

Received July 2, 2019, accepted August 3, 2019, date of publication August 14, 2019, date of current version November 20, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2935417

On Recommendation of Learning Objects Using Felder-Silverman Learning Style Model

SHAIMAA M. NAFAE^{1,2}, FRANÇOIS SIEWE², AND YING HE²

¹School of Management Information System, Arab Academy for Science, Technology & Maritime Transport, Alexandria 13055, Egypt

²School of Computer Science and Informatics, De Montfort University, Leicester LE1 9BH, U.K.

Corresponding authors: Shaimaa M. Nafea (P15017421@myemail.dmu.ac.uk), François Siewe (fsiewe@dmu.ac.uk), and Ying He (ying.he@dmu.ac.uk)

This work was supported in part by the Arab Academy for Science Technology and Maritime in Egypt through the Scholarship Program, and in part by the De Montfort University in the U.K.

ABSTRACT The e-learning recommender system in learning institutions is increasingly becoming the preferred mode of delivery, as it enables learning anytime, anywhere. However, delivering personalised course learning objects based on student's preferences is still a challenge. Current mainstream recommendation algorithms, such as the Collaborative Filtering (CF) and Content-Based Filtering (CBF), deal with only two types of entities, namely users and items with their ratings. However, these methods do not pay attention to student's preferences, such as learning styles, which are especially important for the accuracy of course learning objects prediction or recommendation. Moreover, several recommendation techniques experience cold-start and rating sparsity problems. To address the challenge of improving the quality of recommender systems, in this paper a novel recommender algorithm for machine learning is proposed, which combines students actual rating with their learning styles to recommend personalised course learning objects (LOs). Various recommendation techniques are considered in an experimental study investigating the best technique to use in predicting student ratings for e-learning recommender systems. We use the Felder-Silverman Learning Styles Model (FSLSM) to represent both the student learning styles and the learning object profiles. The predicted ratings are compared with the actual student ratings to determine the accuracy of the recommendation techniques, using the Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) metrics. This approach has been experimented on 80 students for an online course created in the MOODLE Learning Management System. The results of the experiment show that the best recommendation technique is our proposed hybrid recommendation algorithm that combines the collaborative filtering and the content-based filtering techniques to enhance the accuracy of the predictions, and solves the cold-start and the rating sparsity problems using the FSLSM representations of the student learning styles and the learning object profiles.

INDEX TERMS Collaborative filtering (CF), content-based filtering (CBF), e-learning, Felder-Silverman learning style model (FSLSM), hybrid filtering (HF), rating prediction, recommendation system.

I. INTRODUCTION

E-Learning Recommender Systems (E-LRS) have become popular in recent years. Compared with Learning Management Systems (LMS), which offer limited adaptivity and personalisation, adaptive educational systems use intelligent algorithms to adapt to students' learning style, enhance learning performance, accelerate goal achievement, reduce

The associate editor coordinating the review of this manuscript and approving it for publication was Fatos Xhafa.

navigational overheads, and to enhance overall student satisfaction [1]. Besides, the adaptability and diversity of recommendations are desirable in e-learning recommender systems, because learners' preferences and abilities keep changing. The diverse and adaptive Learning Objects (LOs) recommendations should be presented to motivate the learning potential of learners and to ensure a long-term learning experience [2]–[4].

A lot of learners are moving away from utilising e-learning systems, because they do not find them beneficial [24]–[27].

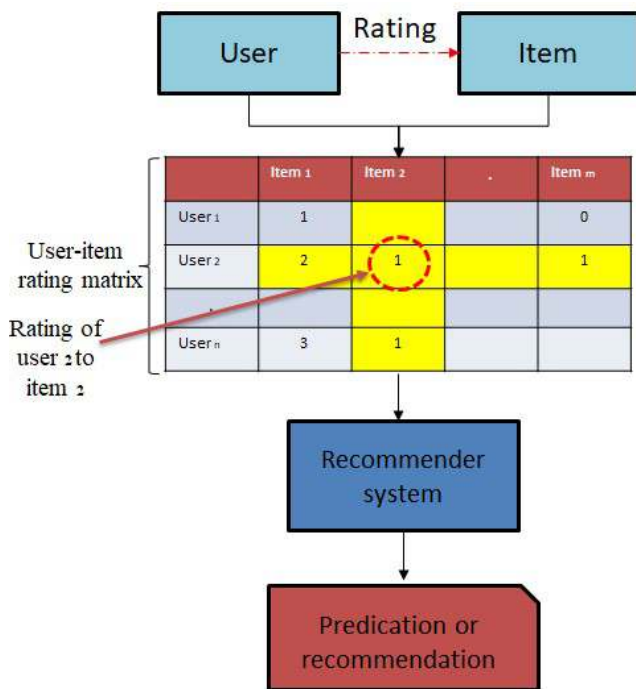


FIGURE 1. A Traditional recommendation approach.

In particular, this is owing to the fact that this type of learning environment cannot interact with learners as well as the traditional face to face one. Moreover, learners usually make a set of choices during learning, for instance, ‘how to learn’, ‘with whom to learn’ and ‘which learning pathway to follow’, and so on. To achieve this goal, it is essential to consider the students’ learning styles and use them in the design and implementation of e-learning environments, to make them more realistic and thus, attractive [28].

This paper proposes a novel algorithm to recommend the most suitable course LOs taking into consideration student learning styles and LO profiles and the students’ ratings of LOs. This is an extended version of work originally presented in [64]. A recommender system (RS) enables users to cope with information overload by providing the most appropriate items based on their requirements. Figure 1 shows the traditional method of a two-dimensional recommender system, which has three main components: user, item, and rating. Rating, in this case, refers to the feedback that a user gives for a specific item, being implicit or explicit.

- *Explicit ratings* are when the user rates an item to express his/her level of interest. Ratings can be in the form of a numeric value on a multi-point scale, e.g. 1 to 5 [11].
- *Implicit ratings* are generated by the RS itself, through inferences from users’ behaviour [11], [91].

A user-item matrix is shown in Fig. 1, where the elements in the matrix are the users’ ratings. In the matrix, the rows

depict the user list, while the columns represent the items list. The numerical values from 1 to 5 in the matrix reflect the level of preference for a particular user for each item. The objective of RS algorithm in this setting is to predict the missing values in the matrix where users have not provided their preferences for certain items. However, this RS always suffers from data sparsity and cold start. Data sparsity refers to the situation where the amount of information (ratings) of a target user is not sufficient enough to generate reliable related users (i.e. the number of commonly rated items among users is very small). Cold start refers to the situation where an RS encounters new users or items with no rating [89]. Over the years, researchers have developed mechanisms and tools for the automatic detection of types of learning style [39]–[41], [53], [58]–[60], [63]. However, few of the studies have discussed the mechanisms for generating an adaptive course content based on detected learning styles and the learning objects and materials already provided by teachers [61], [62], [64].

As a motivating example, let us assume two students with different learning styles have the same ratings of the learning objects. Clearly, from a fully personalised perspective, the top-n LOs list based on predicted ratings cannot be the same for both students, because they do not have the same learning style. Hence, it is important to consider students’ learning styles when predicting their ratings of LOs to improve the accuracy of the recommendations. Accordingly, in this study, a novel hybrid recommendation algorithm is proposed based on the personalised students profile presented in [63] and the K-means clustering as a way to overcome information overload and cold start problems, thus building an effective course learning objects recommendation system.

A. LITERATURE GAP AND RESEARCH CONTRIBUTIONS

Current published relevant research papers in the field of the e-learning recommendation systems, including the content-based filtering, collaborative filtering and hybrid recommendation techniques, are presented in Table 1. Existing e-learning recommendation systems face the following problems.

- 1) First, the majority of the traditional recommendation algorithms have been developed for e-commerce applications that are unable to meet all the requirements of learning environments. In particular, they do not consider the learning process in their recommendation approach [39], [92].
- 2) Second, the recommendation mechanisms that rely exclusively on two dimensions (i.e. users and items) don’t consider the attributes of learners and learning materials [7]. As a result, rich and vital information, such as learner’s learning styles and the properties of learning objects are overlooked.
- 3) Third, during the continuous learning process, learners do not actively make ratings or give comments,

TABLE 1. Summary of existing personalised e-learning recommendation systems.

