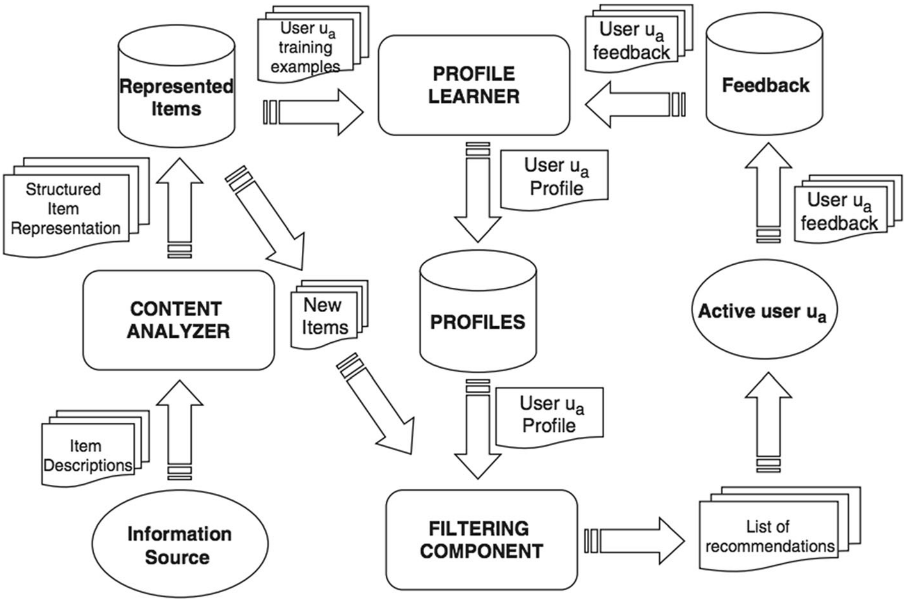

Fig. 1 High level architecture of content-based recommender. Source: Narducci, Musto, Polignano, Gemmis, Lops, & Semeraro (2015, pp. 81–82)

entry ri, j refers to the rating given by user ui for item pj. In order to acquire a recommendation for (a) for the target item (q), the CF algorithm will predict a rating for (q) or generate a list of most preferred top-N items.

To overcome recommendation problems, the CF uses memory-based and model-based algorithms, whereby the former uses an entire database to extract information. This can be a relatively slow process which is why model-based algorithms are often preferred. They are faster as they extract only selected items from which they build a model. Aggarwal (2016a, b) has argued that “Memory-based algorithms can be user-based, item-based, or hybridized. While past preferences of nearest neighbors to ‘a’ are employed in the user-based CF, the ratings of similar items to ‘q’ are used in an item-based approach” (p. 37). “Model-based algorithms aim to build an offline model by applying machine learning and data mining techniques. Building and training such

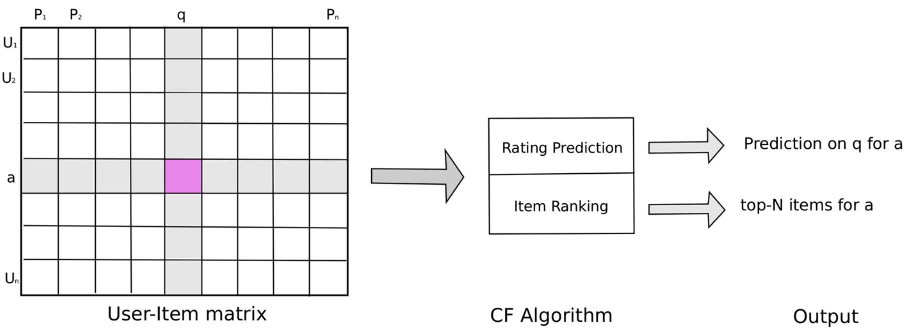

Fig. 2 CF based recommendation system. Source: Batmaz et al. (2018, p. 4)

model allows estimating predictions for online CF tasks. Model-based CF algorithms include Bayesian models, clustering models, decision trees, and singular value decomposition models" (Su and Khoshgoftaar 2009, p. 12).

Hybrid recommendation Hybrid RS is the combination of CB and CF which combines characteristics of both approaches through mergers of individual predictions into one or adding content information to collaborative model or by weighted average of content and collaborative recommendations or getting final recommendations based on the combined rankings.

Knowledge-based recommendation Some recommendation systems used in e-learning are Semantics- or Knowledge-Based. They include Context-Based and Ontology-Based approaches (Felfernig et al. 2018, p. 302). systems are knowledge-based and frame knowledge of content and about stakeholders of the recommendation process through ontology. For e-learning that means that such systems map learner-relevant learning resources through the exploitation of relational knowledge (Tarus et al. 2017).

All recommender systems (RS) are based on one of the above architectures, are present in all recommenders and constitute the framework within which specific recommendation tasks are addressed. We pose that, to be complete, they must utilize datasets, machine learning techniques, and algorithms, be subject to evaluation and valuation and have output. In the following chapters, we discuss each of these components in depth.

2.2 Data used in recommendation systems

Recommendation systems are data reliant, collected implicitly or explicitly. Implicit data are raw data and fall into two categories, those deliberately gathered from available data streams (i.e. search history, user clicks and keystroke logs) and 'exhaust data' (Kitchin 2014, p. 6) which are by-products of user activity that may or may not be used. Explicit data are collected through registration forms or profile information provided by users. They can also be collected from user ratings provided online (Huang and Huang 2015; Kim and Ahn 2008; Al-Shamri and Bharadwaj 2008). Together, these data form the input for models, generated through machine learning (ML) techniques, that predict preferences of users (Horváth and Carvalho 2017), although accuracy of the latter clearly depends on data quality and volume. In terms of machine learning techniques for the processing of such data, most current recommender systems are based on deep learning architectures that "provide a powerful framework for supervised learning" (Goodfellow et al. 2016, p.163).

2.3 Deep learning in recommendation systems

Goodfellow et al. (2016) identified "core parametric function approximation" (p.163) as the core technology on which most current feedforward deep networks (FDN) are based. The authors equate the term 'feedforward' to network input being propelled from function to function while being computationally assessed for its approximation to a target. FDN are building blocks for many of the modern neural networks, including

Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), Deep Belief Networks (DBN) and others. These systems are now widely employed in different industries, mainly for object and speech recognition and natural language procession (NLP) but also for advertising and recommender systems (You et al. 2019). In such cases, the aim is to approximate an item or information to a user (Goodfellow et al. 2016). A recent work on RS by You et al. (2019) combined RNN with a novel Temporal Neural Network (HierTCN) to permit hierarchical analysis of user preferences and match results to recent information to capture the historical impact on recent user decisions.

According to Goodfellow et al. (2016, p. 483), the best results, statistically and computationally, have been obtained with supervised learning, based on supervised classification and regression, using sets of labels which are known and techniques such as KNN, decision trees, rules, Bayesian classifiers and logistic regression. Unsupervised classification extracts labels from association rules or clustering techniques, such as k-means, density or hierarchy-based metrics, message parsing and Bayesian non-parametrics (Amatriain et al. 2010). In terms of the latter, Goodfellow et al. 2016) claim that, despite the development of a significant number of algorithms to overcome the issue of "high dimensionality of random variables" (p. 483), unsupervised learning is still in its infancy. Nevertheless, both training methods are in current use in the design of Recommender Systems.

2.4 Recommendation system evaluation

The above discussion has demonstrated that there is a significant offering of potentially suitable architectures and algorithms, whenever designers introduce a new recommendation system to an application. Such decisions are made based on experiments, performance and evaluation metrics that provide rankings for possible algorithms (Aggarwal 2016a, b, Ch.7.1).There are three types of experiments for RS evaluation – online, offline and user studies. Offline experiments use existing data sets and protocols that replicate user activities and measure accuracy of the prediction. User studies are based on the use of an RS system by a limited number of users and their feedback. Online experiments evaluate the real time online use of RS systems (Aggarwal 2016a, b, Ch.7.1).

Choice of systems and algorithm is based on evaluation metrics. Most commonly used evaluation metrics Recall, Precision, Root mean squared error (RMSE), Mean absolute error (MAE) and F- measures (Bourkoukou et al. 2016, p. 613). Recall evaluates user preference-based RS recommendations - the larger the recall, the more precise the recommendation - while precision measures the percentage of user likes in terms of all recommendations. Thus, high precision equates to quality recommendations. RMSE measures the error in the predicted ratings which ideally should be low (Bhaskaran and Santhi 2017; Krithika and Lakshmi Priya 2016; Riahi 2015).

