



OPEN ACCESS

EDITED BY

Zhi Liu,
Central China Normal University, China

REVIEWED BY

Yacine Lafifi,
8 May 1945 University of Guelma, Algeria
Changsheng Chen,
Shandong Women's University, China
Nouhaila Idrissi,
Mohammed V University, Morocco

*CORRESPONDENCE

Abdallatif Abu-Issa
✉ abuisa@birzeit.edu

RECEIVED 21 March 2024

ACCEPTED 12 August 2024

PUBLISHED 09 September 2024

CITATION

Butmeh H and Abu-Issa A (2024) Hybrid attribute-based recommender system for personalized e-learning with emphasis on cold start problem.

Front. Comput. Sci. 6:1404391.

doi: 10.3389/fcomp.2024.1404391

COPYRIGHT

© 2024 Butmeh and Abu-Issa. This is an open-access article distributed under the terms of the [Creative Commons Attribution License \(CC BY\)](#). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

Hybrid attribute-based recommender system for personalized e-learning with emphasis on cold start problem

Hala Butmeh¹ and Abdallatif Abu-Issa^{2*}

¹Data Science and Business Analytics Program, Arab American University Palestine, Jenin, Palestine,

²Department of Electrical and Computer Engineering, Birzeit University, Birzeit, Palestine

This article introduces a recommendation system that merges a knowledge-based (attribute-based) approach with collaborative filtering, specifically addressing the challenges of the pure-cold start scenario in personalized e-learning. The system generates learning recommendations by assessing item similarities, utilizing the Rogers-Tanimoto similarity measure for materials and users, and Jaccard's similarity for user comparisons. Unlike traditional collaborative methods relying on prior ratings, this approach depends on attributes. Additionally, user and learning material profiling structures were created to serve as fundamental inputs for the recommendation algorithm. These profiles represent student and material knowledge in a two-dimensional space to facilitate matching. Our processes incorporate user learning styles, preferences, and prior knowledge as metrics for achieving the desired level of personalization. The system produces a list of top recommendations based on predicted ratings. To validate its efficacy, a website resembling our learning platform was developed and tested by users. The primary results demonstrate the system's ability to identify similar users even in a pure cold start condition without existing ratings. Consequently, the system proves its capability in recommending suitable materials, modeling students, and identifying similar user groups. The evaluation results of the proposed system showed a good level of satisfaction by the testimonials, quantified by a score of 82% for the recommended materials (16% higher than existing cold-start systems), and an average score of 90% in terms of satisfaction about the generated student profiles. As they proved the capability of the framework in recommending suitable materials, and its capability in modeling students, finding similar groups of users.

KEYWORDS

personalized e-learning, recommender systems, cold start problem, learning styles, learner model

1 Introduction

Over the past decade, e-learning and self-directed learning have experienced a surge in popularity, emerging as fundamental educational approaches. Nonetheless, two significant challenges confronting e-learning involve information overload and information relevancy. In response to these challenges, recommendation systems and personalized learning systems have been introduced as essential solutions (Kulkarni et al., 2020).

Broadly speaking, e-learning systems are advantageous for individuals with hectic schedules or those seeking a flexible learning experience, as learning resources are accessible from any location and at any time. This enables efficient and targeted information delivery, in contrast to traditional classrooms that may frequently face disruptions (Joy et al., 2021).

The benefits offered by e-learning platforms attract millions of learners; however, these platforms struggle to retain their users, resulting in a considerable dropout rate. One major factor contributing to this dropout problem is the provision of uniform learning experiences to all students, irrespective of their interests and learning styles, causing a decline in motivation (Hassan et al., 2021).

Recommender systems, widely adopted by companies like Netflix, Amazon, and YouTube, aim to alleviate information overload. While various techniques strive to extract relevant information, conventional methods like collaborative filtering and content-based approaches focus solely on users and items, overlooking distinctions among learners and items. In e-learning, learner characteristics such as learning style and knowledge level play a crucial role in influencing preferences. Traditional recommendation methods may lack accuracy in suggesting materials due to a neglect of these additional learner features (Tarus J. et al., 2017; Jeevamol and Renumol, 2021). Additionally, traditional techniques heavily depend on users' rating history for identifying similar users, predicting similar materials, and providing recommendations. Past studies highlight that the main challenges faced by these conventional recommendation systems are the cold-start problem and rating sparsity, both of which contribute to a decline in their overall performance (Jeevamol and Renumol, 2021). These challenges provide strong incentives to explore innovative recommendation approaches that can more effectively fulfill the goals of e-learning.

In this paper, we introduced an attribute-based e-learning recommendation model aimed at resolving the cold start problem. The model prioritizes the learner's expectations, seeking to meet them efficiently to enhance the overall learning experience, making it more seamless and encouraging for the user.

2 Background and related work

In this section, we introduce the terms and topics that are used throughout the paper.

2.1 Personalized e-learning systems

Traditionally, learning systems followed a "one-size-fits-all" approach, providing the same learning content in a uniform structure for all students (Imran et al., 2014). However, students exhibit variations in their knowledge levels, learning styles, and preferences. Personalization addresses this issue to enhance overall student satisfaction (Fraihat and Shambour, 2015).

In the present day, there is a shift toward more student-oriented learning, replacing traditional tools with personalized learning systems (Sikka et al., 2012; Abouzeid et al., 2021).

Research has demonstrated that adapting learning content to individual preferences, needs, and progress significantly improves learning success (Sikka et al., 2012). Given that students within the same class have diverse knowledge levels and learning paces, providing individual support increases the likelihood of academic success.

The topic of personalization in e-learning has garnered attention in recent research (Bourkhouk et al., 2016). Recommender systems offer a solution by supporting personalization, recommending suitable learning content based on learners' unique needs and characteristics (Fraihat and Shambour, 2015).

2.2 Recommender systems overview

Recommender systems are software tools and mechanisms designed to suggest items that may be of interest and relevance to users (Ricci et al., 2011; Lü et al., 2012). The term "item" commonly refers to the content recommended by a system. Recommender systems focus on specific types of items, tailoring their design and core recommendation approach to provide effective suggestions for that particular item category. The primary purpose of recommendation systems is to assist individuals in navigating the potentially overwhelming volume of alternative items offered by a system (Ricci et al., 2011). With the vast amount of information available on the Internet, learners encounter challenges in finding useful educational materials. To address this issue, recommendation systems have become a popular solution in the field of e-learning, helping users discover relevant materials (Tarus et al., 2017b). These systems are categorized based on the technique employed for recommendation, and the following subsections will briefly explore the four techniques utilized in this paper.

2.2.1 Content-based recommender systems

Content-based recommendation systems operate by identifying items with content similar to those previously favored by a specific user (Lü et al., 2012). These systems analyze a collection of items rated by a user and create a user model that defines the user's interests based on the features of the rated items (Mladenic, 1999). The user model or profile is a structured representation of user interests, employed for subsequent recommendations.

2.2.2 Collaborative filtering recommender systems

The collaborative filtering algorithm takes into account the ratings provided by other users when generating recommendations. The underlying concept of this algorithm is that users with similar preferences will have a liking for the same items. Consequently, it gathers opinions from a sizable network of interconnected users and consolidates them to generate predictions effectively (ben Schafer et al., 2007; Ekstrand et al., 2010). The primary criterion for assessing similarity between two users is their rating history, with two users considered similar if they express a liking for the same set of items (Ricci et al., 2011).

2.2.3 Cold-start problem

The preceding two methods (content-based and collaborative filtering) face the challenge known as the cold-start problem, which is further categorized into new item cold-start and new user cold-start (Lam et al., 2008).

New-item cold start occurs when a fresh item is introduced to the system, initially lacking any ratings. Consequently, the likelihood of this item being selected and recommended is minimal, leading to a situation where a set of items in the system is overlooked in the rating and recommendation processes (Sobhanam and Mariappan, 2013; Saveski and Mantrach, 2014). This creates a cyclical problem where items remain disregarded.

New user cold-start arises when a new user registers in the system, posing the difficulty of providing meaningful recommendations for users with no associated information (Silva et al., 2019). This presents a significant challenge for collaborative filtering, as a user without any ratings cannot be matched to any group of potentially similar users.

2.2.4 Knowledge-based recommender systems

Knowledge-based recommender systems derive their recommendations from domain knowledge regarding how to align items with user preferences. These systems rely on three types of knowledge: information about user attributes, insights into item characteristics, and understanding of how to match items with user needs. In the context of e-learning recommendation systems, knowledge-based techniques prove effective due to the intricate relationships among items in e-learning. In the domain of e-learning, the knowledge-based approach combines information about the learner with knowledge about the learning material for utilization in the recommendation process. Unlike previous methods, knowledge-based recommenders do not encounter the cold-start problem because their recommendations are not tied to ratings; instead, they leverage domain knowledge. However, the drawback of the knowledge-based approach is its reliance on expertise in the field and its time-consuming nature (Shishehchi et al., 2012; Zhao et al., 2014; Colombo-Mendoza et al., 2015; Zhang et al., 2021).

2.2.5 Hybrid recommender systems

Handling the varied challenges posed by diverse users and scenarios remains a formidable task, even for a proficient recommendation algorithm. Hybrid techniques address this challenge by effectively amalgamating recommendations obtained from different recommendation approaches, such as collaborative filtering and content-based recommendation. Consequently, hybridization is frequently employed in the execution of recommender systems. A noteworthy application of hybrid recommenders is in mitigating the cold-start problem. Moreover, hybrid mechanisms possess the potential to augment the diversity of recommendations by integrating various recommendation approaches (Lü et al., 2012; Harrathi et al., 2017).

2.3 Pattern mining

Pattern mining revolves around the identification of valuable and interesting patterns within databases. The significance of pattern techniques lies in their ability to uncover concealed patterns within extensive databases, making them comprehensible to humans and, consequently, beneficial for decision-making and data interpretation (Fournier-Viger et al., 2017).