Study	Description	Cold-Start	Issue	Used Recommendation Techniques		
				CF	CBF	HF
[32]	Predict the most suitable learning materials to each learner based on collaboration with other learners.	✓	No past preferences	✓		
[36]	Recommend the most suitable learning materials for each learner based on the rating similarity with other learners.	✓	No past preferences or No high rated LO in past preferences	✓		
[35]	Finding learning objects that would be suitable for learners preferences (knowledge level and learning style).	✓	No past preferences		✓	
[34]	Recommend course learning objects based on neighborhood rating.	✓	No high rated LO in past preferences	✓		
[33]	Constructed a course ontology and retrieved the course according to in learner’s learning styles.	✓	No past preferences	✓		✓(+ontology)
[81]	Proposed a hybrid recommender system for learning materials by combining CBF, CF and ontology.	✓	No past preferences or No high rated LO in past preferences	✓	✓	✓(+ontology)
[42]	Suggested a method of clustering learning objects to improve their recommendations.	✓	Zero rating to the LO	✓		✓(+ k-means)
[81]	Proposed a hybrid recommender system to recommend learning items in users learning processes	✓	No past preferences or No high rated LO in past preferences	✓		✓(+SPM algorithm)
[67]	Proposed a recommender system for storing and sharing research papers and glossary terms among university students and industry practitioners.	✓	No past preferences	✓		
[68]	Proposed the courseware management architecture with courseware recommendation that combines the user contents filtering and collaborative filtering.	✓	No past preferences	✓	✓	
[69]	proposed course recommendation system based on ontology and context aware e-learning.	✓	No high rated LO in past preferences		✓	
[3]	Recommend learning contents to users based on similarity between user profiles.	✓	No past preferences	✓		
[71]	Proposed framework for recommending learning materials based on the similarity of content items and good learners’ average rating strategy.	✓	No past preferences or No high rated LO in past preferences		✓	
[74]	proposed a framework for recommending learning resources based on the learner’s recent navigation history and by comparing similarities and differences among different learners’ preferences and instructional content available in the e-learning system.	✓	No past preferences or No high rated LO in past preferences	✓	✓	
[75]	Recommend learning Materials based on multidimensional attributes	✓	No high rated LO in past preferences	✓	✓	
[76]	Proposed approach for selecting and sequencing the most appropriate learning objects.	✓	No high rated LO in past preferences	✓		(+association pattern analysis)
[57]	proposed a hybrid approach for recommending learning objects (LOs) to learners	✓	No past preferences			✓(+intuitionistic fuzzy logic and SPM algorithm)

because they aim to achieve their goals within scheduled but limited learning time [95]. As a result, learners’ learning profiles often seem isolated from each other. The extreme data sparsity caused by these factors can render traditional recommendation techniques ineffective.

- 4) Finally, traditional recommendation methods have a low ability to capture and perceive the changes in learners’ preferences in an adaptive way [66].

In order to avoid these drawbacks and improve the accuracy of course learning objects recommendations, the key contributions of this paper are threefold:

- 1) In comparison with most of existing e-learning recommendation systems such as [32], [36], [78] which used only rating values, the proposed algorithm takes into account multidimensional-attribute (based on FLSLM) of learning objects profiles and students learning styles in addition to rating values in its recommendation process. Thus, compared to these methods, the proposed method produces more accurate recommendations and is more effective in dealing with the cold-start and the the rating sparsity problems using the data from the student learning styles and the learning object profiles (Section III-A)).

- 2) A new approach is proposed to overcome the new-learner zero-rated profile recommendation issue by determining the nearest learners with a similar historical rating and similar learning styles profile (Section III-A.3).
- 3) Several recommendations algorithms have been tested in order to find out the best one for the course learning objects recommendation. The accuracy of the recommendations is measured using traditional evaluation metrics, namely the Mean Absolute Error (MAE) and the Root Mean Squared Error (RMSE). The results indicate that the hybrid recommender technique has higher accuracy in comparison with collaborative and content-based recommendation techniques (Section IV).

Our proposed algorithm has been implemented in C++ using Visual Studio and Windows Presentation Foundation (WPF) to design the Graphical User Interface (GUI). It has been evaluated using a real student dataset from AAST'S MOODLE (Arab Academy for Science and Technology and Maritime Transport - Modular Object-Oriented Dynamic Learning Environment).

The remainder of the paper is organised as follows. The next section defines the main concepts used in the proposed approach. Section III discusses the proposed recommender algorithm, whilst Section IV presents the experimental results and analysis. Section V concludes the paper and proposes future research directions.

II. PRELIMINARIES

In the following sub-sections, we present techniques commonly used in recommender systems. These will be analysed later to improve the accuracy of recommendations. An overview of the similarity metrics and K-means clustering algorithm is also given.

A. RECOMMENDATION TECHNIQUES

The underlying techniques used in recommender systems can be categorised into two broad classes: (a) content-based recommendation (b) collaborative filtering recommendation. New hybrid recommendation algorithms can be generated by synthesising these two methods [19].

1) CONTENT-BASED FILTERING (CBF)

As a traditional recommendation method, the rationale for CBF is simple. The items recommended by this method are similar to the items of users interest [16], with matching information between items and users being the key procedure. In e-learning recommender systems, the items are the learning objects in the e-learning systems and the users are the learners. CBF recommender systems work with profiles of learners that are created at the beginning. A profile has information about a learner and his/her preferences, which are based on how he/she rates the LOs. Generally, when creating

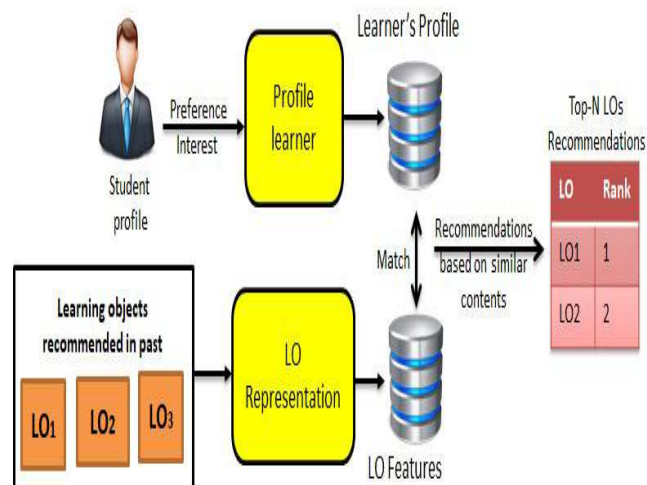


FIGURE 2. Content-based filtering recommendation.

a profile, recommender systems make a survey to get initial information about a user to avoid the new-user problem. In the recommendation process, the engine compares the LOs that have already been positively rated by learner with the LOs he/she has not done so and looks for similarities. Those LOs that are mostly similar to the positively rated ones, will be recommended to the user. In this case, the profiles of other users are not essential and they do not influence the recommendations of the user, for they are based on individual information. Figure 2 presents an illustrative example of CBF. From Fig. 2, we can see that the recommendation process comprises three main steps: item representation, profile learning and recommendation generation. As an example for e-learning application, [74] used learners' recent navigation histories and similarities and dissimilarities among the contents of the learning materials for online automatic recommendations. Clustering was proposed by [79] to group learning documents based on their topics and similarities. Since in the e-learning environment learning materials are in a variety of multimedia formats, including text, hypertext, image, video, audio and slides, it is difficult to calculate the content similarity of two items [81]. In fact, the existing metrics in CBF only detect similarity between LOs that share the same attributes. This causes overspecialised recommendations that only include LOs very similar to those that the learner already knows.

2) COLLABORATIVE FILTERING (CF)

Collaborative filtering became one of the most researched techniques of recommender systems after it was proposed and described by [54]. CF [15] recommends to the target learner learning resources that other similar learners have registered as liking previously. In other words, an important step in CF is to identify those learners most similar to the target learner. The similarity in taste of two learners is calculated based on their rating history. If two learners have the same or almost

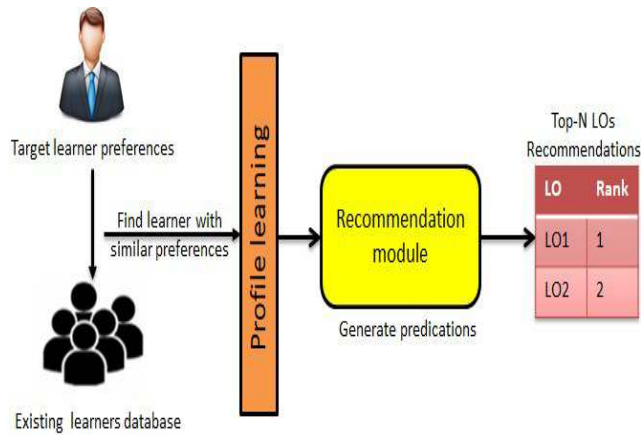


FIGURE 3. Collaborative filtering algorithm.

the same rated LOs in common, then they are deemed to have similar tastes. Such learners and others of the same ilk form a group or a so-called neighbourhood. A learner gets recommendations to choose LOs that he/she has not rated before, but have already been positively rated by those in his/her neighbourhood, as shown in Fig. 3.