The foregoing provides a broad overview of current recommendation approaches for e-learning systems and deep learning architectures. Success of a recommender system depends on training methods (supervised or unsupervised), an innovative approach to the design of algorithms that breaks the 'box-like' nature of CNN (You et al. 2019, p. 2237), the area of application, accuracy of datasets, and the validation and evaluation approaches. Section 2.5 (below) provides a detailed analysis of how recommendation systems are currently used in the field.

2.5 Related works

Current research has been segmented by approach to recommendation into Content-Based, Collaborative Filtering, Hybrid Solutions and Knowledge-Based.

2.5.1 Research based on content-based (CB) recommender systems

Content-Based recommendations are based on known past preferences of users from which recommendations are made to users with similar likes and dislikes. For the education sector, Chen et al. (2016) have used correlation analysis to group learning courses. They developed three categories based on a 'rule-space' model following CB to obtain learning objects for each skill group and optimize the learning path order for each learner (see also Xin et al. 2004; Wang et al. 2015). Similarly, Kolekar et al. (2018) developed a rule-based adaptive user interface which provides learning components and suggests learning objects to learners, based on a set of rules (see also Mamat and Yusof 2013; Fouad 2012; Kolekar et al. 2018).

Shu et al. (2017) utilized historical student data, processed for predictions about learning materials using a CB recommendation algorithm (see also Tian et al. 2016; Goldstein and Osher 2009). Rahman and Abdullah (2018), also, based their system on a fuzzy clustering technique and decision tree to classify learners into beginner, intermediate and master based on their academic record and learning behaviors. This enables RS to recommend learner-centered materials web-based on an e-learning platform. Pereira et al. (2018) developed a new method for using social network interaction and linked data using processes of the CB approach to recommend appropriate online resources (see also Dwivedi and Bharadwaj 2012; Wang and Wu 2011).

The most common learning algorithms such as rule-based and fuzzy-based clustering techniques are based on probabilistic methods, relevance feedback and k-nearest neighbors. The limitation of the CB approach is that it relies on past user experience and is not able to recommend new material which may reduce user motivation and lead to an undesirably narrow focus. This indicates that, on its own, the content-based approach cannot overcome problems of data-scarcity, cold-start, scalability, time consumption and accuracy.

2.5.2 Research based on CF

One of the most popular recommendation techniques is CF and a significant number of recommender systems have been developed based on this technique as it overcomes the limitations of the CB technique (Zhou et al. 2018; Wang and Li 2017; LeCun et al. 2015; Rapečka and Dzemyda 2015). Liu (2017) proposed collaborative filtering based on the influence of e-learning group behavior which increases the accuracy of the recommendation even in the case of data sparsity (see also Nocera and Ursino 2011; Han et al. 2016). Also based on the CF recommendation paradigm, Yera and Martínez (2017) proposed a problem matrix and noise management techniques that generate precise neighborhood and accurate recommendations (see also Wang and Yang 2012; Winoto et al. 2012).

Collaborative filtering is applied with sequential pattern mining to develop a score model to collect learners' feedback as well as extract preferences to weigh learning objects and generate the most appropriate ones in the RS developed by Bourkoukou et al. (2016) (see also, Imran et al. 2015; Xu 2013; Xie et al. 2014). Similarly, Fatahi et al. (2017) have used sequence behavior patterns to predict the learning style of users (see also Truong 2016; Latham et al. 2012).

El-Bishouty et al. (2018) have followed the CF approach using a k-mean algorithm for unsupervised model learning to extract the learning sequence and map learning objects according to learner style. (see also Feldman et al. 2014; Abdullah et al. 2015; Latham et al. 2013). Guo et al. (2018) have implemented a new initiative in CF based on trust relationships of target users with neighbors to obtain the recommendation list using a user-item rating matrix and harmonic parameters (see also Guo et al. 2016; Wang et al. 2016). However, model-based factorization matrices do not easily integrate context with models. To overcome this issue, Kim and Yoon (2014) have proposed a multi-dimensional trust model based on tensor factorization that considers learners' history, behavior-based rating and viewing pattern to share online materials (see also Jeong et al. 2013; Ekstrand 2011). Dwivedi et al. (2017) based their work on a genetic algorithm following the CF approach to generate an optimized learning path for learners using a variable length genetic algorithm (see also Kamsa et al. 2016). Zhang et al. (2018) introduced deep belief networks to use with massive open online courses to enhance learners' efficiency using prediction ratings (see also Salama et al. 2010; Qiao et al. 2015; Holtz et al. 2018).

The limitation of the CF approach is that it is difficult to attach attributes to items which may lead to vague recommendations. It is also time consuming for items to get enough ratings to provide accurate recommendations, which is also due to issues such as cold-start, scalability and data sparsity. On its own, CF is, therefore, not able to function as a satisfactory component of an efficient recommender system.

2.5.3 Research based on hybrid solutions

Experiments have been conducted with combinations of CB and CF which consider learner preference, background, interests and memory capacity to store information that can counter the cold start problem by adding a recommendation module and using only explicit data. Benhamdi et al. (2016) evaluated multi-dimensional similarity of not only prior knowledge and learner interest but also time spent on different tests and memory capacity using a co-relation matrix and prediction ratings. Addition of memory capacity makes recommendations more accurate. This system not only recommends the learning material but also learning activities for learners (see also Ronghuai et al. 2012). However, if a newly added resource is not accessed or rated by learners, then it will still be an issue and there still exists the problem of data sparsity and scalability of the algorithm.

Hussain et al. (2018) used an artificial neural network, decision tree, logistic regression and support vector machine (SVM) to predict the difficulty that students face in subsequent sessions of a digital design course (see also Hlosta et al. 2017; Vahdat et al. 2015; Donzellini and Ponta 2007). Using an identical technique, Li et al.

(2018) identified personalized teaching ressources by describing user interests (see also Zhang et al. 2017; Chen et al. 2015; Huang et al. 2016).

Karga and Satratzemi (2017) used a similarity matrix, as well as a 'tag and ratings' based approach combining CF and CB to mobilize pre-existing learning designs (LD) and then assist teachers in re-designing the LD process in accordance with their needs and preferences (see also Clements et al. 2015; Bennett et al. 2015). This technique overcomes the disadvantages of CB and CF and it also appears to be more accurate when few ratings are available. All these advantages make the hybrid technique the best current solution even if it is more time consuming and costly.

2.5.4 Research based on knowledge

"Knowledge-based systems recommend items based on specific domain knowledge about how certain item features meet users' needs and preferences and, ultimately, how the item is useful for the user" (Ning et al. 2015, p. Ch.4). Aeiad and Meziane (2018) have applied an ontology-based model using dependency ratios and parse trees to produce accurate learning material according to learners' requirements (see also Phobun and Vicheanpanya 2010; Latham et al. 2014). Tarus et al. (2017) also used an ontology-based approach relying on classification and analysis to propose the best techniques for use in RS. (see also Manouselis et al. 2010; Najafabadi and Mahrin 2015).

Wan and Niu (2018) used a knowledge-based approach based on self-organization theory and a leaner model to propose learning objects. It overcomes cold start but is time consuming due to multiple layers of algorithms (see also Wan and Niu 2016; Aher and Lobo 2013). Trifa et al. (2018) used semantic correlation and a dynamic key value memory network based on a knowledge tracing agent to improvise the RS. (see also Salonen and Karjaluoto 2016; Piech et al. 2015). Nitchot et al. (2018) relied on knowledge representation using logic, ontology and computation to construct knowledge structures and recommend links of study material. (see also Chiou et al. 2012; Yang and Sun 2013). Overall, the knowledge-based approach is also time consuming and expensive as it consists of multiple layers, methods and algorithms not suitable for all scales of e-learning systems.

This demonstrates that each of the four approaches can contribute significantly in one or more area of recommendation technique. As discussed earlier, CB and CF on their own do not have capabilities to overcome cold start, scalability and scarcity issues. Hybrid solutions, however, have demonstrated that they do. Some are even able to provide a certain capacity for scalability, opening possibilities for future research which also should include a reduction in time consumption and expense.

2.6 Architecture of hybrid recommendation systems

The foregoing evaluation of RS has clearly identified that the Hybrid solution is the most appropriate based on its ability to overcome the issues that exist in both CB and CF methods if applied in isolation. By utilizing both methods, the advantages of each are exploited, which mitigates cold start and data sparsity issues. The architecture of the Hybrid solution is described in Fig. 3 below.