Frequent pattern mining has been a focal point in data mining research for more than 10 years. Frequent patterns denote subsequences or sets of items that occur in a dataset with a frequency equal to or exceeding a user-defined threshold. To illustrate, consider “milk” and “bread” forming a set of items that commonly appear together in transactions within a dataset (Han et al., 2007).

2.3.1 Association rules

Association rules are rule-based techniques designed to uncover relationships among variables in a dataset, particularly in a set of transactions with various variables. The Apriori Algorithm is a key method used for generating desired association rules in relational databases. This algorithm works iteratively, starting with the identification of individual frequent items and gradually expanding them one item at a time into larger itemsets, ensuring they meet specified minimum support and confidence criteria (Singh et al., 2013).

2.3.2 Sequential pattern mining

A sequence database comprises elements or events arranged in a specific order, with or without a clear temporal aspect. Sequence data finds applications in various domains, including customer shopping patterns, biological sequences, and web clickstreams. Sequential pattern mining involves identifying regularly occurring ordered events and designating them as patterns. Unlike association rules, where the order of items is not crucial, sequential pattern mining specifically focuses on the sequence of items (Han et al., 2007).

2.4 Recommender systems for e-learning in literature

The utilization of recommender systems in the context of e-learning has gained prominence in the literature. Various approaches have sought to leverage users' histories for generating recommendations. In this section, we will examine research that explores recommendation possibilities without considering knowledge as a factor.

Many recommender systems took of fuzzy algorithm a way of enhancing their suggestions. For example, Lu (2014) proposed a framework for a personalized learning recommender system, where two technologies have been included: a multi-agent attribute evaluation method for justifying students' need, and a fuzzy matching method to figure out the appropriate learning materials that best fulfill each student's needs. However, there was no

implementation or experiment for this model. [Antony Rosevelt and Arokia Renjit \(2020\)](#) also, utilized fuzzy temporal decision trees in their proposed content-based recommendation algorithm, in order to offer accurate content for learners based on their level of understanding.

In another work, [Niknam and Thulasiraman \(2020\)](#) used Fuzzy C-Mean algorithm for clustering learners and bioinspired ant colony optimization algorithm to find the most suitable learning path for learners, taking into account their prior improvements. Whereas [Wu and Feng \(2020\)](#) developed a network education knowledge recommendation system based on an improved neural network-path sorting algorithm with fuzzy uncertainty.

Moreover, [Rahman and Abdullah \(2018\)](#) designed a learner-centered recommendation system based on fuzzy clustering methods and decision tree to categorize learners into beginner, intermediate and expert depending on their academic achievement and learning behaviors.

Several papers worked on improving the performance of collaborative filtering techniques using machine learning methods. [Chee et al. \(2001\)](#) built a collaborative filtering method with enhanced efficiency, called Recommendation Tree (RecTree), to address the scalability problem by a divide-and-conquer approach. The technique first applies a k-means-like clustering, then implements subsequent clustering upon partitioned, smaller databases.

Additionally, [Li and Chang \(2005\)](#) described a personalized e-learning system using collaborative filtering, based on the IEEE Learning Technology Systems Architecture to adapt to learner's interests. The paper suggested a feedback extractor with fusion capability that combines multiple feedback measures for inferring user preferences.

A collaborative filtering recommendation system was proposed by [Liu \(2017\)](#) according to the influence of e-learning groups' behavior. From another perspective, [Dwivedi et al. \(2017\)](#) grounded their work on genetic algorithm enhancing the collaborative filtering approach, to provide learners with optimized learning paths using variable-length genetic algorithm.

From the idea that successful students, or students with high grades has followed the right path in selecting courses, [Bobadilla et al. \(2009\)](#) proposed a collaborative filtering recommendation system, with users having greater knowledge have greater weight in the equation of calculating the recommendations. Where knowledge is defined by the scores achieved by every user in a number of level tests. Similarly, the recommender system by [Ghauth and Abdullah \(2010\)](#) took students' achievements as a measure. However, it's a content-based filtering along with good learners rating for increasing learners' performance, where the results show that such systems outperforms content-based filtering with traditional collaborative filtering techniques.

Hybrid systems were thoroughly studied for the purpose of e-learning. The recommendation system that's discussed in [Salehi and Kmalabadi \(2012\)](#) is a hybrid system, combining content-based filtering and collaborative filtering techniques. The system models the learners and materials in a multidimensional space to facilitate recommendations. They used a set of material attributes to model the learning materials in a multidimensional manner. The materials attribute were considered also in modeling learners.

Learners rated multiple materials, and the ratings were used to identify the learners attributes based on the materials they liked. The approaches involved in making recommendations were content-based, collaborative-based and hybrid-based approaches. [Yang et al. \(2010\)](#) assumed that users evaluate various core concepts in accordance with domain knowledge, as there is a degree of correlation amongst core concepts, which means that users' interests have similarities as well. Thus, similarity between core concepts can be used to calculate the similarity between users. For that research, users provided ratings to the learning materials, where these ratings were used to compute learners' similarities. Then users' preference level for each interest category of the closest neighbors, and lastly nearest neighbors' interests are used to provide recommendations. These methods rely mainly on users' ratings, which might work for a partial cold-start, but will not be useful in pure cold-start condition.

[Lalitha and Sreeja \(2020\)](#) provided guidelines for Personalized Self-Directed Learning Recommender System (PSDLR), consisting of three steps. First, to classify learning materials with collaborative based and content-based filtering methods based on advancement level. Second, to assess learners' abilities and prior knowledge. Finally, to collaborate these two modules to generate the recommendation results. Collaborative and content-based filtering were also the basis of the work by [Benhamdi et al. \(2016\)](#), where they presented a New multi-Personalized Recommender for e-Learning.

In regard of collecting learners' information, [Bhaskaran et al. \(2021\)](#) proposed to apply a cluster-based linear pattern mining algorithm for extracting patterns of the learners, then to provide recommendations by the evaluation of ratings of frequent sequences.

[Sikka et al. \(2012\)](#) outlined an approach for building a software agent which can construct an online user model describing the user's behavior, and uses this model to suggest relevant activities and shortcuts. The recommendations aim to improve users' performance depending on the behavior of successful learners. [Ali et al. \(2022\)](#) designed their architecture with the help of virtual agent as well, for building semantic recommendations based on user needs and preferences, which helps in finding the most suitable materials in a real-world setting. Agents were also present in the work for [Amane et al. \(2021\)](#), who implemented a content-based course recommender system with the help of a multi-agent for university online learning platforms.

2.5 Knowledge-based e-learning recommender systems

Knowledge-based systems provide a remedy for the cold-start problem, and researchers have invested significant efforts in enhancing e-learning recommendations by incorporating knowledge.

In references [Obeid et al. \(2018\)](#), [Bouihi and Bahaj \(2019\)](#), [Brik and Touahria \(2020\)](#), [Chebbi and Imen \(2021\)](#), models were examined to identify the optimal assessment pathways according to the learner's needs and profile. These models were constructed

using ontology-based methods and semantic rules, resulting in improved recommendations for e-learning systems. However, additional experimental evidence is required to further validate these systems.

Sequential pattern mining has been integrated with context-aware and knowledge-based systems, as discussed in references (Tarus et al., 2017a; Shanshan et al., 2021). These systems leverage a hybrid recommendation approach that combines collaborative filtering, sequential pattern mining, and context awareness to address the cold-start problem.

In the realm of e-learning recommender systems, the application of ontology is believed to offer a higher degree of personalization, overcoming limitations found in traditional recommender systems. Despite its advantages, ontology design is a time-consuming process requiring knowledge engineering. As a result, hybrid systems have emerged as a prevalent solution in research to enhance performance by amalgamating multiple recommendation strategies (George and Lal, 2019).

While progress has been made in e-learning recommender systems, there remains room for improvement. It is suggested that machine learning techniques could contribute to handling implicit and explicit feedback to enhance accuracy (Kulkarni et al., 2020). A recent study on knowledge-based recommender systems (Rahayu et al., 2022) notes that the proposed models and prototypes are still being evaluated using experiments with a specific population size of non-real students, involving descriptive and inferential statistics, questionnaires, and observations. Therefore, future analyses should broaden the scope to include more participants and employ proper testing methods.

Additionally, in Ompusunggu et al. (2021), it is demonstrated that in the context of recommender systems, machine learning outperforms rule-based approaches in certain aspects, while rule-based approaches excel in others. Hence, despite the undeniable capability of machine learning, there is still a demand for rule-based approaches. Creating a well-performing system may involve a hybrid approach that combines both machine learning and rule-based techniques.

2.6 Student and material modeling in literature

Creating a student profile involves representing a student's information in a standardized format to facilitate analysis. Learner data can be gathered in two ways: explicitly, by directly collecting information from the learner, or implicitly, by monitoring their activities through a learning platform. A well-constructed student profile can be easily tailored for each student based on their preferences. In the personalization of e-learning systems, learner models play a crucial role, as they guide the recommendation of learning resources based on the characteristics derived from the learner model (Ciloglugil and Inceoglu, 2018). Various features are utilized in recommender systems for personalization, encompassing learners' preferences,

goals, prior knowledge, and motivation level (Jeevamol and Renumol, 2021).

2.6.1 Learning style

Learning styles play a vital role in tailoring online learning experiences to individual preferences. As defined by Keefe, a learning style is a combination of cognitive, affective, and psychological factors that serve as relatively stable indicators of how a learner perceives, interacts with, and responds to the learning environment (Keefe, 1979). Karen and Felicetti (1992) further describe learning styles as the educational conditions under which a student learns best, emphasizing the importance of how students prefer to learn rather than what they actually learn. Various operational and conceptual models exist for learning styles, including Kolb, Felder–Silverman, Myers–Briggs, Honey and Mumford, and VARK, each offering different conclusions and conditions to enhance learning.