To this end, several research efforts have been made to identify similarity measures so as to identify these users with common profiles [37], [77]. CF was used by [32] for prediction of the most suitable materials for the learner as follows. First, the weight between all users and the active learner is calculated by the Pearson correlation. Then, the n users that have the highest similarity to the active learner are selected as belonging to the neighbourhood. Finally, using the weight combination obtained from the neighbourhood, the rating prediction is calculated. Regardless of its success in many application domains, collaborative filtering has two serious drawbacks. First, its applicability and quality are limited by the so-called sparsity problem, which occurs when the available data are insufficient for identifying similar users. Second, it requires knowing many user profiles in order to elaborate accurate recommendations for a given user. Given in some e-learning environments the learner population is low, recommendation results with this method in such cases will have poor accuracy.

3) HYBRID FILTERING (HF)

In the last few years, researchers of recommender systems have explored hybridisation of recommendation techniques as an approach for developing effective recommender systems. Table 1 lists some of the techniques that have been used to this end. Hybrid filtering entails combining two or more recommendation techniques to improve performance, as shown in Fig. 4. In [68], a combination of content-based and collaborative filtering was implemented to make personalised recommendations for a courseware selection module. The algorithm starts with user u entering some keywords

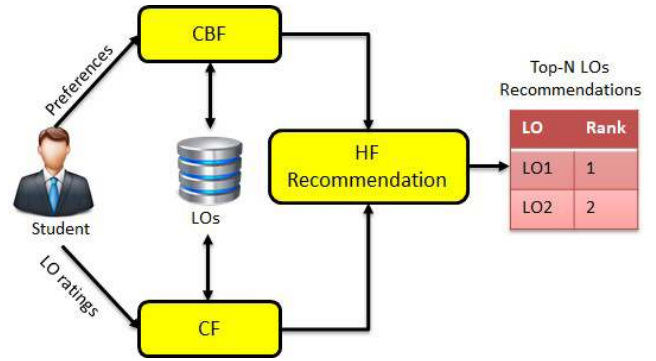


FIGURE 4. Hybrid filtering algorithm.

on the portal of the courseware management system. Then, the courseware recommendation module finds within the same user interest group of user u the k courseware with the same or similar keywords that others have chosen. Reference [21] applied association rule mining to identify interesting information through students usage data in the form of IF-THEN recommendation rules and then, used a collaborative recommender system to share and score the recommendation rules obtained by teachers with similar profiles as well as other experts in education.

B. SIMILARITY METRICS

Similarity metrics are the backbone of CF and CBF helping to predict the ratings of unrated items. Regarding which, in this study, the two most convenient similarity metrics, namely, Pearson’s correlation and cosine similarity [64] are considered.

1) PEARSON CORRELATION

The Pearson correlation coefficient is a measure of the linear dependence between two variables (real-valued vectors). Specifically, that of two variables x and y is formally defined as the covariance of the two variables divided by the product of their standard deviations (which acts as a normalisation factor) [72] and it can be equivalently defined by Eq. (1).

$$P(x, y) = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^n (y_i - \bar{y})^2}} \quad (1)$$

where \bar{x} and \bar{y} are the mean values of x and y , respectively.

The coefficient $P(x, y)$ ranges from -1 to 1 and is invariant to linear transformations of either variable. The value -1 represents perfect negative linear dependence, 0 no linear dependence, and 1 perfect positive linear dependence. Used as a similarity metric, negative values indicate dissimilarity, while positive values measure the similarity between the two variables with 1 be the perfect similarity.

2) COSINE SIMILARITY

The cosine similarity involves measuring the angle between two vectors [94] and is calculated by Eq. (2), as the ratio of

the scalar product by the product of the magnitudes.

$$c(x, y) = \frac{x \cdot y}{\|x\| \cdot \|y\|} \quad (2)$$

The values of $c(x, y)$ range from -1 to 1 in general, and from 0 to 1 if the coordinates of x and y are non-negative values. This paper is interested in the latter where the value 0 represents no similarity and 1 perfect similarity.

C. K-MEANS CLUSTERING ALGORITHM

Clustering is one of the most common data mining techniques used in recommendation systems in order to develop recommendation rules or build recommendation models from large data sets [73]. It can be defined as the process of organising objects in a database into clusters (or groups), such that objects within the same cluster have a high degree of similarity, while those belonging to different ones have a high degree of dissimilarity. The K-means algorithm [65] is one of the most popular clustering algorithms due to its simplicity and intuitive interpretation. The algorithm has the following steps.

Step 1: Select K random points from the dataset as initial cluster centroids.

Step 2: Create K clusters by associating each data point with its closest cluster centroid, according to the Euclidean distance defined by Eq. (3), where x and y are two real-valued vectors of size n .

$$D(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (3)$$

Step 3: Recalculate the centroid of each cluster as the mean of all the data points in that cluster.

Step 4: Repeat steps 2 and 3 until the centroids no longer change.

In our proposed system, we apply data clustering by employing the K-means algorithm to improve the computational efficiency, the accuracy and the quality of recommendations.

D. DESCRIPTION OF THE FELDER-SILVERMAN LEARNING STYLE MODEL

The term ‘learning style’ refers to the preferential way in which the student perceives, processes, understands and retains information [45]. Various learning style models have been presented in the past by researchers, such as those of Felder *et al.* [17], Honey and Mumford [90], Kolb [47], Mayer and Myers [56], Dunn [55] and Pask [48]. In our case, we use the Felder and Silverman model (FSLSM) [17] to represent both the student learning styles and the learning object profiles for the following reasons.

- First, it is the most widely used in educational systems thanks to its ability to quantify students learning styles, as shown in Table 2.
- Second, it is used very often in technology-enhanced learning and some researchers even argue that it is the

most appropriate learning style model for the use in adaptive learning systems such as [18]–[21], as well as being easy to implement [30], [57].

FSLSM describes learning styles by characterising each learner according to four dimensions, each of which, is defined as below.

The *information processing* dimension (active/reflective) tells how one prefers to process information. An active learner wants to try things out, working with others in groups, whilst a reflective one chooses to think things through, working alone or with a familiar partner.

The *information input* dimension (visual/verbal) determines how one prefers information to be presented. A visual learner likes visual presentations, pictures, diagrams, and flow charts. A verbal learner prefers written and spoken explanations.

The *information understanding* dimension (sequential/global) determines how one prefers to organise and progress towards understanding information. A sequential learner prefers linear thinking and learning in small incremental steps. By contrast, a global learner prefers holistic thinking, systems thinking, and learns in large leaps.

The *information perception* dimension (sensing/intuitive) states how one prefers to perceive or take in information. A sensing learner is attracted to concrete thinking, is practical as well as being concerned about facts and procedures. While an intuitive learner opts for conceptual thinking, being innovative, as well as being interested in theories and meanings.

It should be noted that each of these dimensions is characterised by a pair X/Y of learning style attributes (i.e. active/reflective, sequential/global, visual/verbal, and sensing/intuitive) meaning that the learning style of a learner in a particular dimension ranges from perfect X to perfect Y. For example, in the information processing dimension, the learning style of a student can be 70% *active* and 30% *reflective*. Of course, the percentage of X and the percentage of Y must sum up to 100%. Felder *et al.* [17] developed an Index of Learning Styles (ILS) questionnaire, which comprises 44 questions that have been shown to be effective in identifying the learning style of each individual learner. ILS provides a method of calculating the percentage values of learning style attributes from the learner’s answers to the questionnaire [17], [63]. The next section presents a novel algorithm for recommending learning objects based on student learning style.

III. PROPOSED ALGORITHM FOR RECOMMENDING PERSONALISED LEARNING OBJECTS

In this work, a new algorithm for rating prediction of the learning objects is proposed. The proposed algorithm predicts the ratings for a given learning object and a given student based on the student’s learning style, the learning object’s profile, and the ratings of learning objects. The FSLSM

TABLE 2. Summary of existing personalised e-learning systems considering learning styles.

System Name	LS Model	Adaptation technique
Protus 2.0 [84]	FSLSM	Learning style identification & personalised content recommendation
WELSA [49]	Unified LS Model	Presentation recommendation & sequencing
MAS-PLANG [50]	FSLSM	Content personalisation
UZWEBMAT [85]	VARK	Content recommendation
PLORS [51]	FSLSM	learning objects recommendation
Tortorella and Graf [83]	FSLSM	Video, audio, presentation score calculation in mobile environment
Christudas [86]	FSLSM	Compatibility and complexity of learning objects
e-Teacher [87]	FSLSM	Course personalisation
OSCAR CITS [88]	FSLSM	Course personalisation (SQL tutorial)
TANGOW [52]	FSLSM	Adapt course structure & sequencing
iWeaver [71]	Dunn & Dunn Model	Media recommendation (Flash animations or streaming audio)
learnFit [53]	MBTI	Course personalisation (PHP material)
CS383 [18]	FSLSM	Course personalisation
Kurilovas [82]	Honey & Mumford learning style	Teaching/learning strategies

TABLE 3. Examples of student learning style vectors.

	act	ref	vis	ver	seq	glo	sen	int
Fatima	0.7	0.3	0.2	0.8	0.5	0.5	0.6	0.4
Tom	0.4	0.6	0.1	0.9	0.7	0.3	0.8	0.2
Clara	0.5	0.5	0.6	0.4	0.8	0.2	0.7	0.3

learning style model described above is adopted to represent both the student learning styles and the learning object profiles. The rating is given on a scale of 1-5. First, work is carried out to design an effective algorithm for recommending top-n personalised learning objects in e-Learning systems based on student learning styles, as presented in (Sect. III-A). Then an experimental study is undertaken to find out which algorithm produces the best accuracy for rating prediction. The best performing algorithm is then retained for the recommender system. To present this study clearly, we consider two definitions.