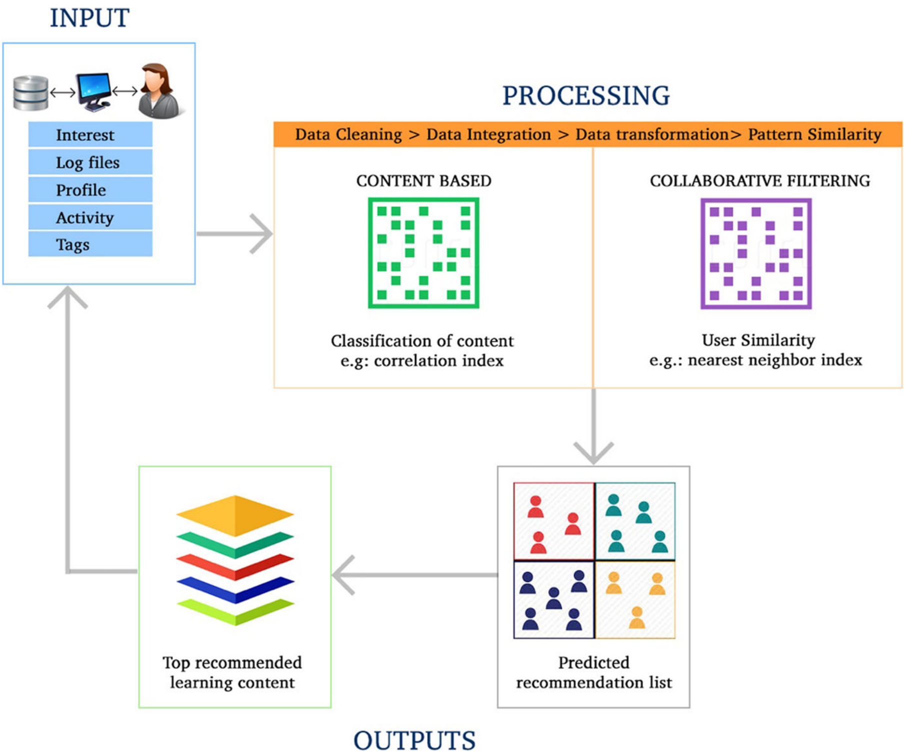

Fig. 3 System architecture of Hybrid recommendation system

The first module is the input or data collection module where the data from users are collected either from the learners' history log in the database or through the registration form completed by users; hence, data may be implicit and explicit. Data can also be obtained from online activity of learners through data mining techniques extracting their interests, preferences and learning patterns. This identifies student performance and measures if student is achieving their learning goals. Students' learning style can also be collected through test assessments which grade students from 0 to 25 in each learning style. To make accurate recommendations to learners, data collection should be as specific as possible in this module. Students can also rate the learning contents on a scale of 1–5. These ratings are taken as a basis for implementing CF algorithms.

The second recommendation processing module is the main module where collected data are filtered and cleaned removing unnecessary data or noise. Then the content-based engine analyzes item descriptions according to user preferences using machine learning algorithms as shown earlier in Fig. 1. This identifies items with high levels of similarity from descriptions stored in the user profile from which the recommendation list is produced. On the other hand, collaborative filtering builds the data set based on user preferences and generates a similarity matrix or nearest neighbor index using a range of machine learning algorithms to match active users with similar others and their preferences, as shown in Fig. 2. A Pearson correlation coefficient may be applied to find out the correlation between the users.

In the output module, results of CB and CF are compared and based on data sparsity, the list may be selected as correlation for the second technique to generate the recommendation list. Thus, users will be given output with learning content such as learning objects, materials, activities or learning paths that reside on the top N recommendation list.

Since both CB and CF engines are applied, this resolves the cold start and the data sparsity issue to some extent, so that there are no problems when a new item is added to the system or a new user joins the system. Those are the three special cases handled by the hybrid recommendation system. When a new learner is added, their rating is not available in the system. In this case, the system will consider the recommendation value of the instructor and ratings collected from similar users with similar learning styles. Hence, accurate prediction can be provided even when minimum ratings are available in the system. Another case is when new learning content is added to the system. This will also initially lack ratings, although those activities consist of meta-data like learning difficulty, author, version, etc. The CB algorithms, thus, utilizes the instructor's recommendations. Later, when the user ratings are received, the CF technique is used for more accurate results. The third scenario is that the Hybrid system detects sparsity of data which is the case for a high number of users and learning materials in any system. This problem occurs when only a small amount of learning material has been rated. In such cases, student ratings are collected as they are engaged in learning activities, assessments or performance evaluations as they go along using the online system. Thus, ratings of materials are collected along with calculations of similar paths followed by different learners.

These processes are mapped out in Fig. 3 below which describes the architecture of the hybrid system.

Hybrid systems, thus, offer significant advantages, although CB and CF, particularly, are still widely used on their own in different combinations (i.e. neighbor-based or classifier-based). There are, however, attributes and instances that are common to most recommender systems (Table 1). Other commonalities come from the challenges these systems are attempting to overcome. From the result of our research, we selected the ten most important papers. Selection criteria were based on the success these researchers had in overcome one or several of the existing challenges. See Table 2 below for details.

2.7 Main attributes and common instances used in E-learning recommendation systems

Table 1 shows the factors that are used in the four different types of recommendation systems in e-learning systems along with the most common attributes and instances that occur within those systems (shown and discussed in detail in Figs. 1, 2 and 3 and Table 7 below).

Table 1 provides an overview of data types, methods of classification, ratings applied, and software and hardware used to generate results. There is little difference between the systems, and all relied to some extent on the same type of input, data treatment and analysis and result generation. The differences between them lay less in general input and output, but in the challenges the researchers aimed to overcome, such as 'quality of recommendation', the 'cold-start' issue, reduction of time and

Table 1 Main Attributes and common instances used in E-learning recommendation systems

Factor	Main Attributes	Common instances
Student Profile Data	Learner profile	Academic, Behaviour, Context, Personal data, Marks/Scores, Log file, learning style, Learning goals, User registers
Student Classification	T-scores, Explicit score, Implicit score, clusters,	Beginners, Intermediate, Masters, Learner per learning objects, Learner per clusters, regular student, bad student, worker student, casual student, absent student, active learners, alumni learners
Data Analyzer	Content Analyzer, Behavioural Analyzer	Academic, Behavioural
Data Adaptation	Adaptation module, ratings	Selection, clustering, similarity ratings, influence set, prediction ratings, user item ratings, learner's demand, learning object list, user-resource evaluation matrix, Item denoising
Interaction and Development	Software	MATLAB, PHP, JAVA, MySQL, Moodle, platforms
	Hardware	Mouse, Keyboard, Computers (Windows / Mac), Tablets, Phone, Laptops, Touch devices

computational expense as well as scalability. For basic operation, all research works under review relied on most, or all, of the elements in Table 1, leading to the conclusion that current recommender system methods are comprehensive and effective since no attempts were made to introduce new approaches.

There are, however, significant differences between the works in terms of the challenges they attempted to address and the RS they chose to achieve their aims. Section 3 provides classification of the systems by aim and type of RS.

3 Classification, discussion and future challenges

Section 3 provides a deeper analysis of some of the elements set out in Table 1. We first classify recommender systems found in the research under review and investigate the challenges addressed by researchers, followed by data mining methods and experimental designs. We then analyze evaluation methods and discuss challenges regarding input and output and summarize the overall findings.

3.1 Classification of recommendation systems

We have classified recommendation systems in four categories, CB, CF, Hybrid systems and Knowledge-based systems. We have further classified CB into classifier-based and neighbor-based. In CB (classifier-based RS), users are classified based on

Table 2 Classification of RS and challenges addressed

Citation (reference)	CB – Classifier-based	CB – Neighbor-based	CF - Neighborhood based	CF – Model-based	Knowledged-based	Hybrid System	cyberspace syndrome	Cold start issue	Quality of recommendation
	Recommendation System						Challenges addressed		
Bourkoukou et al. (2016)			✓				✓		
Klašnja-Milićević et al. (2017)				✓					
El Aissaoui et al. (2018)				✓					✓
Kolekar et al. (2018)	✓								✓
Rahman and Abdullah (2018)	✓						✓		✓
Li et al. (2018)						✓		✓	✓
Bhaskaran and Santhi (2017)						✓			✓
Wan and Niu (2018)					✓				✓
Dwivedi et al. (2017)				✓					✓
Liu (2017)				✓					✓

Table 2 (continued)

Citation (reference)	Shorten execution time	Decrease in memory requirements	Lack of self-awareness of learners	data sparsity issue	Uncertainty of data	gain trust of learners	Sequential order of recommendation	Expandability	Stability
	Challenges addressed								
Bourkoukou et al. (2016)									
Klašnja-Milićević et al. (2017)	✓	✓							
El Aissaoui et al. (2018)			✓		✓				
Kolekar et al. (2018)									
Rahman and Abdullah (2018)									
Li et al. (2018)				✓					
Bhaskaran and Santhi (2017)						✓			
Wan and Niu (2018)			✓						
Dwivedi et al. (2017)			✓				✓		
Liu (2017)				✓					✓

their profiles and recommended materials are provided based on the content of the item, whereas, in CB (neighbor-based RS), items are stored based on the user ratings and new items are recommended based on similar properties of items. Hybrid systems and Knowledge-based systems are not classified into sub-types.