For this research, Neil Fleming's VARK model is adopted to define students' learning styles. VARK, an acronym for Visual, Aural, Read/write, and Kinesthetic styles, was introduced by Fleming and Mills, drawing on experiences of students and teachers (Fleming and Mills, 1992). In the VARK model, the Visual learning style represents information through maps, diagrams, charts, graphs, flow charts, and symbols. Aural learning style preferences involve information that is heard or spoken, with learners benefiting from group discussions, lectures, phone conversations, radio, and talking about concepts. Read/Write learning style individuals prefer information displayed as words, emphasizing text-based input and output such as reading and writing manuals, books, assignments, essays, and reports. Kinesthetic learning style suggests a perceptual preference for hands-on experiences and practice, encompassing demonstrations, case studies, videos, simulations, and movies (Bajaj and Sharma, 2018).

The VARK model is chosen for this paper as it is a practical tool for maximizing students' learning by accurately assessing the effectiveness of different knowledge acquisition methods. The model is grounded in two main ideas: firstly, teaching students using their preferred methods significantly impacts their success in processing information and education; secondly, students acquiring knowledge through their individual learning style leads to the highest levels of motivation, interest, and understanding (Hanurawan, 2017; Movchun et al., 2020). The VARK inventory provides metrics for the four modes, allowing individuals to have preferences ranging from one to all four modes (Hawk and Shah, 2007).

Determining the VARK learning style is accomplished through the VARK questionnaire, which has demonstrated its validity, good reliability, and has been widely adopted by many researchers.

2.6.2 Misconception

Another aspect that this study aims to explore is students' misconceptions. Recent papers have sought to identify misconceptions in specific fields through various methods, including implicit approaches that examine learners' knowledge using behavior monitoring agents, as well as explicit methods relying on tests administered within the system (Halim et al., 2021).

2.6.3 Learning object modeling

The other significant element of an E-learning recommender system is the learning object, and its modeling is deemed as crucial as that of students. As demonstrated in the earlier section, ontology has been extensively employed in modeling learning objects to address the cold-start issue (Jeevamol and Renumol, 2021). However, alternative methods have also been explored, involving the use of data mining and machine learning.

In Cakula and Sedleniec (2013), the characteristics that define a high-quality learning object were outlined. It was emphasized that learning objects should be considerably shorter than traditional learning units and self-contained, allowing each learning object to be utilized independently. Additionally, they should include descriptive data to facilitate easy identification.

2.7 Similarity measures

The precision of similarity computation holds a pivotal role in the performance of any recommender system (Joy and Renumol, 2020). Cosine similarity is a method used to calculate the similarity between two vectors within an inner product space. For two vectors, x and y , the cosine similarity function is represented by Equation 1:

$$\text{sim}(x, y) = \frac{x \cdot y}{\|x\| \|y\|} \quad (1)$$

Where $\|x\|$ is the Euclidean norm of vector x , and $\|y\|$ is the Euclidean norm of vector y . Essentially, the cosine similarity measure involves assessing the lengths of vectors. It calculates the cosine of the angle between vectors x and y . A cosine value of 0 indicates that the two vectors are orthogonal, signifying no match. As the cosine value approaches 1, the angle becomes smaller, indicating a greater match between the vectors.

In the case of vectors with binary-valued attributes, the interpretation of cosine similarity is based on common attributes. If object x possesses the i th attribute ($x_i = 1$), then $x^t \cdot y$ represents the number of shared attributes between x and y , and $\sqrt{x \cdot x \cdot y \cdot y}$ denotes the geometric mean of the number of attributes possessed by x and y . Therefore, $\text{sim}(x, y)$ becomes a measure of the relative possession of shared attributes. For this scenario, a variation of cosine similarity is presented in Equation 2:

$$\text{sim}(x, y) = \frac{x \cdot y}{\sqrt{x \cdot x + y \cdot y - x \cdot y}} \quad (2)$$

This variation defines the ratio of the number of common attributes in x and y , to the number of attributes possessed by either x or y (Han et al., 2012).

3 Materials and methods

The proposed system presents a comprehensive process for constructing and enhancing a hybrid attribute-based recommendation system. Figure 1 illustrates the overall architecture of the designed framework. The initial phase of the system involves building user and material models. The user model is built via an interactive user interface, which collects user

data and stores the generated profiles in database. Subsequently, these models are utilized in the recommendation component, which employs a hybrid approach to extract pertinent materials for a user. Users are then able to rate the suggested materials, and these ratings are employed to update the material profile through a background task based on the Apriori algorithm. Furthermore, our system has been encapsulated as a web-hosted application, featuring a user interface that integrates all the components, allowing users to interact with the system seamlessly. The subsequent subsections provide detailed descriptions of the various components within the proposed system.

3.1 Student and material modeling

Modeling students and materials is a crucial phase in the recommendation process, especially for our system, where the goal is to achieve efficient modeling in terms of time and storage. To create suitable profiles for both the learning object and the student, three key factors were considered. Firstly, selecting attributes that effectively define each object. Next, determining the appropriate representation for these attributes to facilitate matching between a student and a corresponding set of learning materials. Lastly, devising a method to collect these attributes in a user-friendly manner, avoiding any tedious tasks for users.

3.1.1 Student profile

As mentioned earlier, the creation of a student profile involves establishing a general model for users and populating this model with information specific to each user. The selected attributes for the student can be divided into two parts: personal information used for identification and attributes directly linked to the recommendation task, formatted in a way suitable for aiding the recommendation process.

The personal information in the model includes the user's name, personal image, and contact details such as email and phone number. Attributes relevant to making recommendations can be further categorized into learning preferences and knowledge-related attributes. Learning-preference attributes describe the optimal conditions in which a student learns better, encompassing the student's learning style. For this research, the VARK model was adopted to determine learning styles due to its accuracy in distinguishing between various means of acquiring knowledge effortlessly, involving multiple perceptual modes (Bajaj and Sharma, 2018).

Learning-preferences also encompass preferences related to material structure, extracted from the properties of the learning object. These attributes assist in identifying which characteristics of a learning object a student prefers. Attributes include the material format (text, video, slides, exercises, quizzes), material length, level of detail, language, clarity of agenda, and video theme. In selecting these attributes, input was gathered not only from the literature but also from direct inquiries with several students regarding the factors they consider in online materials and what influences their decision to study or reject certain materials.

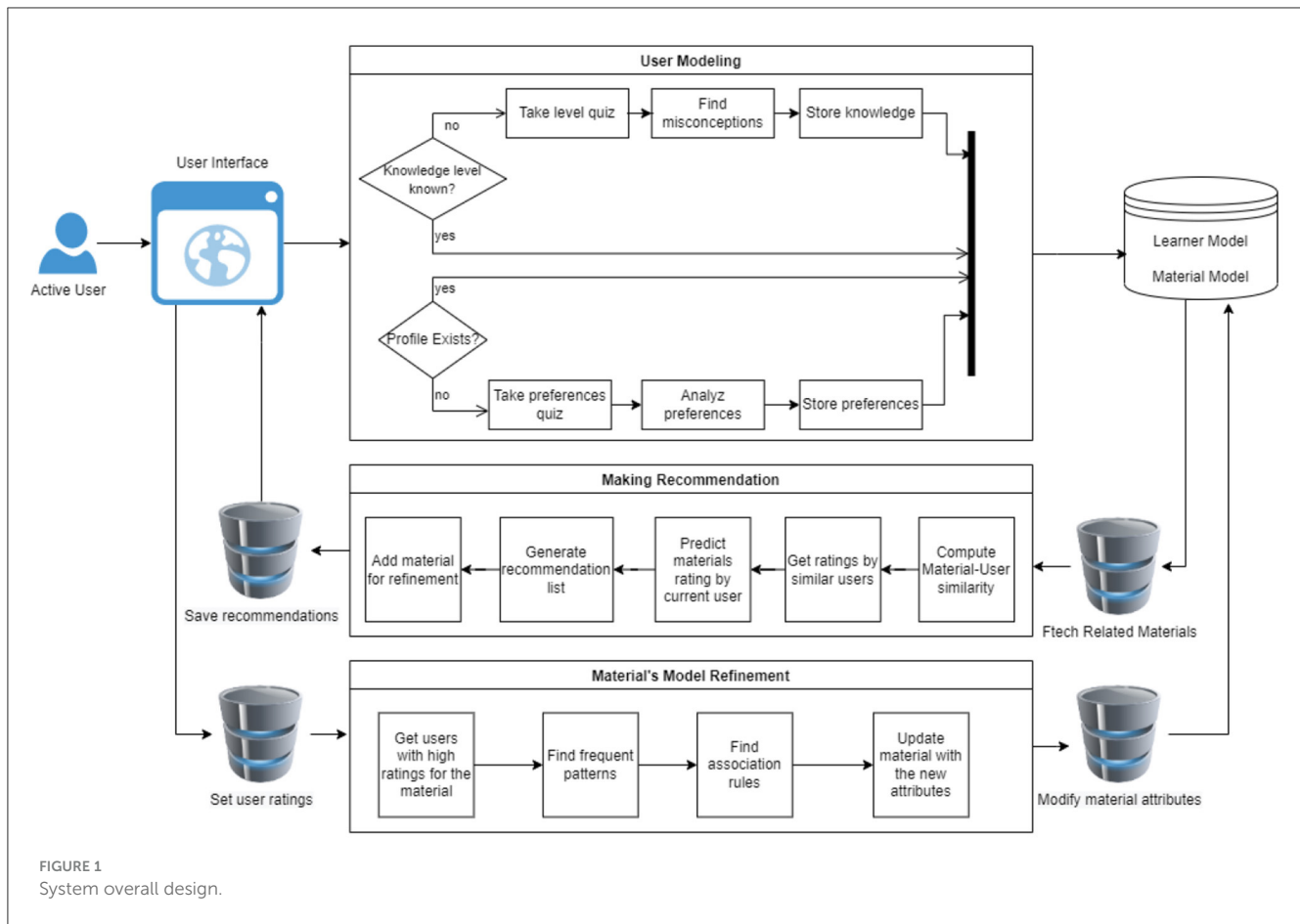


FIGURE 1 System overall design.