Definition 1 (Student Profile): It is assumed that the student learning style is represented by a vector of real values ranging from 0 to 1 (or from 0% to 100%) as in Eq. (4), where the prefixes of learning style attributes are used as place holders.

$$LS = (act, ref, vis, ver, seq, glo, sen, int) \quad (4)$$

Some examples of student learning style vectors are given in Table 3 and these can be calculated using the student’s responses to the ILS questionnaire [63] or according to his/her learning behaviour [64].

Definition 2 (Object Profile): The learning content materials are structured into learning objects for each topic. Learning objects are provided in various formats and media in order to meet the learning styles of individual learners. They can be text documents (e.g. pdf), presentations (e.g. powerpoint slides), images, audios, videos, simulations, etc. For example, a visual learner will prefer to watch a video than to read a pdf document, while a verbal one will choose to do opposite. Hence, a learning object profile (OP) can be

TABLE 4. Examples of learning object profile.

	act	ref	vis	ver	seq	glo	sen	int
OP ₁	0.7	0.3	0.5	0.5	1	0	0.3	0.7
OP ₂	0.2	0.8	0.8	0.2	1	0	0.4	0.6
OP ₃	0.5	0.5	0.2	0.8	0.1	0.9	0.6	0.4
OP ₄	0.5	0.5	0.5	0.5	0.5	0.5	0.3	0.7
OP ₅	0.9	0.1	0.7	0.3	0.3	0.7	0.1	0.9

represented by a FLSLM learning style vector indicating the category of learners that this learning object is suitable for, as in Eq. (5).

$$OP = (act, ref, vis, ver, seq, glo, sen, int) \quad (5)$$

Unlike the student learning styles that are calculated through the ILS questionnaire or behaviour, it is assumed that the learning object profile is set by the teacher or an education professional. Some examples of learning object profiles are given in Table 4 for illustration.

In the following subsections, we give a description of the proposed predicated rating algorithms in detail and analyse the accuracy of the recommendations.

A. RATING PREDICTION ALGORITHMS

We propose three different algorithms for predicting the ratings of learning objects, based on the three approaches: collaborative filtering, content-based filtering and hybrid filtering. Unlike traditional approaches, each of these algorithms handles the cold-start and the rating sparsity problems effectively using information from the students learning styles and the learning objects profiles.

1) PREDICTING RATINGS BASED ON COLLABORATIVE FILTERING

From the description of the previous section, we notice that the traditional CF methods heavily rely on the co-rated items. However, the similarity computation cannot be performed when there are no rated items, which is called cold start

problem (see Sect. I). To improve the accuracy and quality of recommendation, our research CF is implemented as follows:

Let LS be the learning style vector of the active student.

- 1) Apply K-means to cluster the students learning styles.
- 2) Select cs the nearest cluster to LS as in Eq. (2)
- 3) Foreach LO x
 - a) Let I = set of the top-n nearest elements to LS in cs **that have rated** x , as in Eq. (1)
 - b) If $\|I\| > 0$ then calculate the predicted rating for x as in Eq. (6)
 - c) If $\|I\| = 0$ then calculate the predicted rating for x as in Eq. (7)
- 4) Recommend the top-n highly rated LOs.

$$\tilde{r}_1(LS, x) = \frac{\sum_{u \in I} P(LS, u) \times r(u, x)}{\sum_{u \in I} P(LS, u)} \quad (6)$$

$$\tilde{r}_2(LS, x) = \text{int}(0.5 + P(LS, x) \times 5) \quad (7)$$

where $\tilde{r}(LS, x)$ denotes the predicted rating value of the LO x for the active student LS . $P(LS, u)$ denotes the Pearson correlation coefficient (defined as in Eq. (1)) of the two vectors LS and u ; and $r(u, x)$ represents the actual rating of the LO x by the student u .

Equation (7) is used to solution the cold-start and the rating sparsity problems (case $\|I\| = 0$). In this case the predicted rating is measured as the similarity between the LO x and the active student LS multiplied by the maximum rating value which is 5. The value 0.5 is added so that the result is and integer between 1 and 5.

2) PREDICTING RATINGS BASED ON CONTENT-BASED FILTERING

The general principle of content-based approaches is to identify the common characteristics of learning objects that have received a favourable rating from a learner, and then recommend to him/her new learning objects that share these characteristics. In this work, we proposed an algorithm to enhance the accuracy of recommendations as follows:

Let LS be the learning style vector of the active student.

- 1) Let \mathcal{O} be the set of all learning objects **rated** by LS .
- 2) If $\mathcal{O} \neq \emptyset$ then
 - a) Apply K-means to cluster \mathcal{O}
 - b) Foreach LO x
 - i) Let co_x = the nearest cluster to x as in Eq. (2)
 - ii) Let J = set of the top-n nearest elements to x in co_x as in Eq. (1)
 - iii) Calculate the predicted rating for x as in Eq. (8)
 - c) Recommend the top-n highly rated LOs.
- 3) If $\mathcal{O} = \emptyset$ then
 - a) Apply K-means to cluster all the learning learning objects

- b) Let co = the nearest cluster to LS
- c) Foreach $x \in co$
 - i) Calculate the predicted rating for x as in Eq. (7)
- d) Recommend the top-n highly rated LOs in co .

$$\tilde{r}_3(LS, x) = \frac{\sum_{u \in J} P(x, u) \times r(LS, u)}{\sum_{u \in J} P(x, u)} \quad (8)$$

3) PREDICTING RATINGS BASED ON HYBRID FILTERING

We build a hybrid recommendation algorithm based the CF and CBF algorithms presented above as follows:

Let LS be the learning style vector of the active student.

- 1) Let α be the weight of CF in the hybrid model; $0 \leq \alpha \leq 1$.
- 2) Apply K-means to cluster the students learning styles
- 3) Select cs the nearest cluster to LS
- 4) Let \mathcal{O} be the set of all learning objects **rated** by LS .
- 5) Apply K-means to cluster \mathcal{O}
- 6) Foreach LO x
 - a) Let I = set of the top-n nearest elements to LS in cs **that have rated** x
 - b) Let co_x = the nearest LO cluster to x
 - c) Let J = set of the top-n nearest elements to x in co_x
 - d) If $\|I\| > 0$ and $\|J\| > 0$ then calculate the predicted rating for x as in Eq. (9)
 - e) If $\|I\| = 0$ and $\|J\| > 0$ then calculate the predicted rating for x as in Eq. (8)
 - f) If $\|I\| > 0$ and $\|J\| = 0$ then calculate the predicted rating for x as in Eq. (6)
 - g) If $\|I\| = 0$ and $\|J\| = 0$ then calculate the predicted rating for x as in Eq. (7)
- 7) Recommend the top-n highly rated LOs.

$$\tilde{r}(LS, x) = \alpha \times \tilde{r}_1(LS, x) + (1 - \alpha) \times \tilde{r}_3(LS, x) \quad (9)$$

Note that in Eq. (9), the value of α is between 0 and 1; and $\tilde{r}_1(LS, x)$ and $\tilde{r}_3(LS, x)$ are defined as in Eq. (6) and Eq. (8), respectively. Here are some examples:

- $\tilde{r}(LS, x) = 0.5 \times \tilde{r}_1(LS, x) + (1 - 0.5) \times \tilde{r}_3(LS, x)$
 $= 0.5 \times \tilde{r}_1(LS, x) + 0.5 \times \tilde{r}_3(LS, x)$
 $= \frac{\tilde{r}_1(LS, x) + \tilde{r}_3(LS, x)}{2}$
- $\tilde{r}(LS, x) = 0.2 \times \tilde{r}_1(LS, x) + (1 - 0.2) \times \tilde{r}_3(LS, x)$
 $= 0.2 \times \tilde{r}_1(LS, x) + 0.8 \times \tilde{r}_3(LS, x)$
- $\tilde{r}(LS, x) = 0.8 \times \tilde{r}_1(LS, x) + (1 - 0.8) \times \tilde{r}_3(LS, x)$
 $= 0.8 \times \tilde{r}_1(LS, x) + 0.2 \times \tilde{r}_3(LS, x)$
- $\tilde{r}(LS, x) = 0.75 \times \tilde{r}_1(LS, x) + (1 - 0.75) \times \tilde{r}_3(LS, x)$
 $= 0.75 \times \tilde{r}_1(LS, x) + 0.25 \times \tilde{r}_3(LS, x)$

“What is the recommendation algorithm that provides the best prediction of student ratings of learning objects?” An experimental study is carried out in order to find the answer to this question.