From the ten superior systems we have classified in Table 1, the majority follow CF recommendation systems, a common technique used due to its practicality and ease of application in different online applications and social networks. As shown in Fig. 4, from all papers reviewed; 48.5% papers follow the CF technique with the majority using a model-based RS. In a model-based approach, recommendations are made based on ratings which include characteristics of user and item from factors collected from user ratings. The most popular machine learning technique used in model-based CF is SVD generating factorization of user-item rating matrix. It is popular due to its accuracy and scalability. A further observation from Table 1 is that, despite the ability of the Hybrid technique to overcome the disadvantages of both CF and CB, it is not particularly popular with developers of RS for e-learning; only 20% have used the hybrid technique.

Figure 4 provides an assessment of the popularity of different RS techniques based on the research sample here presented.16 from 35 papers relied on Content Filtering, 6 were Content-Based and 5 focused on Knowledge. Only 7 research works utilized Hybrid systems, despite their obvious advantages.

This may be due to the type of challenge the researchers attempted to overcome. Table 2 below provides an overview of these research aims.

Table 2 identifies the challenges addressed by different research papers. The main objective of researchers when proposing a new RS was to improve the quality of the recommendation and learner awareness of the appropriateness of learning materials, although there were also fringe issues, such as ‘cyberspace syndrome’ and stability (addressed by 2). The most commonly addressed issue was ‘quality of recommendation’ on which 8 papers focused. Only 1 group of researchers addressed the ‘cold-start’ issue, although several also attempted to reduce time and computational expense. None of the papers experimented with scalability.

From the brief analysis in Sect. 3.1, it is evident that more specific issues like cold start, data sparsity and scalability have not been resolved with any single method and remain an area for future research. One interesting finding came from Bhaskaran and Santhi (2017) who focused on gaining the trust of learners, a new challenge addressed

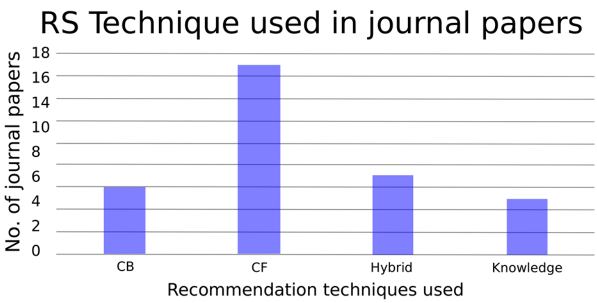

Fig. 4 RS technique used in journal papers

Table 3 Machine learning algorithms applied in RS – E-learning

Author	Sequence pattern	Collaborative tagging	Clustering	Learner learning object rating matrix	K-nearest neighbor	Tensor factorization	Adapted page rank algorithm	Folkrank	Tag-based	Higher Order Singular Value Decomposition	KNN algorithm
	Types of Machine learning algorithms applied										
Bourkoukou et al. (2016)	✓			✓	✓						✓
Klašnja-Milićević et al. (2017)		✓	✓			✓	✓	✓	✓	✓	
El Aissaoui et al. (2018)			✓								
Kolekar et al. (2018)			✓								
Rahman and Abdullah (2018)											
Li et al. (2018)				✓	✓						
Bhaskaran and Santhi (2017)			✓								
Wan and Niu (2018)			✓	✓	✓						
Dwivedi et al. (2017)											
Liu (2017)					✓						✓

Table 3 (continued)

Author	Matrix factorization	Pearson's correlation	Cosine similarity	K-means	Fuzzy C-means clustering	Rule-based	Decision Tree	Support vector machine (SVM)	Firefly	Genetic algorithm	Swarm Intelligence
	Types of Machine learning algorithms applied										
Bourkoukou et al. (2016)		✓	✓								
Klašnja-Milićević et al. (2017)											
El Aissaoui et al. (2018)				✓	✓						
Kolekar et al. (2018)						✓					
Rahman and Abdullah (2018)							✓				
Li et al. (2018)								✓			
Bhaskaran and Santhi (2017)				✓					✓		
Wan and Niu (2018)										✓	
Dwivedi et al. (2017)										✓	✓
Liu (2017)			✓								

Table 4 Experimental settings for ML algorithm

Citation (reference)	Sample/Subjects of study			No. of datasets	Dataset type	Data size	Dataset sections
	Students	Teachers	Both				
Bourkoukou et al. (2016)	✓			2	Implicit	1143	Training sets – 80%, Testing sets - 20%
Klašnja-Milićević et al. (2017)			✓	3	Explicit	120	Experiment (80%) and test set (20%)
El Aissaoui et al. (2018)	✓			2	Explicit	1361	N/A
Kolekar et al. (2018)	✓			2	Explicit	72	57 are regular users, 15 are new users
Rahman and Abdullah (2018)	✓			4	Explicit	70	N/A
Li et al. (2018)	✓			3	Explicit	540	Training sets – 80%, Testing sets - 20%
Bhaskaran and Santhi (2017)			✓	3	Explicit / Implicit	2,78,858	N/A
Wan and Niu (2018)	✓			2	Explicit	623	N/A
Dwivedi et al. (2017)	✓			2	Explicit	200	Training sets – 80%, Testing sets - 20%
Liu (2017)	✓			10	Explicit	52,168	N/A

through recommendation systems. This needs further investigation as user attitude is closely tied to motivation to use the RS frequently while using an e-learning platform. Hence, RS developed in future should focus also on the preference level of users not only on the accuracy and quality of recommendations.

As foreshadowed under Sect. 3, Section 3.1 has classified the research under review by system type and aim in terms of overcoming challenges. Section 3.2 below analyses the data mining algorithms used by different research teams.

3.2 Machine learning (ML) algorithms

Machine learning and data mining techniques can be divided into three steps: data preprocessing, model learning and testing and validation (Amatriain et al. 2010). However, the number of different algorithms and the lack of description of their precise use made it impossible categorize the ML algorithms into sub-categories. Hence in Table 3, we have only listed machine algorithms used in the research.

Clearly, the clustering algorithm was the most used in RS combined with partitioning methods: k-means, k-nearest neighbor and learning object rating matrix algorithms.

Clustering is unsupervised learning where items are allocated to groups with the objective of grouping items with similar characteristics in one group using distance measures. Similarity between the learners is determined based on the distance measured

Table 5 Evaluation metrics applied in RS – E-learning

Citation (reference)	Evaluation approach		MAE	Precision	Recall	Sparsity	F1-measure	Accuracy	Mean	SD	LP-VLGA	LSs and KLs	Self-selection	Response time	Customization
	Online	Offline	Evaluation metrics												
Bourkoukou et al. (2016)	✓		✓			✓									
Klašnja-Milićević et al. (2017)	✓			✓	✓										
El Aissaoui et al. (2018)	✓			✓	✓		✓	✓							
Kolekar et al. (2018)	✓													✓	✓
Rahman and Abdullah (2018)	✓						✓		✓	✓					
Li et al. (2018)	✓		✓	✓	✓		✓								
Bhaskaran and Santhi (2017)	✓		✓	✓	✓			✓							
Wan and Niu (2018)	✓								✓	✓					
Dwivedi et al. (2017)	✓										✓	✓	✓		
Liu (2017)	✓		✓												

Table 6 Input / Output and Existing Problems in RS

Citation (reference)	Learner's interest	Learner's tags	Learner's log files	Learner's styles	Learner's activity	Learner's profile	Implicit / Explicit	Learning objects	Learning content	Learning path	Problems in current system
	Input data							Output data			
Bourkoukou et al. (2016); Klašnja-Milićević et al. (2017)	✓		✓				Both	✓			Data sparsity, Cold start
El Aissaoui et al. (2018)											Expensive
Kolekar et al. (2018)			✓	✓			Both		✓		Average accuracy
Rahman and Abdullah (2018)			✓	✓			Both		✓		Scalability, poor validation
Li et al. (2018)			✓			✓	Both			✓	Scalability
Bhaskaran and Santhi (2017)			✓				Both			✓	Low degree of personalization
Wan and Niu (2018)			✓	✓			Both		✓		Time consuming
Dwivedi et al. (2017)	✓			✓			Both	✓			Low precision, scalability
Bourkoukou et al. (2016)					✓	✓	Both			✓	Time consuming
Liu (2017)				✓		✓	Both	✓			Scalability

between objects. The objective of clustering is to minimize the distance between the objects (Amatriain et al. 2010).