It's important to note that not every student fits exclusively into a single learning style; some may have a tendency toward two or more styles equally, known as multimodal (Movchun et al., 2020). Therefore, the system allows users to select more than one option per question, providing a comprehensive description of the dominant styles for a user.

The other part of the profile aims to characterize the student's knowledge in each topic, defined in terms of skill level and misconceptions. Misconception detection is crucial in the recommendation process, significantly impacting the content of the suggested materials. If a misconception is detected, the recommended materials focus on explaining that concept rather than presenting information the user already knows.

Before starting a particular topic a short multiple-choice quiz is provided, to assess the knowledge of the student in that topic. The quiz covers the main concepts in the intended topic. If a student has no prior knowledge, they can skip the quiz all together. The answers are analyzed to find out whether the mistakes are due to a misconception or lack of knowledge. Where some options are linked to misconceptions. If the majority of the selected answers hold a certain misconception, then this means that the user should work on correcting the identified misconception. Thus, the system will offer a narrative feedback with the final score, mentioning the knowledge level (poor, good, great) and the detected misconceptions in case of any.

3.1.2 Material modeling

Similar to the student profile, certain attributes of the learning material were utilized for identification and retrieval purposes, while others were essential for making recommendations. The primary set of material attributes was derived from the "IEEE Standard for Learning Object Metadata," encompassing data relevant for both identification and recommendation purposes. These attributes include title, subject, size, relation with other materials, requirements, duration, format (resource type), and language. Additionally, the misconception that a material addresses was also considered as one of the material properties. Furthermore, similar to how material attributes were incorporated into the student's model as preferences, the learning style that a material would cater to was included as part of the material attributes.

The system organizes knowledge into courses, where each course represents a general domain and encompasses a collection of inner topics. Topics correspond to the chapters within a course. Each topic is further composed of learning units, which are discrete pieces of information that a user needs to acquire in that specific topic. To facilitate knowledge acquisition, the system provides a set of alternatives for each learning unit, referred to as materials or learning objects. This is where the recommendation process comes into play. The system is tasked with selecting the most suitable option from these materials and covering only the necessary units

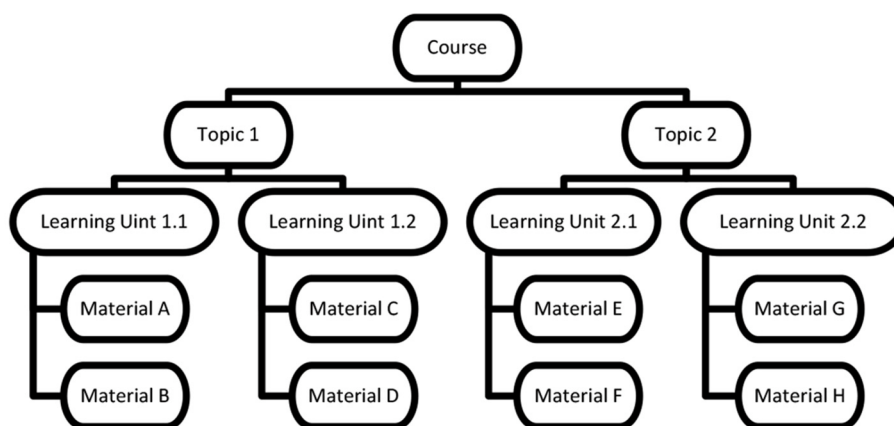


FIGURE 2
Materials organization.

based on the user's prior knowledge. The structure described is illustrated in [Figure 2](#).

The system takes into account the size and segmentation of the material as one of its attributes. Some students prefer learning a large topic in smaller chunks. Consequently, in certain instances, a sequence of learning materials collectively forms one learning unit. However, it's important to note that each material within this sequence provides several options. Consequently, a single material cannot be designated as a prerequisite for another. To address this issue, an order was assigned to the materials. Materials with the same order fulfill the same learning objectives but with different structures.

The materials were collected and labeled manually. Attributes were assigned based on our understanding and experience with the chosen course. We selected a course with which we are familiar and can distinguish which materials are suitable for the selected subtopics. The collected items were organized and integrated into the system in advance, prior to conducting any experiments.

3.1.3 Attributes representation

Given our focus on addressing the cold-start scenario with a knowledge-based element while exploring an alternative approach to ontology, we propose projecting the knowledge encapsulated in ontology into a two-dimensional array for each model. This array can be stored in a traditional Relational Database Management System (RDBMS).

As previously demonstrated, to facilitate the matching of students with materials, the system requires shared data between the learning object and the learner. This means aligning what a learner is seeking with what a learning object offers, while also incorporating exclusive data to identify each object. Therefore, both the material and the student can be represented using two-dimensional vectors. In this representation, columns denote attributes, and rows correspond to either material or student records.

Certain columns may have multiple values, and these values are concatenated using a star symbol (*) and stored in the same cell.

This approach aligns with the designed algorithm, which will be elaborated on later.

3.2 Making recommendations

Once the preferences quiz is completed, and the knowledge level for a specific topic is established, the student model related to that topic is considered prepared. This information can now be utilized to recommend suitable learning objects. ' Given that the learning objects in the proposed system are organized to be browsed by topic, the user must select a specific topic from a course. Only the learning objects under that chosen topic are then taken into account for recommendations. The data of the active student, encompassing preferences and knowledge in the selected topic, is retrieved and stored in a Panda's DataFrame. Simultaneously, the data for the target materials is retrieved and saved in another DataFrame. To refine our dataset, we leverage the knowledge assessment quiz by extracting learning components that address any misconceptions identified in the user. Otherwise, all the materials under the selected topic will be retrieved.

3.2.1 Data encoding

We established an intersection dataframe that incorporates both the retrieved materials and the student. This dataframe consists solely of common attributes shared between the two types of objects, with the student consistently stored as the last record. The common attributes encompass "length," "contentType," "detailed," "bullets," "videoTheme," "isSeparated," "withAgenda," "learningStyle," and "language_pref." Additionally, the original record IDs were preserved in the intersection table to facilitate the merging process with the initial data frames.

All of these common properties are categorical variables with string values that are easily understandable and interpretable for us but lack meaning for the machine, as the words themselves hold no significance to the algorithm. Particularly when comparing a cell with multiple values to another with either multiple or a single

value, the outcome of the comparison will consistently be negative. For instance, the value “RV” does not match the value “VK,” even though they share a sub-value. Furthermore, the order of values matters for the machine but is inconsequential for the system. For example, “RV” and “VR” are entirely different in their current state but have the exact same semantic meaning. Overcoming such challenges in the current representation would demand extensive processing to identify sub-patterns and disregard the order. Therefore, an efficient solution for a more straightforward representation involves encoding the data using binary encoding. Therefore, we expanded the joining dataframe by hot-encoding each attribute to convert our categorical variables into multiple binary columns, with the following preprocessing steps:

1. The values in a multivalued column were separated by the star “*”, which is the concatenating character that was mentioned earlier.
2. The values were turned into binary using the function `get_dummies` from Python’s pandas operations, where the values that are present in each record were set to one and the rest were zeros.
3. For each value the suffix “_attributeName” was added, to end up with columns in the pattern of “value_attributeName”.
4. The value that appears for a record, are given the value “1” in the corresponding column in the binary-encoded table, and “0” otherwise.

As an illustration, consider the `learningStyle` attribute with potential values such as V (for Visual), A (for Aural), R (for Reading), and K (for Kinesthetic). In the binary dataframe, the corresponding columns would be `learningStyle_V`, `learningStyle_A`, `learningStyle_R`, and `learningStyle_K`. For a learner exhibiting visual and reading learning styles, the value would be V*R, which is represented in binary as 1010. Figure 3 provides a depiction of example records in their original state before encoding, while Figure 4 showcases the binary-encoded outcomes of these records.

3.2.2 Content-based filtering

Incorporating the knowledge outlined in the learner profile is essential for tackling the cold-start problem. It establishes the foundation for matching the initial learner properties with the characteristics of learning materials. The advantage of utilizing a Relational Database Management System (RDBMS) lies in its capacity to store various attributes in a unified manner and enable querying and combining them within the same query. In the proposed model, the similarity between a learner and a specific learning object is computed using the Rogers-Tanimoto coefficient, which is applied with bit-vectors, considering both the presence and absence of attributes. In other words, both ones and zeros are taken into account when calculating the similarity. For two vectors S and M of length n , the similarity is defined as illustrated in Equation 3:

$$Sim(S, M) = 1 - \frac{2(C_{TF} + C_{FT})}{C_{TT} + C_{FF} + 2(C_{TF} + C_{FT})} \tag{3}$$

Where:

- F refers to the bits zeros and T refers to the ones.
- C_{ij} is the count of cases where $S[k] = i$ and $M[k] = j$ for $k < n$

The computed similarity result between each learning object and the student is associated with that particular object. This information will be utilized in subsequent stages to predict the rating and determine the final score for each material. The materials with the highest similarity to the student are identified and passed on to the next phase, which involves collaborative filtering.