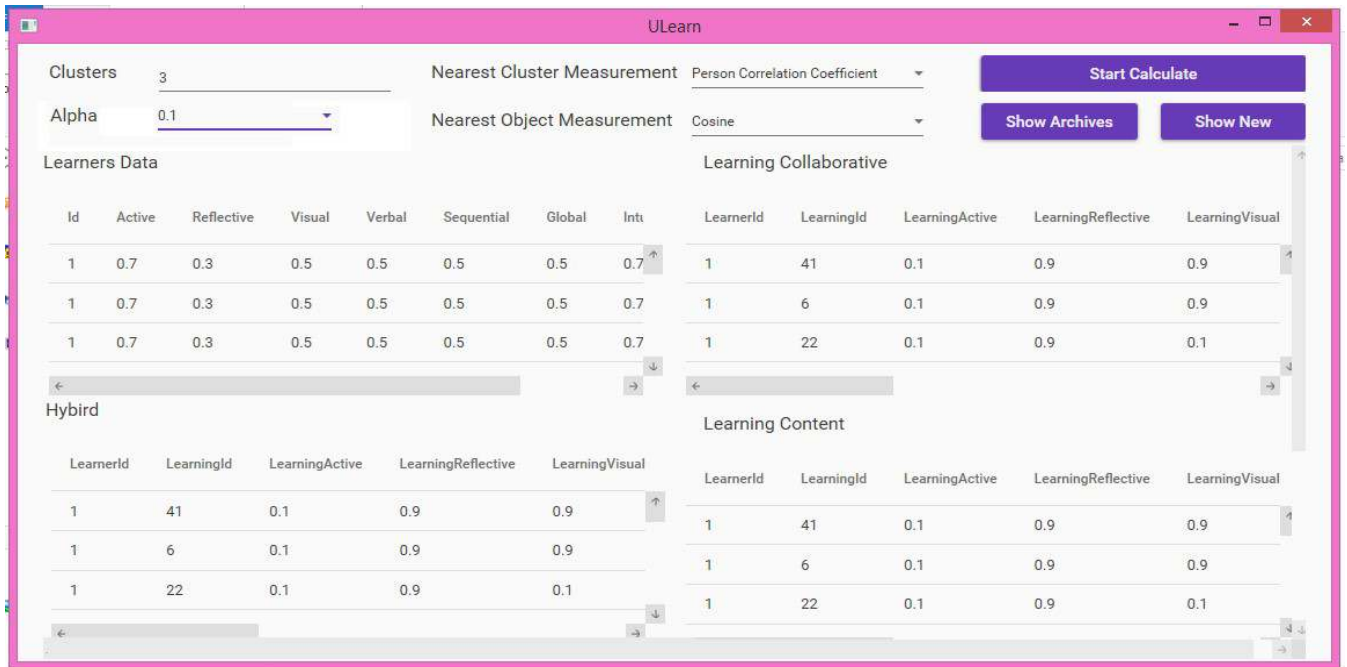


FIGURE 5. ULearn interface.

IV. EXPERIMENTAL STUDY

An experimental study was carried out to determine the most effective recommendation techniques to be used for the recommendation of LOs in e-learning systems. In this section, we describe the dataset, performance measurement, and the results of the proposed approach.

A. DATASET

The dataset of the MOODLE log-file at AAST is used in this study for the autumn and spring semesters in the academic years 2016-2017 and 2017-2018 in the school of business. MOODLE (Modular Object-Oriented Dynamic Learning Environment) is defined as a course management system (CMS), being a free and open source software package designed using pedagogical principles, to help educators by creating effective online learning communities. The course of interest is on “networks and e-commerce” and comprises 20 topics, with each topic having multiple leaning objects in various presentation styles. There was a minimum of 15 learning objects for each topic. The experimental set up consisted of 80 students whose learning styles were identified using the ILS questionnaire, as explained in [63]. During the course, the students were asked to rate each learning object using a 5-level Likert scale, with 1 being “not at all useful” and 5 be “very useful” to their learning. In order to evaluate the different aspects of the proposed method, student dataset was split into different parts, including:

- 1) Cold-start students: a set of students with lower than 5 ratings;
- 2) Cold-start learning objects: a set of new LOs, not rated by any students;
- 3) All students.

Cold-start was utilised to assess the ability of the algorithms to predict the ratings for those students with fewer previous rating of LOs, so little information was available for these students. The goal was to investigate how additional sources of information, such as students learning styles and learning objects profiles, can be used along with rating information to improve the accuracy of rating prediction.

B. PERFORMANCE MEASUREMENT

In this analysis, accuracy metrics are considered to assess the performance of the proposed recommender system algorithm. We use statistical accuracy metrics to evaluate the accuracy of the rating prediction algorithm.

The frequently used statistical metrics are Mean Absolute Error (MAE) and Root-Mean-Square Error (RMSE) defined as in Eq. (10) and Eq. (11), respectively, where r_i denotes the actual student rating of the learning object i and \tilde{r}_i is the predicted student’s rating for that learning object, $1 \leq i \leq n$.

In the computation of MAE, the first sum of absolute value of the difference between the actual and predicted rating is calculated, and then, it is divided by the total number of learning objects involved. Hence, a smaller value of MAE indicates a better prediction accuracy, as in Eq. (10).

$$MAE = \frac{1}{n} \sum_{i=1}^n |r_i - \tilde{r}_i| \tag{10}$$

The Root Mean Square Error (RMSE) is calculated by dividing the sum of squares of the differences of the actual and predicted ratings by the total number of learning objects on

TABLE 5. Showing predicted rating using the proposed algorithm.

S.ID	LO.ID	Act. R	Predicted rating using											
			CBF	CF	HF-0.1	HF-0.2	HF-0.3	HF-0.4	HF-0.5	HF-0.6	HF-0.7	HF-0.8	HF-0.9	
1	1	2	1	2	1	2	2	2	2	2	3	3	2	1
2	5	4	3	3	4	4	4	4	4	4	3	4	4	4
3	13	2	2	3	2	3	3	3	3	2	2	3	3	3
4	25	5	2	3	2	5	5	5	5	5	4	5	5	3
5	10	2	3	3	3	2	1	3	2	3	2	2	2	3
6	1	3	1	2	2	2	2	3	3	3	2	3	3	2
7	5	2	1	2	1	2	2	2	2	2	2	2	2	2
8	13	4	3	4	4	4	4	4	4	4	4	4	4	2
9	15	4	2	2	2	3	3	4	4	4	4	4	3	2
10	20	5	3	4	3	4	3	4	5	4	4	4	4	3
11	22	2	1	2	2	2	2	2	2	2	2	2	2	3
12	4	3	1	2	3	3	3	3	3	3	3	3	3	3
13	8	5	3	4	3	4	4	4	5	4	4	4	3	3
14	7	4	3	3	3	3	4	4	4	4	4	4	3	3
15	16	5	2	4	2	3	5	5	4	5	4	4	4	3

Act. R= Actual Rating CBF= Content Based Filtering CF=Collaborative Filtering HF=Hybrid filtering
 S.ID= Student ID LO.ID= Learning Object ID