From Sect. 3.2, it is discernible that most research employed supervised learning for RS. ML algorithms which were rarely used were Support Vector Machines (SVM) and Neural networks, highly likely because only RS in the E-learning domain were reviewed. Future research might investigate the usefulness of ML algorithms from other domains (e.g. movie and product). We next evaluate how researchers manipulated experimental settings and used evaluation metrics (Sect. 3.3).

3.3 Experimental settings and evaluation metrics

Without experiments, researches are unable to evaluate how RS or ML algorithms are performing, making it highly important to test and evaluate proposed systems before implementation. Evaluation experiments can be carried out online using real time users in on- or offline experiments in lab settings or random trials including A/B tests. In this section we have reviewed journal papers based on experimental settings as shown in Table 4 and then classified based on the evaluation approach and evaluation metrics applied for the experiments. Since data sets, number of data sets, data size and evaluation metrics varied according to research, we were unable to compare results derived from performance metrics as there were no common grounds or common metrics for evaluation. The purpose of Table 5 is, therefore, to identify the most popular evaluation metrics that can be used for evaluation of RS in the e-learning domain.

From Table 5, it can be observed that most of the studies implemented accuracy metrics like Mean, Precision, Recall and F-measure in rather than alternative metrics such as preference and customization. Since all the papers reviewed used ML algorithms for proposing RS, all papers used performance metrics to evaluate and describe the algorithms. From Table 5, it is evident that MAE, Precision and Recall are the most common performance metrics used for evaluation. One of the reasons for this might be that those metrics are easy to understand, and simple to formulate and compare results with other RS techniques. Most papers used those metrics as a group or one of them with other metrics included. While calculating recall and precision, the larger the value of recall, the better the performance of RS. Whereas, in MAE; the lower the value, the better is the recommendation result. A problem that can arise with precision is that, if a learner is recommended all items, it will compute 100% recall value but out of 1000 learning materials only 10 materials are rated or used by learners, then precision is 0.1. Hence, there is negative relationship between precision and recall. Formulation of algorithms should, thus, aim to maximize both precision and recall. Furthermore, there are, limitations in terms of MAE; it only evaluates the accuracy not the order of recommendations. Hence, it can be concluded that most of the research papers have evaluated their technique only with metrics which are focused on accuracy and performance rather than ranking of the recommendations. Ranking metrics should also be given emphasis in future research as recommendation lists can be long and learners can ignore the most relevant learning materials if they are displayed last.

In Table 4, the main subject of studies was analyzed. This table shows that 80% of the journal papers focused only on the learners' perspective while developing or testing the RS system and only 20% focused on both teachers and learners while none looked at the teachers' perspective only. This shows that RS in e-learning are user-centric and

Table 7 Detailed insights of selected journal publications

Citation	Name of RS system	Technology used	Target user group	Application area	Country of study	Data collected
Bourkoukou et al. (2016)	Learner learning objects recommendation (LLOR)	MATLAB, Sequential pattern mining	Students, Universities, Tutor and e-learning	Technology enhanced learning	Saudi Arabia	Real world data extracted from Cognitive Tutor systems
Liu (2017)	Collaborative filtering based on the influence sets of e-learning group's behaviour (CF-ISEGB)	N/A	Students, learners, web-based services	Web based e-learning	China	IT members data from Shandong Province of China
Klašnja-Milićević et al. (2017)	Protus (Personalized recommendations in a programming tutor system)	Java	Students, Academic Institutions, Educators	Tutoring system	Serbia and Norway	Survey of high school students of Centre of Young Talents Project
El Aissaoui et al. (2018)	Learning object – Felder Silverman Model (LO- FSLSM)	Moodle platform	Students, Academic Institutions, Educators	Technology enhanced learning	Morocco	Data extracted from Sup Management Group
Kolekar et al. (2018)	Rule-based adaptive user interface	N/A	Students, Academic Institutions, Educators	Learning portal for Android course	India	Learners data from portal www.mitelearning.com
Rahman and Abdullah (2018)	Personalized group-based recommendation system	Java, PHP, MySQL	Students, Academic Institutions, E-learning	Web search in E-learning	Malaysia	LMS
Li et al. (2018)	Personalized recommendation of teaching resources	N/A	Students, Teachers, Academic institutions, online learning systems	Online learning system	China	Data extracted from a web learning platform (www.evaluate.guoshi.com/publishg/)s
Bhaskaran and Santhi (2017)	Trust-based hybrid recommendation (TBHR)	Java	Universities, Distant learning services, Students	E-learning on cloud-based platform	India	Data extracted from www2.informatik.uni-freiburg.de/~cziegler/-BX/
Wan and Niu (2018)	Learning object self-organization-based recommendation system	N/A	Universities, Schools, Educational institutions	Technology based e-learning	China	Participants' data taken from courses – Visual Basic and C Programming
Dwivedi et al. (2017)	Learning path recommendation system (LPRS)	PHP, MATLAB	Students, Academic Institutions, E-learning	Web based learning portal	India	Database of B. Tech course of University in Allahabad

LMS: Learning Management System

hence the evaluation of the system should reflect this. For experiments, researchers used datasets ranging from 2 to 10 and from 70 to 278,858 learners, including ratings, preferences, tags, etc. Data are divided into training and testing sets. This has repercussions in terms of the time required for the test as well as the total outlay needed for running the experiments.

These insights are now analyzed in terms of evaluation and validation (Table 5).

An observation from Table 5 is that 80% of evaluation was carried out online compared to 20% offline. Compared to offline settings in lab environments, online testing with real users is more effective, although it has limitations in terms of time and cost for execution of the experiments. Nevertheless, overall the online approach is most appropriate for evaluation, but studies so far have focused only on the accuracy of the system. Hence, the area of evaluation of RS needs to be widened for future studies, covering evaluating user satisfaction, usability, diversity, privacy, risk, trust, scalability, etc. Another area for future investigation is to measure methods of interaction of RS with users as most of the recommendation results depend on user ratings and feedback. From the 10 papers that were listed in the table below, only Kolekar et al. (2018) attempted to cover such aspect of evaluation in their study.

The purpose of Sect. 3.3 has been to analyse the use of experimental designs and evaluation methods. It was found that design is entirely user-centric but that the evaluation does not evaluate user issues. Design and evaluation are, therefore, at variance. The following Sect. 3.4 looks at in- and output challenges in the light of findings so far.

3.4 Input/output and existing problems in RS

In Table 6, we have analyzed major data taken as input for the recommendation system in e-learning. Data are composed of information from the users utilized to generate the recommendation and the output that users will receive as recommended list. The type of data varies across the research under investigation in terms of ratings, tags, reviews, preferences, behaviors, history, etc. and can be diverse depending on the RS system and algorithms those are exploited. From Table 6, it is clear most of the systems have taken learners' log files and styles as input data. From the log files, likes and dislikes, interaction histories, user queries, etc. can be extracted by RS. Such data can be used to initialize the whole recommendation system. Learning styles are behavioral information that can also be used to find common interests of users and uncover more specific details such as features of learning material that users like. A further observation from Table 6 is that the main aim of most RS is to generate learning content or learning object as its output to users. Few also provide learning paths as its recommendation results.

Table 6 also shows the issues that still exist in the RS systems recommended by researchers. It is evident that scalability is the major issue, the expansion of data sets in terms of users and items. To enable use of the proposed RS systems under real conditions, they must be capable of handling real time user requirements either to maintain the system speed or resource consumption behavior of the system when data increase. Since RS is expected to give prompt recommendations online, it is of utmost importance that RS lives up to the expectations to produce even when the data scale

Table 8 Term Frequency in 10 journal publications chosen

Term	Frequency	Term	Frequency
recommendation(s) based	914	Knowledge	78
e-learning	989	Adaptive	74
learner(s)	526	Evaluation	73
algorithm(s)	247	Hybrid	55
Recommender	130	Recommended	49
Collaborative	112	personalized / personalization	77
Content	91		

expands. The second major issue is related and refers to time consumption or latency of the system for generating the recommendations.

These problems can be solved with the use of clustering focused on small clusters rather than overall data. Other solutions involve SVD algorithms and combinations of CB and CF techniques. Clustering algorithms can also overcome other challenges such as cold start and data sparsity, which may also be improved through hybrid techniques.

3.5 Detailed insights into selected publications

In Table 7 below, we have listed brief details of the ten selected research papers giving details of recommendation systems along with technology used, target user groups for whom the proposed RS was developed, country of study, data used in the evaluation of the RS and application area of the proposed system.

Table 7 shows that seven of the selected works were studies from Asian countries, and only one each from the Middle East, Northern Africa and Europe. Most of the RS used Java programming language for development with technology-based learning systems as the major application. Target groups were academic institutions and their students. All systems utilized external data for the evaluation of their systems.