3.2.3 Collaborative filtering

In conventional collaborative filtering, user similarity is determined based on their ratings. Users with similar ratings for the same set of items are considered similar. However, given the cold-start problem, where a new user lacks ratings, we leverage the existing knowledge base to identify users with similarities to the current one. To address this, we narrow down the set of users by considering only those who have rated each of the best-matching materials identified in the previous step. The similarity between the current user and users from this reduced dataset is measured using Jaccard similarity, employing the same binary encoding approach as described earlier. Jaccard similarity quantifies non-zero similarity as the ratio of the intersection of two bit-vectors to their union. The pairwise similarity between a student vector (S) and each vector of other users (O) can be expressed as depicted in Equation 4:

$$Sim(S, O) = \frac{S.O}{S^2 + O^2 - S.O} \tag{4}$$

where this representation comes from the fact that the product of bit vectors (each dimension takes the value of either 0 or 1) is equal to their intersection

$$S.O = \sum_i S_i O_i = \sum_i (S_i \wedge O_i)$$

and

$$||S||^2 = \sum_i S_i^2 = \sum_i S_i$$

To identify similar users, a specific threshold was set, wherein users with a Jaccard score exceeding a certain level are considered, as opposed to obtaining the top K most similar users. The threshold approach is employed because the most similar user may not be sufficiently alike to draw conclusions about the current user. Simply relying on a few users who have already rated a material might result in different preferences from the target user. Consequently, predicting ratings for the active user based on the available data could lead to inaccurate predictions.

3.2.4 Ranking and scores predictions

To generate the final list of recommendations, for each material a score is calculated as seen in Equation 5

$$Score_{M_i} = R * Sim(S, M_i) \tag{5}$$

	length	contentType	detailed	bullets	videoTheme	is Seperated	withAgenda	learningStyle	language_pref	id
0			NaN	0		NaN	None		None	1
1	L	V*E*	NaN	1	*S*U*P	0.0	None	V*K*	None	2
2			NaN	0		NaN	None		None	3
3	S	V*	0.0	1	*P	1.0	None	R*K*	A*E	4
4	L		0.0	0	*U	0.0	None	V*K*	A*E	5
...
82			NaN	0		NaN	None		None	86
83	S	V*	0.0	0	*S	NaN	None	K*	A*E	87
84	M	E*	0.0	1	*S	NaN	None	R*	A*E	88
85	S	V*E*	1.0	0	*S*U*P	1.0	None	K*	E	89
86	S	E*	0.0	0	*P	1.0	None	R*	A*E	90

FIGURE 3 Values before encoding.

	L_length	M_length	S_length	E_contentType	T_contentType	V_contentType	0.0_detailed	1.0_detailed	nan_detailed	0_bullets	...	1.0_isSe
0	0	0	0	0	0	0	0	0	1	1	...	0
1	1	0	0	1	0	1	0	0	1	0	...	0
2	0	0	0	0	0	0	0	0	1	1	...	0
3	0	0	1	0	0	1	1	0	0	0	...	1
4	1	0	0	0	0	0	1	0	0	1	...	0
...
82	0	0	0	0	0	0	0	0	1	1	...	0
83	0	0	1	0	0	1	1	0	0	1	...	0
84	0	1	0	1	0	0	1	0	0	0	...	0
85	0	0	1	1	0	1	0	1	0	1	...	1
86	0	0	1	1	0	0	1	0	0	1	...	1

FIGURE 4 Values after encoding.

where $Score_{M_i}$ is the material’s score, $Sim(S, M_i)$ is the similarity between the user and the material M_i and R is the weighted average of ratings by the similar users to the material M_i that is given by Equation 6

$$R = \frac{\sum_j R_{ji}^* Sim(S_j, S)}{\sum_j Sim(S_j, S)} \tag{6}$$

Where R_{ji} is the rating of material M_i by student S_j and $Sim(S_j, S)$ is the similarity between the target student and student S_j . If a material does not have any ratings yet, then the average rating is assumed to be 5 to keep its chances of being selected and later rated.

Now, each material has a score derived from the similarity between the active user and the material, based on ratings by other similar users. This score is utilized to arrange the materials, along with a randomly generated integer assigned to each record. The random number serves to provide equal opportunities for materials that were never selected, as it is used for reordering instead of

adhering strictly to the order returned by the database. In other words, if two materials receive the same score, the random number determines which one will be selected, reducing the likelihood of consistently choosing the same material. It’s important to consider that multiple resources may discuss the same idea and hold the same logical position in the learning sequence. Therefore, for each idea, only the resource with the highest score is chosen. Since the list is sorted, an efficient selection process is implemented by retaining the last occurrence of each unique element and discarding the rest. Ultimately, a list of highly ranked resources is obtained, ensuring comprehensive coverage of all aspects of the topic. Table 1 outlines the steps taken within our algorithm in a cold-start condition.

3.3 Refinement/adjusting component

To enhance the system’s performance, this study proposes an approach for adjusting the parameters of low-rated materials through association rule mining. Before presenting the final list to

TABLE 1 Recommendation algorithm.

Algorithm 1: Generating recommended list
<p>Input: Active learner's preferences vector. Set of learning materials $M = \{m_1, m_2, m_3, \dots, m_n\}$. Similar users' ratings R, where $r \in \{1, 2, 3, 4, 5\}$</p> <p>Output: List of top N recommended materials Predicted rating for each material</p> <p>Method:</p> <ol style="list-style-type: none"> For each material m_i in M, compute similarity between m_i and the active user S, $\text{Sim}(S, m_i)$ using Equation 3 For each material m_i in M <ol style="list-style-type: none"> Compute similarity between the active user and each user S_k of users who rated material m_i, if similarity > 0.6 add rating r_{ki} to the list of ratings R using Equation 4 Compute the ratings weighted average using Equation 5 Compute the final predicted rating using Equation 6 Sort materials based on predicted score and a random number Keep the material highest predicted rating for each piece of information

the user, a material with a low rating (below 3) and no significant relation to the current user (i.e., not having a high score concerning the active user) is added to the list. The ratings provided by the current user and other unrelated users are employed by a background task to assess if an enhancement can be applied to such a material.

The background task selects users who rated a material higher than 3 and employs the Apriori algorithm for pattern mining. Once again, users are binary encoded, with the material's rating being one of the attributes. The Apriori algorithm is applied to the binary data to identify frequent items with a minimum support of 0.5. Association rules related to the ratings are derived from the frequent items, ensuring a confidence level higher than 0.7.

Attributes from the association rules are then incorporated into the material's profile. The attributes observed in this step are those initially obtained from user properties, such as learning style. Each time the attributes of a material are updated, the values are logged in a history table for verification and testing purposes. Despite these attributes being assigned to the material by the admin user, there may be inaccuracies. For instance, a video material might be assumed initially to cater to learners with a visual learning style, but in practice, it may be more effective for learners with an audio learning style, as the content focuses on verbal explanation. For attributes that have a single value, the values are extracted from the association rule containing that attribute with the highest confidence.

This process is triggered when a material obtained for a user for adjustment purposes receives a new rating. The calculations are performed in the background to prevent blocking user operations and are separated as a background task that can run periodically as a cron job, rather than being triggered every time a new rating is given.

Figure 5 illustrates a sample of the Apriori algorithm output. It shows each of the material characteristics in the antecedent column, and the set of associated learning preferences in the consequents column. In this example, only one user has rated the material, resulting in 100% confidence and support. This example is used for illustrative purposes to elucidate the outlined steps. After running the Apriori algorithm, the data is preprocessed to

display only the records where "rating" appears in the consequents columns, either alone or with other parameters. Subsequently, the antecedents of these rows are processed. If an attribute should take a single value, the record with the highest confidence for that attribute is considered, followed by string processing to extract the attribute alone and separate the value from the attribute name (using underscores). This process yields the desired attribute value. If the attribute can take multiple values, each occurrence of that attribute has its value extracted using the same string processing steps, concatenating all values with a star "*".

3.4 User interface and prototype

To streamline the system's experiment and provide a user-friendly experience, a website was developed to encompass all project phases. The website is designed to be intuitive and self-explanatory, requiring minimal guidance for users to navigate through the system. This approach is particularly beneficial as the system is tailored for self-directed learners.

Hence, the content was organized for easy viewing and editing, allowing users to quickly scan through the homepage using either the dashboard or the navigation menu. Additionally, warning messages regarding any missing requirements, such as the preferences quiz, include navigation links to guide users in completing those sections.

For the proof of concept, we focused on four key topics from the "Digital Systems Design" course: minterms minimization, maxterms minimization, prime implicants, and decoder. We gathered a collection of 65 materials covering these topics, with variations in length, structure, and language. The resources were categorized, labeled, and stored in our database based on our learning material modeling.

4 Results and discussion

The testing aimed to assess the accuracy of material recommendations generated by the system. Various evaluation approaches were employed, each tailored to the nature of the element being tested. The proposed recommendation algorithm underwent evaluation through different metrics to determine:

1. The system's ability to predict users' ratings, including the cold-start condition.
2. The system's capability to identify similar users, also considering the cold-start condition.

An experiment was conducted to evaluate the effectiveness of the recommendations, involving 43 participants. As there was no existing dataset aligning with our material mapping, we utilized the 65 learning objects collected and annotated based on our defined modeling. Each participant completed the learning preferences quiz, knowledge assessment quizzes, and rated the materials suggested by the system. This resulted in a total of 448 ratings. The participants were university students who had completed the digital design course during the COVID-19 pandemic closure.

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
18	(S_length)	(4.0_rating)	1.0	1.0	1.0	1.0	1.0	0.0	inf
36	(E_contentType)	(4.0_rating)	1.0	1.0	1.0	1.0	1.0	0.0	inf
52	(0_detailed)	(4.0_rating)	1.0	1.0	1.0	1.0	1.0	0.0	inf
67	(0_bullets)	(4.0_rating)	1.0	1.0	1.0	1.0	1.0	0.0	inf
78	(P_videoTheme)	(4.0_rating)	1.0	1.0	1.0	1.0	1.0	0.0	inf
...
173047	(1_IsSeperated)	(R_learningStyle, 4.0_rating, 0_detailed, S_le...	1.0	1.0	1.0	1.0	1.0	0.0	inf
173048	(0_bullets)	(R_learningStyle, 4.0_rating, 0_detailed, S_le...	1.0	1.0	1.0	1.0	1.0	0.0	inf
173049	(E_language_pref)	(R_learningStyle, 4.0_rating, 0_detailed, S_le...	1.0	1.0	1.0	1.0	1.0	0.0	inf
173050	(E_contentType)	(R_learningStyle, 4.0_rating, 0_detailed, S_le...	1.0	1.0	1.0	1.0	1.0	0.0	inf
173051	(P_videoTheme)	(R_learningStyle, 4.0_rating, 0_detailed, S_le...	1.0	1.0	1.0	1.0	1.0	0.0	inf

FIGURE 5 Apriori output.