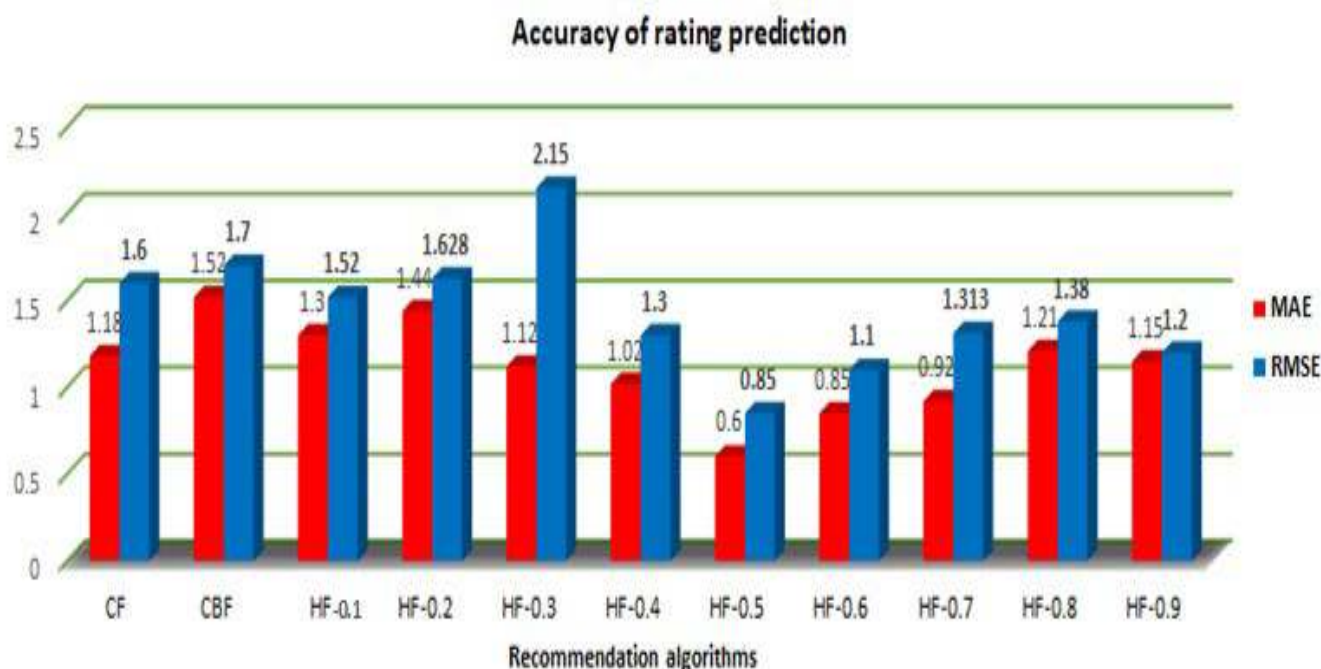


FIGURE 6. Accuracy of the recommender algorithm using MAE and RMSE.

which the predictions are made. The RMSE is obtained by taking the square root of the MSE, as in Eq. (11).

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (r_i - \tilde{r}_i)^2} \tag{11}$$

C. EXPERIMENTS AND EVALUATION

In this work, a new approach to ratings prediction of the course learning objects is proposed. The prototype was implemented in C++ using Visual Studio and Windows Presentation Foundation (WPF) to design the GUI (graphical user interface), with the SQL server being utilised to access the dataset and learners’ ratings. A set of experiments was

conducted on a Windows based PC with an Intel core i5 processor having a speed of 2.40 GHz and 16 GB of RAM. The GUI, which allows for selecting various combinations of similarity metrics, is depicted in Fig. 5. The first experiment focused on the accuracy of rating prediction, whereas the second one focused on the cold-start. Finally, the last part of the evaluation is about the integration of recommendation algorithms into AAST-MOODLE for testing them on real students. Each experiment will be detailed and discussed in the following subsections.

1) EVALUATION ON RATING PREDICTION

After performing the preprocess on student dataset, 15 students were selected and 15 learning objects were chosen

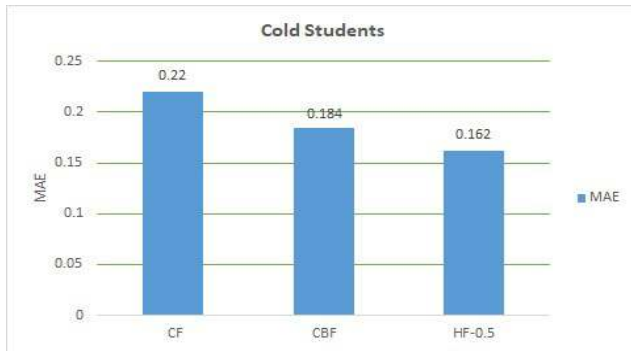


FIGURE 7. Performance comparison for cold students using MAE.

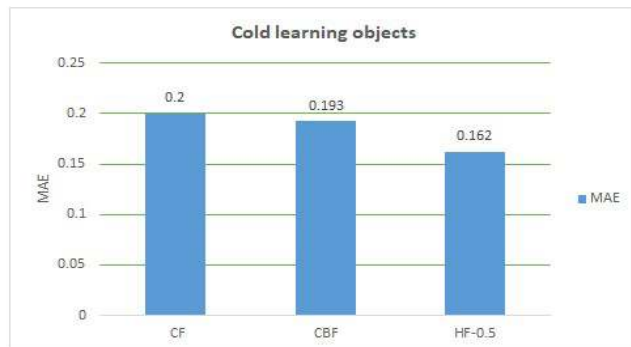


FIGURE 8. Performance comparison for cold students using RMSE.

randomly and the calculated predicted ratings for these learning objects are shown in Table 5. The experimental results in the table show that the HF-0.5 algorithm has the best accuracy. From Fig. 6, it can be inferred that HF-0.5 has the least value of MAE and thus, provides better predictions. The MAE value of HF-0.5 is 0.9, whilst that of CBF is 1.52, which is the greatest compared to all the other approaches. Hence, the latter method will produce the least accuracy in prediction. Theoretically and experimentally, it has already been proven that the root-mean-square error is always greater than the Mean Absolute Error. Figure 6 shows that the proposed HF-0.5 algorithm again delivers a smaller RMSE than the others, which indicates that it is the most accurate.

2) EVALUATION OF COLD-START

From another point of view, the experiments were repeated to evaluate the proposed approaches with respect to the handling of cold-start problem.

- **New students:** The three different algorithms (CF, CBF and HF-0.5) can deal with new students by incorporating their personalised learning styles with their rating. Fig. 7 and Fig. 8 compare the accuracy of the different recommendation algorithms. The results achieved by the hybrid filtering approach are impressive. Given the above results, analysis, and discussion, it is concluded that the proposed algorithm HF-0.5 performs better than CF and CBF.

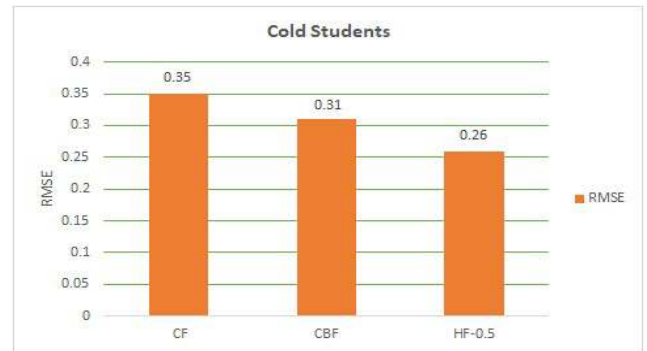


FIGURE 9. Performance comparison for cold LOs using MAE.

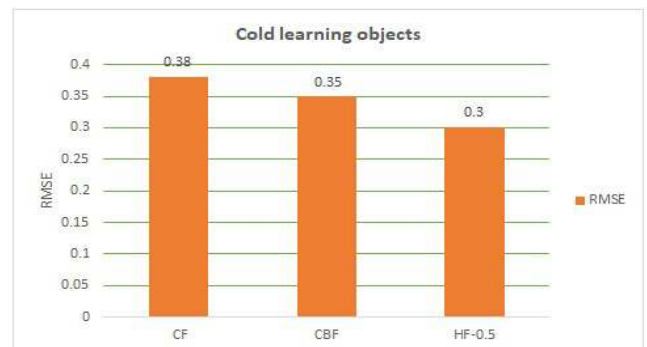


FIGURE 10. Performance comparison for cold LOs using RMSE.

- **New learning objects:** The three algorithms can make recommendation for new LOs by measuring the similarity between learning object profile and student learning styles. From Fig. 9 and Fig. 10 we observe that HF-0.5 consistently outperforms in all the experiments, which indicates that our model handles new items better than CF and CBF.
- **New students and learning objects:** One special case is where neither the student nor the LOs exist in the previous user-item rating matrix. Most of the existing algorithms cannot deal with this situation. However, our proposed algorithm can still make recommendations by considering the relations between student and LO profiles.

The results achieved by the hybrid filtering approach (HF-0.5) are impressive. Given the above results, analysis, and discussion, it is concluded that the proposed algorithm performs better than CF and CBF.

3) REAL STUDENT EVALUATION

The last part of the evaluation was to validate our method in real circumstances by integrating it in the AAST MOODLE. The system was modified to be able to read student profiles and, subsequently, recommend the course LOs. To evaluate the student satisfaction with recommendations, a closed-ended questionnaire was administered to the 80 students who participated in the experiment. Previous studies on recommender systems have identified user satisfaction as one

FIGURE 11. Questionnaire interface.