Based on the foregoing analysis, the appropriateness of the research approach this paper has taken is demonstrated with Table 8. Here, the 13 most used terms and frequency of use in the publications are shown. Our classification tables have covered four major techniques: content, collaborative, hybrid and knowledge. Comparing the classification with this table, it shows that the analysis has covered all the most discussed terms in the 10 reviewed journal papers.

Table 9 Journal Papers available in CSU Primo search from year 2016–2018

S. No.	Keywords	Total number of results
1	E-learning	2627
2	Recommendation system e-learning	820
3	Content recommendation system e-learning	652
4	Collaborative recommendation system e-learning	369
5	Hybrid	185

From Table 9, the total number of results from a library search are shown based on selected keywords and phrases It shows the relevancy of keywords in the journal database used in the study.

4 Conclusion

Since the development of RS, recommender systems are generally used in e-learning platforms to implement adaptive learning environments and to solve the issue of information overload for users. This review is based on 101 papers related to RS in e-learning published in the last few years. The main contribution of this paper is that it provides a taxonomy of RS systems along with ML algorithms, evaluation metrics applied along insights on challenges and issues that are yet to be addressed by future research. This study has shown CF as popular recommendation technique used in E-learning with most of the studies aiming to improve the quality of recommendations. Among the four techniques, hybrid techniques have a competitive edge, yet their popularity is low. ML algorithms could not be classified in this paper due to multiple algorithms used in single systems. Clustering was a common ML technique with future research pointing to SVN and neural networks to enhance results. Although RS in e-learning are based on learners, evaluation of the system was more focused on measuring the accuracy of the algorithms through Mean, Precision, Recall and F-measure rather than evaluating the impact on satisfaction and preference level of users. Major issues that still need to be resolved in RS are scalability and latency with new emerging issues of privacy and shilling attacks. Results have shown that the hybrid technique can resolve most common challenges in current systems. However, further investigation is needed to improve user trust and aspects of interaction of users with systems.

References

Abdullah, M., Daffa, H. W., Bashmail, R. M., Alzahrani, M., & Sadik, M. (2015). The impact of learning styles on learner's performance in e-learning environment. *International Journal of Advanced Computer Science and Applications, 6*(9). https://doi.org/10.14569/ijacsa.2015.060903.

Adomavicius, G., & Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering, 17*(6), 734–749. https://doi.org/10.1109/TKDE.2005.99.

Aeiad, E., & Meziane, F. (2018). An adaptable and personalised E-learning system applied to computer science Programmes design. *Education and Information Technologies, 24*, 1485–1509. https://doi.org/10.1007/s10639-018-9836-x.

Aggarwal, C. C. (2016a). An introduction to recommender systems. In C. C. Aggarwal (Ed.), *Recommender systems* (1st ed., pp. 1–28). Cham: Springer Retrieved from https://link.springer.com/book/10.1007%2F978-3-319-29659-3.

Aggarwal, C. C. (2016b). Evaluation of recommender systems. In C. C. Aggarwal (Ed.), *Recommender systems* (1st ed.). Cham: Springer Retrieved from https://link.springer.com/book/10.1007%2F978-3-319-29659-3.

Aher, S., & Lobo, L. (2013). Combination of machine learning algorithms for recommendation of courses in E-learning system based on historical data. *Knowledge-Based Systems, 51*, 1–14. https://doi.org/10.1016/j.knosys.2013.04.015.

Al-Shamri, M. Y., & Bharadwaj, K. K. (2008). Fuzzy-genetic approach to recommender systems based on a novel hybrid user model. *Expert Systems with Applications, 35*(3), 1386–1399. https://doi.org/10.1016/j.eswa.2007.08.016.

Amatriain, X., Jaimes, A., Oliver, N., & Pujol, J. (2010). Data mining methods for recommender systems. In F. Ricci, L. Rokach, B. Shapira, & P. Kantor (Eds.), *Recommender systems handbook* (pp. 39–71). Boston, MA: Springer. https://doi.org/10.1007/978-0-387-85820-3_2.

Batmaz, Z., Yurekli, A., Bilge, A., & Kaleli, C. (2018). A review on deep learning for recommender systems: Challenges and remedies. *Artificial Intelligence Review, 52*, 1–37. https://doi.org/10.1007/s10462-018-9654-y.

Benhamdi, S., Babouri, A., & Chiky, R. (2016). Personalized recommender system for e-learning environment. *Education and Information Technologies, 22*(4), 1455–1477. https://doi.org/10.1007/s10639-016-9504-y.

Bennett, S., Agostinho, S., & Lockyer, L. (2015). Technology tools to support learning design: Implications derived from an investigation of university teachers' design practices. *Computers & Education, 81*, 211–220. https://doi.org/10.1016/j.compedu.2014.10.016.

Bhaskaran, S., & Santhi, B. (2017). An efficient personalized trust-based hybrid recommendation (TBHR) strategy for e-learning system in cloud computing. *Cluster Computing, 22*, 1137–1149. https://doi.org/10.1007/s10586-017-1160-5.

Bourkoukou, O., El Bachari, E., & El Adnani, M. (2016). A recommender model in E-learning environment. *Arabian Journal for Science and Engineering, 42*(2), 607–617. https://doi.org/10.1007/s13369-016-2292-2.

Chen, H., Li, Z., & Hu, W. (2015). An improved collaborative recommendation algorithm based on optimized user similarity. *The Journal of Supercomputing, 72*(7), 2565–2578. https://doi.org/10.1007/s11227-015-1518-5.

Chen, Y., Tseng, C., Huang, C., Deng, L., & Lee, W. (2016). Recommendation system based on rule-space model of two-phase blue-red tree and optimized learning path with multimedia learning and cognitive assessment evaluation. *Multimedia Tools and Applications, 76*(18), 18237–18264. https://doi.org/10.1007/s11042-016-3717-3.

Chiou, C., Lee, L., & Liu, Y. (2012). Effect of Novak colorful concept map with digital teaching materials on student academic achievement. *Procedia - Social and Behavioral Sciences, 64*, 192–201. https://doi.org/10.1016/j.sbspro.2012.11.023.

Clements, K., Pawlowski, J., & Manouselis, N. (2015). Open educational resources repositories literature review – Towards a comprehensive quality approaches framework. *Computers in Human Behavior, 51*, 1098–1106. https://doi.org/10.1016/j.chb.2015.03.026.

Donzellini, G., & Ponta, D. (2007). A simulation environment for e-learning in digital design. *IEEE Transactions on Industrial Electronics, 54*(6), 3078–3085. https://doi.org/10.1109/tie.2007.907011.

Dwivedi, P., & Bharadwaj, K. K. (2012). E-learning recommender system for learners in online social networks through association retrieval. *Proceedings of the CUBE International Information Technology Conference on - CUBE 12*. https://doi.org/10.1145/2381716.2381846.

Dwivedi, P., Kant, V., & Bharadwaj, K. K. (2017). Learning path recommendation based on modified variable length genetic algorithm. *Education and Information Technologies, 23*(2), 819–836. https://doi.org/10.1007/s10639-017-9637-7.

Ekstrand, M. (2011). Collaborative filtering recommender systems. *Foundations and Trends® in Human–Computer Interaction, 4*(2), 81–173. https://doi.org/10.1561/1100000009.

El Aissaoui, O., El Alami El Madani, Y., Oughdir, L. and El Allioui, Y. (2018). A fuzzy classification approach for learning style prediction based on web mining technique in e-learning environments. *Education and Information Technologies, 24*(3), pp.1943–1959.

El-Bishouty, M., Aldraiweesh, A., Alturki, U., Tortorella, R., Yang, J., Chang, T., et al. (2018). Use of Felder and Silverman learning style model for online course design. *Educational Technology Research And Development, 67*, 161–177. https://doi.org/10.1007/s11423-018-9634-6.

Fatahi, S., Shabanali-Fami, F., & Moradi, H. (2017). An empirical study of using sequential behavior pattern mining approach to predict learning styles. *Education and Information Technologies, 23*(4), 1427–1445. https://doi.org/10.1007/s10639-017-9667-1.

Feldman, J., Monteserin, A., & Amandi, A. (2014). Automatic detection of learning styles: State of the art. *Artificial Intelligence Review, 44*(2), 157–186. https://doi.org/10.1007/s10462-014-9422-6.

Felfernig, A., Polat-Erdeniz, S., Uran, C., Reiterer, S., Atas, M., Tran, T., et al. (2018). An overview of recommender systems in the internet of things. *Journal of Intelligent Information Systems, 52*, 285–309. https://doi.org/10.1007/s10844-018-0530-7.

Fouad, K. M. (2012). Proposed approach to build semantic learner model in adaptive e-learning. *International Journal of Computer Applications, 58*(17), 40–47. https://doi.org/10.5120/9377-3859.