4.1 Mean absolute error (MAE)

The mean absolute error (MAE) is the average of the absolute differences between the predicted values generated by the model and the actual results. This error is computed by taking the absolute value of the difference between the actual and predicted values for each individual record and then calculating the mean of all these differences. The absolute value is utilized because the focus is on the variation irrespective of its sign (Willmott and Matsuura, 2005).

In our context, the predicted values are the ratings generated by the model, and the actual values are the ratings provided by users. The MAE value indicates how much the predicted rating deviates from the actual rating. A smaller MAE value signifies less deviation and better accuracy. The mathematical representation of MAE is given by Equation 7:

$$MAE = \frac{\sum_{i=1}^n |y_i - x_i|}{n} \tag{7}$$

The model-generated scores, utilized for ranking materials and indicating the predicted rating for each user-material pair, were normalized within the 1 to 5 range. The error analysis involved three scenarios: overall error, error in the absence of similar users' ratings, and error when considering ratings by similar users (referred to as collaborative filtering or CF).

The computation of the overall mean absolute error (MAE) considered all 443 records, resulting in an error of 1.05, which is relatively small. Given the five possible rating values, if a user assigns a rating of 3, the system may predict a rating of either 2 or 4.

For the second case, the error was calculated when only records with the collaborative filtering flag set to true were considered. This flag indicates situations where users similar to the active user had already rated the target material, and their ratings were part of the score calculation process. With 154 records, the resulting error was 1.03.

TABLE 2 MAE for the model.

Case	MAE
All records	1.05
Records with CF	1.03
Records without CF	1.09

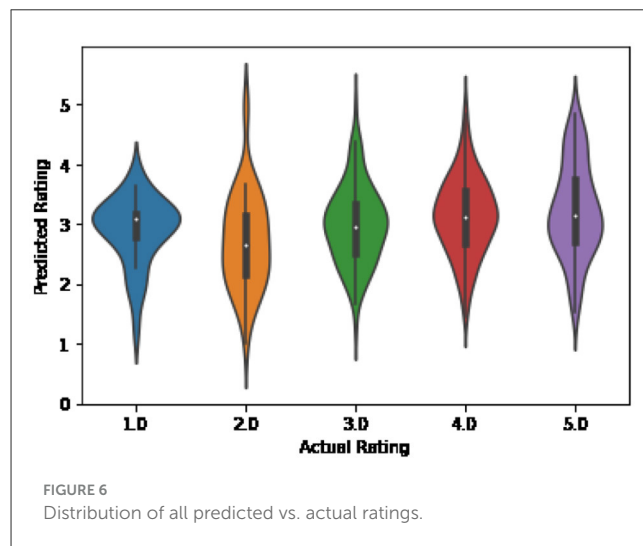
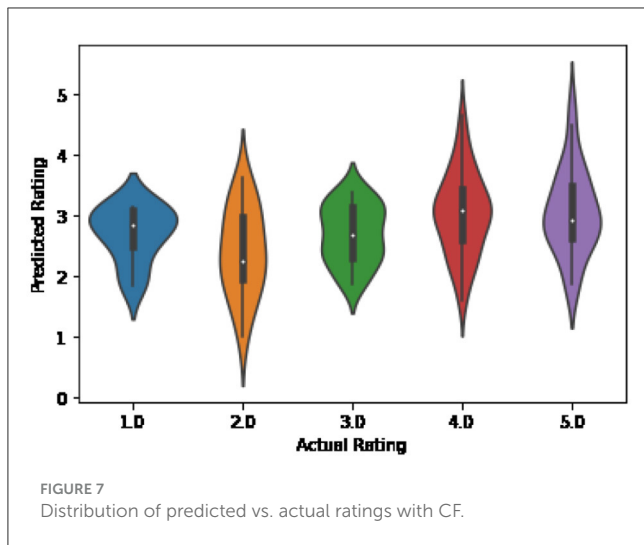


FIGURE 6 Distribution of all predicted vs. actual ratings.

In the third case, the opposite scenario was considered, where the user was the first among their similar group to rate the marked material. For these records, the error was around 1.09. The errors for the three cases are summarized in Table 2. Although there is a slight improvement observed with the inclusion of similar users' ratings, the change is not significant. The limited number of records



under study might contribute to this observation, preventing definitive conclusions.

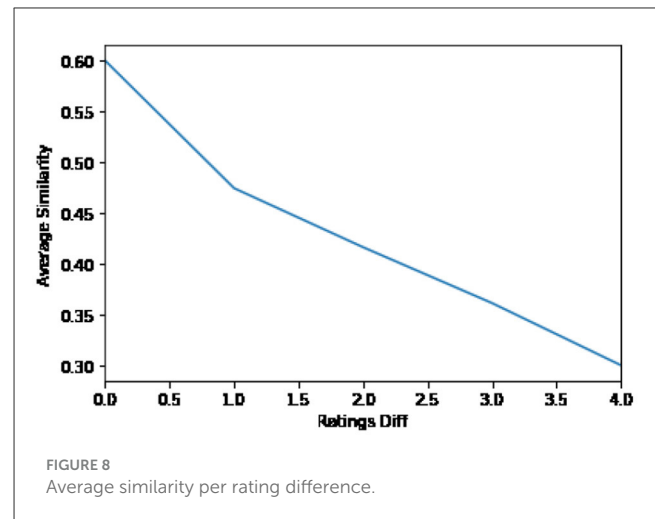
The impact of the proposed collaborative filtering approach was further investigated by visualizing the distribution of actual ratings vs. predicted ones. Figure 6 illustrates the distribution of all predicted values, while Figure 7 depicts the distribution when collaborative filtering is implemented. The visual representation utilized a violin plot, combining features of box and density plots. The plot showcases the minimum and maximum values, along with the interquartile range represented by the black rectangle in the center. The width of the plot sections is determined by kernel density estimation, reflecting the data's distribution shape. Wider sections indicate a higher probability of instances taking on that value, while narrower sections imply lower probability.

Upon comparing the two graphs, it becomes evident that, in the absence of considering similar users' ratings, there are some extreme values (depicted by the peak values of the plots) in the wrong direction, which are not present when collaborative filtering is employed. For instance, for an actual rating of 2, Figure 7 shows a predicted value of 5. Additionally, the actual rating of 3 is widely spread across all possible predicted values in the first graph, whereas in the second graph, it is more compact and centered around the value of 3.

In this research ratings <3 refer to irrelevant material suggestions, and materials with ratings of 3 or above are assumed to be well categorized. The total number of records with good ratings is 363 out 443, which refers to a level of satisfaction of around 82%. Comparing this result to previous work, it shows an increase of 16% of satisfaction for models considering cold-start problem (Jeevamol and Renumol, 2021).

4.2 Users similarity

The next evaluation metric is the capability of the system in identifying similar users. To do so, we analyzed the users' similarities with the consistency in their ratings, as we also used multivariate clustering to test similar users groups.



4.2.1 Users' similarity vs. ratings

This investigation seeks to assess the divergence in ratings provided by similar users for a specific material. Ratings were categorized by material, and a similarity matrix among users within each group was established, incorporating the rating differences between every pair of users. The same similarity function employed in our model was used for this matrix. A DataFrame was then generated, with each row containing the similarity between two users and the rating difference for a particular material. Given the rating scale from 1 to 5, the difference ranges from 0 to 4.

The records in the newly created DataFrame were grouped by rating difference, and the average similarity among users in each group was computed. Figure 8 visualizes the relationship between the difference in ratings and the average similarity among users for each difference level. The figure suggests that smaller rating differences correspond to higher similarities among users in that category. This observation aligns with traditional collaborative filtering, which identifies related users based on the proximity of their ratings. Thus, we were able to identify similar users in a pure cold-start condition without existing ratings. Moreover, our method can seamlessly integrate classical collaborative filtering as the system progresses in hot-start settings.

4.2.2 Using clustering for learner similarity

The second method employed to evaluate the model's capability to identify similar user groups involves the application of a clustering algorithm. Now that a sufficient number of users have interacted with the system, a multivariate clustering algorithm is employed to associate users with similar characteristics. Multivariate clustering is an analytical approach designed to explore patterns within multiple variables simultaneously. For our evaluation, the K-means algorithm was chosen to cluster users. The clustering algorithm was fed with 200 records, encompassing all users in the database, not exclusively those who have rated some materials. Including the maximum number of users is preferred in machine learning algorithms.

K-modes clustering, an enhancement of k-means tailored for categorical data, was utilized. It is an unsupervised machine

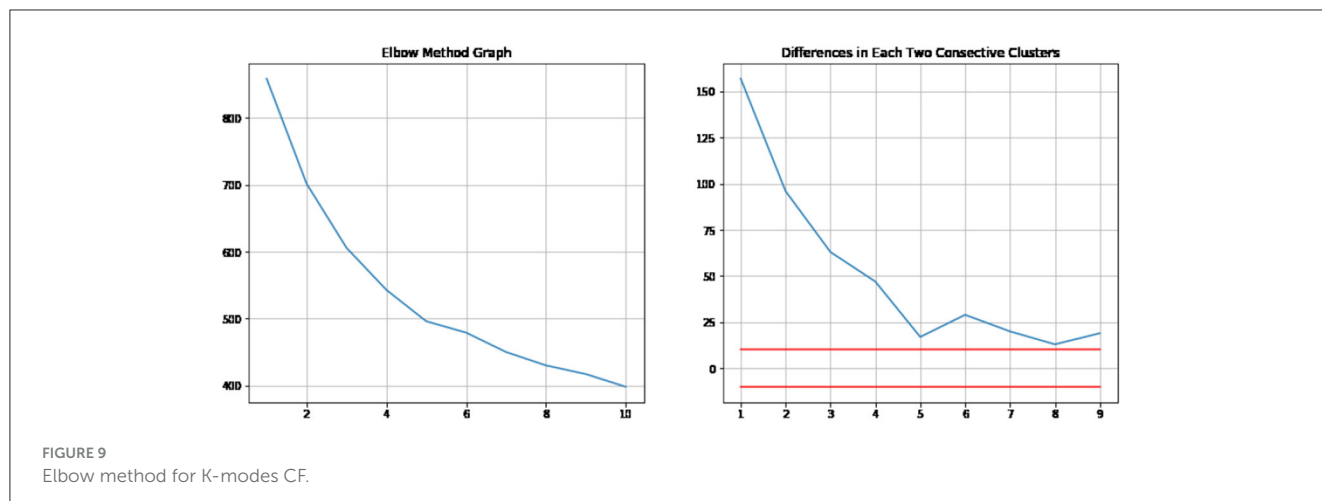


TABLE 3 Similarity in each cluster.