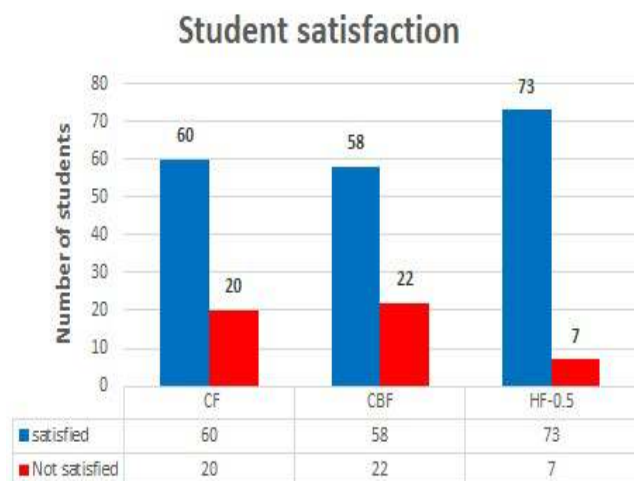


FIGURE 12. Student satisfaction with LO recommendations.

of the important evaluation measures [93]. First, they were asked to fill in the FSLSM questionnaire [63] to create their profile, as shown in Fig. 11. After that, they were required to take five lessons over a period of five weeks, with each including a set of LOs. The empirical study was carried out in the Business Information Systems department at AAST. The questionnaire sought to find out whether the learner was satisfied or not satisfied with the LO recommendations. Figure 12 illustrates the responses of the learners to the questionnaire with regards to the three recommendation algorithms.

From Fig. 12, it is evident that the majority (95%) of the students were satisfied with the LO recommendations from the HF-0.5 algorithm. On the other hand, just (60%) and (56%) of the students were satisfied with the recommendations from the CF and CBF recommendation algorithms, respectively.

V. CONCLUSION AND FUTURE DIRECTIONS

Whilst recommender systems have been studied in the past decade, the study of rating prediction for recommender systems is a more recent phenomenon. In this paper, we have concentrated on improving the accuracy and quality of recommendation in the case of cold start and data sparsity. To this end, an improved rating prediction algorithm has been proposed. We tested and compared the performances of three main algorithms: hybrid filtering, collaborative filtering, and content-based filtering. The results of the experiments shows that the proposed HF-0.5 hybrid algorithm provides the best prediction accuracy. It was also established that the HF-0.5 hybrid algorithm handles effectively the cold-start and rating sparsity problems.

However, this work has some limitations, which could be addressed in future work. First, the dataset in the current work was quite small and a larger one would add more weight to the findings. A second future direction is working on some other challenges of recommendation systems, such as scalability.

REFERENCES

- [1] S. Sosnovsky and P. Brusilovsky, "Evaluation of topic-based adaptation and student modeling in QuizGuide," *Model. Adapted Interact.* vol. 25, no. 4, pp. 371–424, 2015.
- [2] A. Bennane, "Adaptive educational software by applying reinforcement learning," *Inform. Educ.-Int. J.*, vol. 12, no. 1, pp. 13–27, 2013.
- [3] K. I. Ghauth and N. A. Abdullah, "Learning materials recommendation using good learners' ratings and content-based filtering," *Educ. Technol. Res. Develop.*, vol. 58, no. 6, pp. 711–727, 2010.
- [4] P. Vu, S. Fredrickson, and C. Moore, *Handbook of Research on Innovative Pedagogies and Technologies for Online Learning in Higher Education*. Hershey, PA, USA: IGI Global, 2016.
- [5] A. R. Anaya, M. Luque, and T. García-Saiz, "Recommender system in collaborative learning environment using an influence diagram," *Expert Syst. Appl.*, vol. 40, no. 18, pp. 7193–7202, 2013.
- [6] I. Kamsa, R. Elouahbi, F. El Khoukhi, T. Karite, and H. Zouiten, "Optimizing collaborative learning path by ant's optimization technique in e-learning system," in *Proc. 15th Int. Conf. Inf. Technol. Based Higher Educ. Training (ITHET)*, Sep. 2016, pp. 1–5.