Goldberg, D., Nichols, D., Oki, B., & Terry, D. (1992). Using collaborative filtering to weave an information tapestry. *Communications of the ACM, 35*(12), 61–70. https://doi.org/10.1145/138859.138867.

Goldstein, T., & Osher, S. (2009). The Split Bregman method for L1-regularized problems. *SIAM Journal on Imaging Sciences, 2*(2), 323–343. https://doi.org/10.1137/080725891q7637-6_8.

Goodfellow, L. Bengio, Y. & Courville, A. (2016). *Deep learning*. MIT Press. http://www.deeplearningbook.org

Guo, G., Zhang, J., & Yorke-Smith, N. (2016). A novel recommendation model regularized with user trust and item ratings. *IEEE Transactions on Knowledge and Data Engineering, 28*(7), 1607–1620. https://doi.org/10.1109/tkde.2016.2528249.

Guo, L., Liang, J., Zhu, Y., Luo, Y., Sun, L., & Zheng, X. (2018). Collaborative filtering recommendation based on trust and emotion. *Journal of Intelligent Information Systems, 53*, 113–135. https://doi.org/10.1007/s10844-018-0517-4.

Gupta, V., & Pandey, S. R. (2019). Recommender Systems for Digital Libraries: A review of concepts and concerns. *Library Philosophy and Practice*, 1–9.

Han, J., Jo, J., Ji, H., & Lim, H. (2016). A collaborative recommender system for learning courses considering the relevance of a learner's learning skills. *Cluster Computing, 19*(4), 2273–2284. https://doi.org/10.1007/s10586-016-0670-x.

Herlocker, J. L., Konstan, J. A., Terveen, L. G., & Riedl, J. T. (2004). Evaluating collaborative filtering recommender systems, ACM Transactions on Information Systems., *22*(1), 5–53. https://doi.org/10.1145/963770.963772.

Hlosta, M., Zdrahal, Z., & Zendulka, J. (2017). Ouroboros. *Proceedings of the Seventh International Learning Analytics & Knowledge Conference on - LAK, 17*, 6–15. https://doi.org/10.1145/3027385.3027449.

Holtz, P., Kimmerle, J., & Cress, U. (2018). Using big data techniques for measuring productive friction in mass collaboration online environments. *International Journal of Computer-Supported Collaborative Learning, 13*(4), 439–456. https://doi.org/10.1007/s11412-018-9285-y.

Horváth, T., & Carvalho, A. (2017). Evolutionary computing in recommender systems: A review of recent research. *Natural Computing, 16*(3), 441–462. https://doi.org/10.1007/s11047-016-9540-y.

Huang, T., & Huang, Y. (2015). Where are my cooperative learning companions: Designing an intelligent recommendation mechanism? *Multimedia Tools and Applications, 76*(9), 11547–11565. https://doi.org/10.1007/s11042-015-2678-2.

Huang, S., Zhang, J., Wang, L., & Hua, X. (2016). Social friend recommendation based on multiple network correlation. *IEEE Transactions on Multimedia, 18*(2), 287–299. https://doi.org/10.1109/tmm.2015.2510333.

Hussain, M., Zhu, W., Zhang, W., Abidi, S. M., & Ali, S. (2018). Using machine learning to predict student difficulties from learning session data. *Artificial Intelligence Review, 52*, 381–407. https://doi.org/10.1007/s10462-018-9620-8.

Imran, H., Belghis-Zadeh, M., Chang, T., Kinshuk, & Graf, S. (2015). PLORS: A personalized learning object recommender system. *Vietnam Journal of Computer Science, 3*(1), 3–13. https://doi.org/10.1007/s40595-015-0049-6.

Jeong, W., Kim, S., Park, D., & Kwak, J. (2013). Performance improvement of a movie recommendation system based on personal propensity and secure collaborative filtering. *Journal of Information Processing Systems, 9*(1), 157–172. https://doi.org/10.3745/jips.2013.9.1.157.

Kamsa, I., Elouahbi, R., Khoukhi, F. E., Karite, T., & Zouiten, H. (2016). Optimizing collaborative learning path by ant's optimization technique in e-learning system. *2016 15th International Conference on Information Technology Based Higher Education and Training (ITHET)*. https://doi.org/10.1109/ithet.2016.7760697.

Karga, S., & Satratzemi, M. (2017). A hybrid recommender system integrated into LAMS for learning designers. *Education and Information Technologies, 23*(3), 1297–1329. https://doi.org/10.1007/s10639-017-9668-0.

Kim, K., & Ahn, H. (2008). A recommender system using GA K-means clustering in an online shopping market. *Expert Systems with Applications, 34*(2), 1200–1209. https://doi.org/10.1016/j.eswa.2006.12.025.

Kim, S., & Yoon, Y. (2014). Recommendation system for sharing economy based on multidimensional trust model. *Multimedia Tools and Applications, 75*(23), 15297–15310. https://doi.org/10.1007/s11042-014-2384-5.

Kim, J., Lee, J., Park, D., Lim, Y., Lee, Y., & Jung, E. (2013). Adaptive mining prediction model for content recommendation to coronary heart disease patients. *Cluster Computing, 17*(3), 881–891. https://doi.org/10.1007/s10586-013-0308-1.

Kitchin, R. (2014). *The data revolution: Big data, open data, data infrastructures and their consequences*. Thousand Oaks: Sage.

Klašnja-Milićević, A., Ivanović, M., Vesin, B., & Budimac, Z. (2017). Enhancing e-learning systems with personalized recommendation based on collaborative tagging techniques. *Applied Intelligence, 48*(6), 1519–1535. https://doi.org/10.1007/s10489-017-1051-8.

Kolekar, S. V., Pai, R. M., & Manohara Pai, M. M. (2018). Rule based adaptive user interface for adaptive E-learning system. *Education and Information Technologies*, 1–29. https://doi.org/10.1007/s10639-018-9788-1.

Krithika, L. B., & Lakshmi Priya, G. G. (2016). Student emotion recognition system (SERS) for e-learning improvement based on learner concentration metric. *Procedia Computer Science, 85*, 767–776. https://doi.org/10.1016/j.procs.2016.05.264.

Latham, A., Crockett, K., McLean, D., & Edmonds, B. (2012). A conversational intelligent tutoring system to automatically predict learning styles. *Computers & Education, 59*(1), 95–109.

Latham, A., Crockett, K., & Mclean, D. (2013). Profiling student learning styles with multilayer perceptron neural networks. *2013 IEEE International Conference on Systems, Man, and Cybernetics*. https://doi.org/10.1109/smc.2013.428.

Latham, A., Crockett, K., & McLean, D. (2014). An adaptation algorithm for an intelligent natural language tutoring system. *Computers & Education, 71*, 97–110. https://doi.org/10.1016/j.compedu.2013.09.014.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature, 521*(7553), 436–444.

Li, H., Li, H., Zhang, S., Zhong, Z., & Cheng, J. (2018). Intelligent learning system based on personalized recommendation technology. *Neural Computing and Applications*, 1–8. https://doi.org/10.1007/s00521-018-3510-5.

Liu, X. (2017). A collaborative filtering recommendation algorithm based on the influence sets of e-learning group's behavior. *Cluster Computing, 22*, 2823–2833. https://doi.org/10.1007/s10586-017-1560-6.

Mamat, N., & Yusof, N. (2013). Learning style in a personalized collaborative learning framework. *Procedia - Social and Behavioral Sciences, 103*, 586–594. https://doi.org/10.1016/j.sbspro.2013.10.376.

Manouselis, N., Vuorikari, R., & Van Assche, F. (2010). Collaborative recommendation of e-learning resources: An experimental investigation. *Journal of Computer Assisted Learning, 26*(4), 227–242.

Marinho, L. B., Hotho, A., Jäschke, R., Nanopoulos, A., Rendle, S., Schmidt-Thieme, L., Stumme, G., & Symeonidis, P. (2012). *Recommender systems for social tagging systems*. New York: Springer.

Najafabadi, M., & Mahrin, M. (2015). A systematic literature review on the state of research and practice of collaborative filtering technique and implicit feedback. *Artificial Intelligence Review, 45*(2), 167–201. https://doi.org/10.1007/s10462-015-9443-9.

Narducci, F., Musto, C., Polignano, M., Gemmis, M., Lops, P., & Semeraro, G. (2015, May). A recommender system for connecting patients to the right doctors in the healthnet social network. In *Proceedings of the 24th international conference on World Wide Web* (pp. 81–82). New York: ACM.

Ning, X., Desrosiers, C., & Karypis, G. (2015). A comprehensive survey of neighborhood-based recommendation methods. *Recommender Systems Handbook*, 37–76. https://doi.org/10.1007/978-1-4899-7637-6_2.