Case	MAE
1	0.7
2	0.7
3	0.67
4	0.8
5	0.9

learning algorithm that partitions the data into k clusters and identifies the closest cluster when adding an entry. The essential parameters for this clustering technique are the number of clusters (K) and the set of variables (attributes) to analyze. The selected attributes mirror those used in our model, describing learning preferences and learning style.

To determine the appropriate number of clusters, the elbow method was employed. This method involves running the clustering algorithm multiple times with different numbers of clusters and assessing the variance in groups for each cluster number. When plotting the variations within clusters against the corresponding number of clusters, the elbow of the curve is identified and serves as the optimal value “ k .” The elbow of the curve signifies the point at which the curve noticeably bends, indicating that the decrease in variance beyond that point is not significant. In clustering, this implies that increasing the number of clusters beyond a certain point does not significantly enhance the model. The elbow graph shown in Figure 9 led to the selection of 5 as the preferred number of clusters.

The algorithm employed in our model was used to compute pair-wise similarities among users within each cluster. The mean of these similarities was then calculated to assess the homogeneity that our system attributes to these clusters. The results, presented in Table 3, all satisfy our similarity threshold of 0.6.

Based on this analysis, it can be inferred that the system’s similarity measure can be substituted with a multivariate clustering mechanism, specifically K-modes. As we move beyond the cold-start scenario, defining similar users can be accomplished by

identifying the relevant cluster and determining the average rating given by users in that cluster.

5 Conclusion and future work

In this paper, a model was introduced for both learners and learning objects, incorporating common attributes shared between the two models. These models encompassed learning preferences, learning styles, and knowledge. The proposed model was employed in a recommendation system aimed at identifying similar learning objects for a given learner. The system utilized common attributes between learners and learning objects, incorporating ratings from similar users to enhance the accuracy of predictions. The top-rated materials were then suggested to users, taking into account user misconceptions and relationships between materials. Additionally, the attributes of materials were adjusted using sequential pattern mining based on user ratings.

The system underwent experimentation to evaluate the effectiveness of recommended materials. The results indicated that the system successfully modeled users and identified their misconceptions. Evaluation based on predicted vs. actual ratings yielded a mean absolute error (MAE) of 1.05, suggesting that, on a five-point scale, predictions were off by an average of one point. Another aspect of the analysis involved assessing user similarities, revealing that similar users tended to rate the same materials more closely compared to dissimilar users. Lastly, K-Mode multivariate clustering was applied to a set of users, and users within each cluster exhibited significant similarities according to the applied similarity function. The system shows promise for future testing across diverse groups of students with varied backgrounds and enrolled in different courses spanning various disciplines. Additionally, machine learning algorithms can be employed to assign weights to the attributes of learners and learning objects, to increase the accuracy of the predicted ratings.

Data availability statement

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

Author contributions

HB: Investigation, Methodology, Software, Supervision, Validation, Writing – original draft, Writing – review & editing. AA-I: Conceptualization, Data curation, Investigation, Methodology, Supervision, Validation, Writing – original draft, Writing – review & editing.

Funding

The author(s) declare that no financial support was received for the research, authorship, and/or publication of this article.

References

- Abouzeid, E., Fouad, S., Wasfy, N. F., Alkhadragey, R., Hefny, M., and Kamal, D. (2021). Influence of personality traits and learning styles on undergraduate medical students' academic achievement. *Adv. Med. Educ. Pract.* 12:777. doi: 10.2147/AMEP.S314644
- Ali, S., Hafeez, Y., Humayun, M., Jamail, N. S. M., Aqib, M., and Nawaz, A. (2022). Enabling recommendation system architecture in virtualized environment for e-learning. *Egyptian Informat. Journal* 23, 33–45. doi: 10.1016/j.eij.2021.05.003
- Amane, M., Aissaoui, K., and Berrada, A. (2021). Multi-agent M, and content-based course recommender system for university e-learning platforms. *Lecture Notes Netw. Syst.* 211, 663–672. doi: 10.1007/978-3-030-73882-2_60
- Antony Rosewelt, L., and Arokia Renjit, J. (2020). A content recommendation system for effective e-learning using embedded feature selection and fuzzy DT based CNN. *J. Intellig. Fuzzy Syst.* 39, 795–808. doi: 10.3233/JIFS-191721
- Bajaj, R., and Sharma, V. (2018). Smart Education with artificial intelligence based determination of learning styles. *Procedia Comput. Sci.* 132, 834–842. doi: 10.1016/j.procs.2018.05.095
- ben Schafer, J., Frankowski, D., Herlocker, J., and Sen, S. (2007). "Collaborative filtering recommender systems," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (Berlin; Heidelberg: Springer), 291–324.
- Benhamdi, S., Babouri, A., and Chiky, R. (2016). Personalized recommender system for e-Learning environment. *Educ. Inf. Technol.* 22, 1455–1477. doi: 10.1007/s10639-016-9504-y
- Bhaskaran, S., Marappan, R., and Santhi, B. (2021). Design and analysis of a cluster-based intelligent hybrid recommendation system for e-learning applications. *Mathematics* 9:197 doi: 10.3390/math9020197
- Bobadilla, J., Serradilla, F., and Hernando, A. (2009). Collaborative filtering adapted to recommender systems of e-learning. *Knowl. Based Syst.* 22, 261–265. doi: 10.1016/j.knsys.2009.01.008
- Bouihi, B., and Bahaj, M. (2019). Ontology and rule-based recommender system for e-learning applications. *Int. J. Emerg. Technol. Learn.* 14, 4–13. doi: 10.3991/ijet.v14i15.10566
- Bourkhouk, O., Bachari, E., and Adnani, M. (2016). A personalized e-learning based on recommender system. *Int. J. Learn.* 2, 99–103. doi: 10.18178/ijlt.2.2.99-103
- Brik, M., and Touahria, M. (2020). Contextual information retrieval within recommender system: case study 'e-learning system. *TEM J.* 9, 1150–1162. doi: 10.18421/TEM93-41
- Cakula, S., and Sedleniece, M. (2013). Development of a personalized e-learning model using methods of ontology. *Procedia Comput. Sci.* 26, 113–120. doi: 10.1016/j.procs.2013.12.011
- Chebbi, C., and Imen, C. (2021). "Ontological model for personalized the inclusive learning," in *E-Learning Framework and Assessment*. doi: 10.13140/RG.2.2.21218.04800/1
- Chee, S. H. S., Han, J., and Wang, K. (2001). RecTree: an efficient collaborative filtering method. *Lecture Notes Comp. Sci.* 2114, 141–151. doi: 10.1007/3-540-44801-2_15
- Ciloglugil, B., and Inceoglu, M. M. (2018). A learner ontology based on learning style models for adaptive e-learning. *Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics)*. LNCS 10961, 199–212. doi: 10.1007/978-3-319-95165-2_14
- Colombo-Mendoza, L. O., Valencia-García, R., Rodríguez-González, A., Alor-Hernández, G., and Samper-Zapater, J. J. (2015). RecomMetz: a context-aware knowledge-based mobile recommender system for movie showtimes. *Expert Syst Appl.* 42, 1202–1222. doi: 10.1016/j.eswa.2014.09.016
- Dwivedi, P., Kant, V., and Bharadwaj, K. K. (2017). Learning path recommendation based on modified variable length genetic algorithm. *Educ. Inf. Technol.* 23, 819–836. doi: 10.1007/s10639-017-9637-7
- Ekstrand, M. D., Riedl, J. T., and Konstan, J. A. (2010). *Collaborative Filtering Recommender Systems*. Boston, MA.
- Fleming, N. D., and Mills, C. (1992). Not another inventory, rather a catalyst for reflection. *Improve Acad.* 11, 137–155. doi: 10.1002/j.2334-4822.1992.tb02013.x
- Fournier-Viger, P., Chun, J., Lin, W., Kiran, R. U., Koh, Y. S., and Thomas, R. (2017). A survey of sequential pattern mining. *Data Sci. Pattern Recogn.* 1:1.
- Fraihat, S., and Shambour, Q. (2015). A framework of semantic recommender system for e-learning. *J. Softw.* 10, 317–330. doi: 10.17706/jsw.10.3.317-330
- George, G., and Lal, A. M. (2019). Review of ontology-based recommender systems in e-learning. *Comput. Educ.* 142:103642. doi: 10.1016/j.compedu.2019.103642
- Ghauth, K. I., and Abdullah, N. A. (2010). Learning materials recommendation using good learners' ratings and content-based filtering. *Educ. Technol. Res. Dev.* 58, 711–727. doi: 10.1007/s11423-010-9155-4
- Halim, A., Mahzum, E., Yacob, M., Irwandi, I., and Halim, L. (2021). The impact of narrative feedback, e-learning modules and realistic video and the reduction of misconception. *Educ. Sci.* 11:158. doi: 10.3390/educsci11040158
- Han, J., Cheng, H., Xin, D., and Yan, X. (2007). Frequent pattern mining: current status and future directions. *Data Mining Knowl. Discov.* 15, 55–86. doi: 10.1007/s10618-006-0059-1
- Han, J., Kamber, M., and Pei, J. (2012). "Getting to know your data," in *Data Mining* (Berlin; Heidelberg: Springer), 39–82.
- Hanurawan, N. (2017). Teaching writing by using Visual, Auditory, Read/Write, And Kinesthetic (VARK) learning style in descriptive text to the seventh grade students of SMPN 2 Jiwan. *English Teach. J.* 5:1. doi: 10.25273/etj.v5i1.4721
- Harrathi, M., Touzani, N., and Braham, R. (2017). "A hybrid knowledge-based approach for recommending massive learning activities," in *IEEE/ACS 14th International Conference on Computer Systems and Applications (AICCSA)* (Hammamet: IEEE), 49–54.
- Hassan, M. A., Habiba, U., Majeed, F., and Shoaib, M. (2021). Adaptive gamification in e-learning based on students' learning styles. *Interact. Learn. Environm.* 29, 545–565. doi: 10.1080/10494820.2019.1588745
- Hawk, T. F., and Shah, A. J. (2007). Using learning style instruments to enhance student learning. *Decis. Sci. J. Innovat. Educ.* 5, 1–19. doi: 10.1111/j.1540-4609.2007.00125.x
- Imran, H., Hoang, Q., Chang, T.-W., and Graf, S. (2014). A framework to provide personalization in learning management systems through a recommender system approach. *LNAI 8397*, 271–280. doi: 10.1007/978-3-319-05476-6_28

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

- Jeevamol, J., and Renumol, V. G. (2021). An ontology-based hybrid e-learning content recommender system for alleviating the cold-start problem. *Educ. Inform. Technol.* 26, 4993–5022. doi: 10.1007/s10639-021-10508-0
- Joy, J., Raj, N. S., and Renumol, V. G. (2021). Ontology-based E-learning content recommender system for addressing the pure cold-start problem. *ACM J Data Inf Qual.* 13:3. doi: 10.1145/3429251
- Joy, J., and Renumol, V. G. (2020). "Comparison of generic similarity measures in E-learning content recommender system in cold-start condition," in *2020 IEEE Bombay Section Signature Conference, IBSSC* (Bombay: IEEE), 175–179.
- Karen, S., and Felicitati, L. (1992). Learning styles of marketing majors. *Educ. Res. Quart.* 15, 15–23.
- Keefe, J. (1979). "An overview," in *NASSP's Student Learning Styles: Diagnosing and Prescribing Programs* (Reston, VA: National Association of Secondary School Principals).
- Kulkarni, P. V., Rai, S., and Kale, R. (2020). "Recommender system in eLearning: a survey," in *Proceeding of International Conference on Computational Science and Applications* (Singapore: Springer), 119–126.
- Lalitha, T. B., and Sreeja, P. S. (2020). "Personalised self-directed learning recommendation system," *Procedia Comp. Sci.* 171, 583–592. doi: 10.1016/j.procs.2020.04.063
- Lam, X. N., Vu, T., Le, T. D., and Duong, A. D. (2008). "Addressing cold-start problem in recommendation systems," in *Proceedings of the 2nd International Conference on Ubiquitous Information Management and Communication - ICUIMC '08*. New York: ACM Press, 208–211.
- Li, X., and Chang, S.-K. (2005). "A personalized e-learning system based on user profile constructed using information fusion," in *Proceedings International Conference on Database and Expert Systems Applications (DMS)* (Banff, AB), 109–114
- Liu, X. (2017). A collaborative filtering recommendation algorithm based on the influence sets of e-learning group's behavior. *Cluster Comput.* 22, 2823–2833. doi: 10.1007/s10586-017-1560-6
- Lu, J. (2014). "Personalized e-Learning Material Recommender System," in *Proceedings International Conference on Information Technology and Applications*.
- Lü, L., Medo, M., Yeung, C. H., Zhang, Y. C., Zhang, Z. K., and Zhou, T. (2012). Recommender systems. *Phys Rep.* 519, 1–49. doi: 10.1016/j.physrep.2012.02.006
- Mladenic, D. (1999). Text-learning and related intelligent agents: a survey. *IEEE Intellig. Syst. Their Appl.* 14, 44–54. doi: 10.1109/5254.784084
- Movchun, V., Lushkov, R., and Pronkin, N. (2020). Prediction of individual learning style in e-learning systems: opportunities and limitations in dental education. *Educ. Inf. Technol.* 26, 2523–2537. doi: 10.1007/s10639-020-10372-4
- Niknam, M., and Thulasiraman, P. (2020). LPR: A bio-inspired intelligent learning path recommendation system based on meaningful learning theory. *Educ. Inf. Technol.* 25, 3797–3819. doi: 10.1007/s10639-020-10133-3
- Obeid, C., Lahoud, I., Khoury, H., and Champin, P. A. (2018). "Ontology-based recommender system in higher education," in *The Web Conference 2018 - Companion of the World Wide Web Conference, WWW*, 1031–1034. doi: 10.1145/3184558.3191533
- Ompusunggu, L. D., Senses, D. I., Wahbi, A., and Mahdalina, R. (2021). "Comparison between rule-based expert support system and machine learning expert support system in KM," in *2021 2nd International Conference on Smart Computing and Electronic Enterprise: Ubiquitous, Adaptive, and Sustainable Computing Solutions for New Normal, ICSCEE*, 114–120.
- Rahayu, N. W., Ferdiana, R., and Kusumawardani, S. S. (2022). A systematic review of ontology use in E-Learning recommender system. *Comp. Educ.* 3:100047. doi: 10.1016/j.caeai.2022.100047
- Rahman, M. M., and Abdullah, N. A. (2018). A personalized group-based recommendation approach for web search in E-learning. *IEEE Access.* 6, 34166–34178. doi: 10.1109/ACCESS.2018.2850376
- Ricci, F., Rokach, L., and Shapira, B. (2011). "Introduction to recommender systems handbook," in *Recommender Systems Handbook* (Boston, MA: Springer), 1–35.
- Salehi, M., and Kmalabadi, I. N. (2012). A hybrid attribute-based recommender system for e-learning material recommendation. *IERI Procedia.* 2, 565–570. doi: 10.1016/j.ieri.2012.06.135
- Saveski, M., and Mantrach, A. (2014). "Item cold-start recommendations," in *Proceedings of the 8th ACM Conference on Recommender systems - RecSys '14*. New York: ACM Press, 89–96.
- Shanshan, S., Mingjin, G., and Lijuan, L. (2021). An improved hybrid ontology-based approach for online learning resource recommendations. *Educ. Technol. Res. Dev.* 69, 2637–2661. doi: 10.1007/s11423-021-10029-0
- Shishchchi, S., Banihashem, Y., Azan, N., Zin, M., Azman, S., and Noah, M. (2012). Ontological approach in knowledge based recommender system to develop the quality of E-learning system. *Aust J. Basic Appl. Sci.* 6, 115–123.
- Sikka, R., Dhankhar, A., and Rana, C. (2012). A survey paper on e-learning recommender system. *Int J Comput Appl.* 47, 975–888. doi: 10.5120/7218-0024
- Silva, N., Carvalho, D., Pereira, A. C. M., Mourão, F., and Rocha, L. (2019). The Pure Cold-Start Problem: A deep study about how to conquer first-time users in recommendations domains. *Inf. Syst.* 80, 1–12. doi: 10.1016/j.is.2018.09.001
- Singh, J., Ram, H., and Sodhi, J. S. (2013). Improving efficiency of apriori algorithm using transaction reduction. *Int. J. Scient. Res. Publicat.* 3:1.
- Sobhanam, H., and Mariappan, A. K. (2013). "Addressing cold start problem in recommender systems using association rules and clustering technique," in *International Conference on Computer Communication and Informatics, ICCCI* (Coimbatore: IEEE).
- Tarus, J., Niu, Z., and Khadija, B. (2017). E-learning recommender system based on collaborative filtering and ontology. *Int. J. Comp. Inform. Eng.* 11, 256–262. doi: 10.5281/zenodo.1129067
- Tarus, J. K., Niu, Z., and Kalui, D. (2017a). A hybrid recommender system for e-learning based on context awareness and sequential pattern mining. *Soft Comp.* 22, 2449–2461. doi: 10.1007/s00500-017-2720-6
- Tarus, J. K., Niu, Z., and Yousif, A. (2017b). A hybrid knowledge-based recommender system for e-learning based on ontology and sequential pattern mining. *Future Generat. Comp. Syst.* 72, 37–48. doi: 10.1016/j.future.2017.02.049
- Willmott, C. J., and Matsuura, K. (2005). Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. *Clim Res.* 30, 79–82. doi: 10.3354/cr030079
- Wu, J., and Feng, Q. (2020). Recommendation system design for college network education based on deep learning and fuzzy uncertainty. *J. Intellig. Fuzzy Syst.* 38, 7083–7094. doi: 10.3233/JIFS-179787
- Yang, Q., Sun, J., Wang, J., and Jin, Z. (2010). "Semantic web-based personalized recommendation system of courses knowledge research," in *Proceedings 2010 International Conference on Intelligent Computing and Cognitive Informatics (ICICCI)* (Kuala Lumpur: IEEE), 214–217.
- Zhang, Q., Lu, J., and Zhang, G. (2021). Recommender systems in E-learning. *J. Smart Environm. Green Comp.* 1, 76–89. doi: 10.20517/jsecg.2020.06
- Zhao, X., Niu, Z., Chen, W., Shi, C., Niu, K., and Liu, D. (2014). A hybrid approach of topic model and matrix factorization based on two-step recommendation framework. *J. Intellig. Informat. Syst.* 44, 335–353. doi: 10.1007/s10844-014-0334-3