Nitchot, A., Wettayaprasit, W., & Gilbert, L. (2018). Assistive tool for constructing knowledge structures and suggesting related study materials links. *Education and Information Technologies, 24*, 219–230. https://doi.org/10.1007/s10639-018-9762-y.

Nocera, A., & Ursino, D. (2011). An approach to providing a user of a "social folksonomy" with recommendations of similar users and potentially interesting resources. *Knowledge-Based Systems, 24*(8), 1277–1296. https://doi.org/10.1016/j.knosys.2011.06.003.

Pereira, C. K., Campos, F., Ströele, V., David, J. M., & Braga, R. (2018). BROAD-RSI – Educational recommender system using social networks interactions and linked data. *Journal of Internet Services and Applications, 9*(1). https://doi.org/10.1186/s13174-018-0076-5.

Phobun, P., & Vicheanpanya, J. (2010). Adaptive intelligent tutoring systems for e-learning systems. *Procedia - Social and Behavioral Sciences, 2*(2), 4064–4069. https://doi.org/10.1016/j.sbspro.2010.03.641.

Piech, C., Bassen, J., Huang, J., Ganguli, S., Sahami, M., Guibas, L. J., & Sohl-Dickstein, J. (2015). Deep knowledge tracing. In *Advances in neural information processing systems* (pp. 505–513).

Qiao, J., Pan, G., & Han, H. (2015). Design and application of continuous deep belief network. *Acta Automatica Sinica*.

Rahman, M., & Abdullah, N. (2018). A personalized group-based recommendation approach for web search in E-learning. *IEEE Access, 6*, 34166–34178. https://doi.org/10.1109/access.2018.2850376.

Rapečka, A., & Dzemyda, G. (2015). A new recommendation model for the user clustering-based recommendation system. *Information Technology and Control, 44*(1).

Resnick, P., & Varian, H. R. (1997). Recommender systems. *Communications of the ACM, 40*(3), 56–59.

Riahi, G. (2015). E-learning systems based on cloud computing: A review. *Procedia Computer Science, 62*, 352–359. https://doi.org/10.1016/j.procs.2015.08.415.

Ronghuai, H., Junfeng, Y., & Yongbin, H. (2012). From digital to smart: The evolution and trends of learning environment. *Open Education Research, 01*, 75–84.
Salama, M., Hassanien, A., Fahmy, A. (2010). Deep belief network for clustering and classification of a continuous data. *The 10th IEEE International Symposium on Signal Processing and Information Technology. IEEE*, 473–477.
Salonen, V., & Karjaluoto, H. (2016). Web personalization: The state of the art and future avenues for research and practice. *Telematics and Informatics, 33*(4), 1088–1104.
Shu, J., Shen, X., Liu, H., Yi, B., & Zhang, Z. (2017). A content-based recommendation algorithm for learning resources. *Multimedia Systems, 24*(2), 163–173. https://doi.org/10.1007/s00530-017-0539-8.
Smys, S., & Bala, G. (2012). Performance analysis of virtual clusters in personal communication networks. *Cluster Computing, 15*(3), 211–222. https://doi.org/10.1007/s10586-012-0209-8.
Su, X., & Khoshgoftaar, T. (2009). A survey of collaborative filtering techniques. *Advances in Artificial Intelligence, 2009*, 1–19. https://doi.org/10.1155/2009/421425.
Tarus, J., Niu, Z., & Mustafa, G. (2017). Knowledge-based recommendation: A review of ontology-based recommender systems for e-learning. *Artificial Intelligence Review, 50*(1), 21–48. https://doi.org/10.1007/s10462-017-9539-5.
Tian, L., Fan, C., & Ming, Y. (2016). Multiple scales combined principle component analysis deep learning network for face recognition. *Journal of Electronic Imaging, 25*(2), 023025. https://doi.org/10.1117/1.jei.25.2.023025.
Trifa, A., Sbaï, A.H., & Chaari, W.L. (2018). Knowledge tracing with an intelligent agent, in an e-learning platform. *Education and Information Technologies, 24,* 711–741.
Truong, H. M. (2016). Integrating learning styles and adaptive e-learning system: Current developments, problems and opportunities. *Computers in Human Behavior, 55*, 1185–1193.
Vahdat, M., Oneto, L., Anguita, D., Funk, M., & Rauterberg, M. (2015). A learning analytics approach to correlate the academic achievements of students with interaction data from an educational simulator. *Design for Teaching and Learning in a Networked World Lecture Notes in Computer Science*, 352–366. https://doi.org/10.1007/978-3-319-24258-3_26.
Wan, S., & Niu, Z. (2016). A learner-oriented learning recommendation approach based on mixed concept mapping and immune algorithm. *Knowledge-Based Systems, 103*, 28–40. https://doi.org/10.1016/j.knosys.2016.03.022.
Wan, S., & Niu, Z. (2018). An e-learning recommendation approach based on the self-organization of learning resource. *Knowledge-Based Systems, 160*, 71–87. https://doi.org/10.1016/j.knosys.2018.06.014.
Wang, D., & Li, M. (2017). Stochastic configuration networks: Fundamentals and algorithms. *IEEE Transactions on Cybernetics, 47*(10), 3466–3479.
Wang, S., & Wu, C. (2011). Application of context-aware and personalized recommendation to implement an adaptive ubiquitous learning system. *Expert Systems with Applications, 38*(9), 10831–10838. https://doi.org/10.1016/j.eswa.2011.02.083.
Wang, P., & Yang, H. (2012). Using collaborative filtering to support college students' use of online forum for English learning. *Computers & Education, 59*(2), 628–637. https://doi.org/10.1016/j.compedu.2012.02.007.
Wang, Z., Liao, J., Cao, Q., Qi, H., & Wang, Z. (2015). Friendbook: A semantic-based friend recommendation system for social networks. *IEEE Transactions on Mobile Computing, 14*(3), 538–551. https://doi.org/10.1109/tmc.2014.2322373.
Wang, R., Jiang, Y., Li, Y., & Lou, J. (2016). A collaborative filtering recommendation algorithm based on multiple social trusts. *Journal of Computer Research Device, 53*, 1389–1399.
Winoto, P., Tang, T., & McCalla, G. (2012). Contexts in a paper recommendation system with collaborative filtering. *The International Review of Research in Open and Distributed Learning, 13*(5), 56. https://doi.org/10.19173/irrodl.v13i5.1243.
Xiao, J., Wang, M., Jiang, B., & Li, J. (2017). A personalized recommendation system with combinational algorithm for online learning. *Journal of Ambient Intelligence and Humanized Computing, 9*(3), 667–677. https://doi.org/10.1007/s12652-017-0466-8.
Xie, F., Chen, Z., Shang, J., & Fox, G. (2014). Grey forecast model for accurate recommendation in presence of data sparsity and correlation. *Knowledge-Based Systems, 69*, 179–190. https://doi.org/10.1016/j.knosys.2014.04.011.
Xin, T., Xu, Z., & Tatsuoka, K. (2004). Linkage between teacher quality, student achievement, and cognitive skills: A rule-space model. *Studies in Educational Evaluation, 30*(3), 205–223. https://doi.org/10.1016/j.stueduc.2004.09.002.

Xu, C. (2013). Personal recommendation using a novel collaborative filtering algorithm in customer relationship management. *Discrete Dynamics in Nature and Society, 2013*, 1–9. https://doi.org/10.1155/2013/739460.

Yang, B., & Sun, Y. (2013). An exploration of link-based knowledge map in academic web space. *Scientometrics, 96*(1), 239–253. https://doi.org/10.1007/s11192-012-0919-y.

Yera, R., & Martínez, L. (2017). A recommendation approach for programming online judges supported by data preprocessing techniques. *Applied Intelligence, 47*(2), 277–290. https://doi.org/10.1007/s10489-016-0892-x.

You, J., Wang, Y., Pal, A., Eksombatchai, P., Rosenburg, C., & Leskovec, J. (2019, May). Hierarchical temporal convolutional networks for dynamic recommender systems. In *The world wide web conference* (pp. 2236–2246). New York: ACM.

Zhang, Y., Lo, D., Xia, X., Scanniello, G., Le, T. B., & Sun, J. (2017). Fusing multi-abstraction vector space models for concern localization. *Empirical Software Engineering, 23*(4), 2279–2322. https://doi.org/10.1007/s10664-017-9585-2.

Zhang, H., Huang, T., Lv, Z., Liu, S., & Yang, H. (2018). MOOCRC: A highly accurate resource recommendation model for use in MOOC environments. *Mobile Networks and Applications, 24*, 34–46. https://doi.org/10.1007/s11036-018-1131-y.

Zhou, Y., Huang, C., Hu, Q., Zhu, J., & Tang, Y. (2018). Personalized learning full-path recommendation model based on LSTM neural networks. *Information Sciences, 444*, 135–152.