- [7] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 6, pp. 734–749, Jun. 2005.
- [8] O. Bourkoku and E. El Bachari, "E-learning personalization based on collaborative filtering and learner's preference," *J. Eng. Sci. Technol.*, vol. 11, no. 11, pp. 1565–1581, 2016.
- [9] G. Adomavicius and A. Tuzhilin, "Context-aware recommender systems," in *Recommender Systems Handbook*. Boston, MA, USA: Springer, 2011, pp. 217–253.
- [10] R. Turnip, D. Nurjanah, and D. S. Kusumo, "Hybrid recommender system for learning material using content-based filtering and collaborative filtering with good learners' rating," in *Proc. IEEE Conf. e-Learn., e-Manage. e-Services (IC3e)*, Nov. 2017, pp. 61–66.
- [11] J. B. Schafer, D. Frankowski, J. Herlocker, and S. Sen, "Collaborative filtering recommender systems," in *The Adaptive Web*. Berlin, Germany: Springer, 2007, pp. 291–324.
- [12] L. Yao, Q. Z. Sheng, A. H. Ngu, J. Yu, and A. Segev, "Unified collaborative and content-based Web service recommendation," *IEEE Trans. Services Comput.*, vol. 8, no. 3, pp. 453–466, May/June 2015.
- [13] W. Jian, J. He, K. Chen, Y. Zhou, and Z. Tang, "Collaborative filtering and deep learning based recommendation system for cold start items," *Expert Syst. Appl.*, vol. 69, pp. 29–39, Mar. 2017.
- [14] E. Q. da Silva, C. G. Camilo, Jr., L. M. L. Pascoal, and T. C. Rosa, "An evolutionary approach for combining results of recommender systems techniques based on collaborative filtering," *Expert Syst. Appl.*, vol. 53, pp. 204–218, Jul. 2016.
- [15] J. A. R. Nt, L. F. C. Tomaz, J. M. de Souza, and G. Xexéo, "Bringing knowledge into recommender systems," *J. Syst. Softw.*, vol. 86, no. 7, pp. 1751–1758, 2013.
- [16] M. J. Pazzani and D. Billsus, "Content-based recommendation systems," in *The Adaptive Web*. Berlin, Germany: Springer, 2007, pp. 325–341.
- [17] R. M. Felder and L. K. Silverman, "Learning and teaching styles in engineering education," *Eng. Educ.*, vol. 78, no. 7, pp. 674–681, 1988.
- [18] C. A. Carver, Jr., R. A. Howard, and W. D. Lane, "Enhancing student learning through hypermedia courseware and incorporation of student learning styles," *IEEE Trans. Educ.*, vol. 42, no. 1, pp. 33–38, Feb. 1999.
- [19] J. Kuljis and F. Liu, "A comparison of learning style theories on the suitability for elearning," *Web Technol., Appl., Services*, vol. 2005, pp. 191–197, Jul. 2005.
- [20] F. Da and A. Geçer, "Relations between online learning and learning styles," *Procedia-Social Behav. Sci.*, vol. 1, no. 1, pp. 862–871, 2009.
- [21] P. García, A. Amandi, S. Schiaffino, and M. Campo, "Evaluating Bayesian networks' precision for detecting students' learning styles," *Comput. Educ.*, vol. 49, no. 3, pp. 794–808, 2007.
- [22] R. M. Felder and J. Spurlin, "Applications, reliability and validity of the index of learning styles," *Int. J. Eng. Educ.*, vol. 21, no. 1, pp. 103–112, 2005.
- [23] G. Salton and M. J. McGill, *Introduction to Modern Information Retrieval*. New York, NY, USA: McGraw-Hill, 1986.
- [24] S. Carr, "As distance education comes of age, the challenge is keeping the students," *Chronicle Higher Educ.*, vol. 46, no. 23, 2000.
- [25] F. A. Inan, E. Yukselturk, and M. M. Grant, "Profiling potential dropout students by individual characteristics in an online certificate program," *Int. J. Instruct. Media*, vol. 36, no. 2, pp. 163–177, 2009.
- [26] S. B. Kotsiantis, C. J. Pierrakeas, and P. E. Pintelas, "Preventing student dropout in distance learning using machine learning techniques," in *Proc. Int. Conf. Knowl.-Based Intell. Inf. Eng. Syst.* Berlin, Germany: Springer, Sep. 2003, pp. 267–274.
- [27] P. A. Willging and S. D. Johnson, "Factors that influence students' decision to dropout of online courses," *J. Asynchronous Learn. Netw.*, vol. 13, no. 3, pp. 115–127, Oct. 2009.
- [28] A. Niesler and G. Wydmuch, "User profiling in intelligent tutoring systems based on Myers-Briggs personality types," in *Proc. Int. Multiconf. Eng. Comput. Sci.*, vol. 1, 2009, pp. 18–20.
- [29] C.-M. Chen, H.-M. Lee, and Y.-H. Chen, "Personalized e-learning system using item response theory," *Comput. Educ.*, vol. 44, no. 3, pp. 237–255, 2005.
- [30] J. Lu, "A personalized e-learning material recommender system," in *Proc. Int. Conf. Inf. Technol. Appl.* Sydney, NSW, Australia: Macquarie Scientific, 2004, pp. 374–379.
- [31] F. Ricci, L. Rokach, and B. Shapira, "Introduction to recommender systems handbook," in *Recommender Systems Handbook*. Boston, MA, USA: Springer, 2011, pp. 1–35.
- [32] N. Soonthornphisaj, E. Rojsattarat, and S. Yim-Ngam, "Smart e-learning using recommender system," in *Proc. Int. Conf. Intell. Comput.* Berlin, Germany: Springer, Aug. 2006, pp. 518–523.
- [33] Z. Gulzar, A. A. Leema, and G. Deepak, "PCRS: Personalized course recommender system based on hybrid approach," *Procedia Comput. Sci.*, vol. 125, pp. 518–524, Jan. 2018.
- [34] N. Manouselis and C. Costopoulou, "Experimental analysis of design choices in multiattribute utility collaborative filtering," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 21, no. 2, pp. 311–331, 2007.
- [35] M. Yaghmaie and A. Bahreininejad, "A context-aware adaptive learning system using agents," *Expert Syst. Appl.*, vol. 38, no. 4, pp. 3280–3286, 2011.
- [36] M. M. Recker, A. Walker, and K. Lawless, "What do you recommend? Implementation and analyses of collaborative information filtering of Web resources for education," *Instruct. Sci.*, vol. 31, nos. 4–5, pp. 299–316, 2003.
- [37] G. Linden, B. Smith, and J. York, "Amazon.com recommendations: Item-to-item collaborative filtering," *IEEE Internet Comput.*, vol. 7, no. 1, pp. 76–80, Jan./Feb. 2003.
- [38] Q. Diao, M. Qiu, C.-Y. Wu, A. J. Smola, J. Jiang, and C. Wang, "Jointly modeling aspects, ratings and sentiments for movie recommendation (JMARS)," in *Proc. 20th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2014, pp. 193–202.
- [39] A. Klačnja-Milićević, B. Vesin, M. Ivanović, and Z. Budimac, "E-Learning personalization based on hybrid recommendation strategy and learning style identification," *Comput. Educ.*, vol. 56, no. 3, pp. 885–899, 2011.
- [40] E. Kurilovas, S. Kubilinskiene, and V. Dagiene, "Web 3.0—Based personalisation of learning objects in virtual learning environments," *Comput. Hum. Behav.*, vol. 30, pp. 654–662, Jan. 2014.
- [41] A. Latham, K. Crockett, and D. McLean, "An adaptation algorithm for an intelligent natural language tutoring system," *Comput. Educ.*, vol. 71, pp. 97–110, Feb. 2014.
- [42] P. Rodríguez, S. Heras, J. Palanca, N. Duque, and V. Julián, "Argumentation-based hybrid recommender system for recommending learning objects," in *Multi-Agent Systems and Agreement Technologies*. Cham, Switzerland: Springer, 2015, pp. 234–248.
- [43] H. L. dos Santos, C. Cechinel, R. M. Araujo, and M.-Á. Sicilia, "Clustering learning objects for improving their recommendation via collaborative filtering algorithms," in *Proc. Res. Conf. Metadata Semantics Res.* Cham, Switzerland: Springer, Sep. 2015, pp. 183–194.
- [44] P. Winoto and T. Y. Tang, "The role of user mood in movie recommendations," *Expert Syst. Appl.*, vol. 37, no. 8, pp. 6086–6092, 2010.
- [45] H. Pashler, M. McDaniel, D. Rohrer, and R. Bjork, "Learning styles: Concepts and evidence," *Psychol. Sci. Public Interest*, vol. 9, no. 3, pp. 105–119, 2008.
- [46] P. Honey and A. Mumford, *The Manual of Learning Styles*. Maidenhead, U.K.: McGraw-Hill, 1982.
- [47] D. A. Kolb, Englewood Cliffs, NJ, USA, 1984.
- [48] G. Pask, "Styles and strategies of learning," *Brit. J. Educ. Psychol.*, vol. 46, no. 2, pp. 128–148, 1976.
- [49] E. Popescu, C. Badica, and L. Moraret, "Accommodating learning styles in an adaptive educational system," *Informatica*, vol. 34, no. 4, 2010.
- [50] C.-I. Peña, J.-L. Marzo, and J.-L. de la Rosa, "Intelligent agents in a teaching and learning environment on the Web," in *Proc. Int. Conf. Adv. Learn. Technol.* Palmerston North, New Zealand: IEEE Learning Technology Task Force, Sep. 2002, pp. 21–27.
- [51] H. Imran, M. Belghis-Zadeh, T.-W. Chang, Kinshuk, and S. Graf, "PLOGS: A personalized learning object recommender system," *Vietnam J. Comput. Sci.*, vol. 3, no. 1, pp. 3–13, 2016.
- [52] P. Paredes and P. Rodríguez, "A mixed approach to modelling learning styles in adaptive educational hypermedia," in *Proc. Adv. Technol. Learn.*, vol. 1, no. 4, pp. 210–215, 2004.

- [53] E. El Bachari, E. H. Abelwahed, and M. El Adnani, "An adaptive teaching strategy model in e-learning using learners' preference: LearnFit framework," *Int. J. Web Sci.*, vol. 1, no. 3, pp. 257–274, 2012.
- [54] P. Resnick and H. R. Varian, "Recommender systems," *Commun. ACM*, vol. 40, no. 3, pp. 56–59, Mar. 1997.
- [55] R. Dunn, "Understanding the Dunn and Dunn learning styles model and the need for individual diagnosis and prescription," *Reading, Writing, Learn. Disabilities*, vol. 6, no. 3, pp. 223–247, 1990.
- [56] I. Briggs-Myers and P. B. Myers, *Gifts Differing: Understanding Personality Type*. 1995.
- [57] G.-J. Hwang, H.-Y. Sung, C.-M. Hung, and I. Huang, "A learning style perspective to investigate the necessity of developing adaptive learning systems," *J. Educ. Technol. Soc.*, vol. 16, no. 2, pp. 188–197, 2013.
- [58] Y.-C. Chang, W.-Y. Kao, C.-P. Chu, and C.-H. Chiu, "A learning style classification mechanism for e-learning," *Comput. Educ.*, vol. 53, no. 2, pp. 273–285, 2009.
- [59] E. Özpölat and G. B. Akar, "Automatic detection of learning styles for an e-learning system," *Comput. Educ.*, vol. 53, no. 2, pp. 355–367, 2009.
- [60] F. A. Dorça, L. V. Lima, M. A. Fernandes, and C. R. Lopes, "Comparing strategies for modeling students learning styles through reinforcement learning in adaptive and intelligent educational systems: An experimental analysis," *Expert Syst. Appl.*, vol. 40, no. 6, pp. 2092–2101, 2013.
- [61] E. Sanginetto, N. Capuano, M. Gaeta, and A. Micarelli, "Adaptive course generation through learning styles representation," *Universal Access Inf. Soc.*, vol. 7, nos. 1–2, pp. 1–23, 2008.
- [62] S. Graf and Kinshuk, "Advanced adaptivity in learning management systems by considering learning styles," in *Proc. IEEE/WIC/ACM Int. Joint Conf. Web Intell. Intell. Agent Technol.*, vol. 3, Sep. 2009, pp. 235–238.
- [63] S. M. Nafea, F. Siewe, and Y. He, "ULEARN: Personalised learner's profile based on dynamic learning style questionnaire," in *Proc. SAI Intell. Syst. Conf.* Cham, Switzerland: Springer, Sep. 2018, pp. 1105–1124.
- [64] S. M. Nafea, F. Siewe, and Y. He, "A novel algorithm for course learning object recommendation based on student learning styles," in *Proc. Int. Conf. Innov. Trends Comput. Eng. (ITCE)*, Feb. 2019, pp. 192–201.
- [65] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, and A. Y. Wu, "An efficient k-means clustering algorithm: Analysis and implementation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 881–892, Jul. 2002.
- [66] A. I. Schein, A. Popescul, L. H. Ungar, and D. M. Pennock, "Methods and metrics for cold-start recommendations," in *Proc. 25th Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Aug. 2002, pp. 253–260.
- [67] T. Y. Tang and G. McCalla, "Smart recommendation for an evolving e-learning system," in *Proc. Workshop Technol. Electron. Documents Supporting Learn. Int. Conf. Artif. Intell. Educ.*, Jul. 2003, pp. 699–710.
- [68] G. Liang, K. Weining, and L. Junzhou, "Courseware recommendation in e-learning system," in *Proc. Int. Conf. Web-Based Learn.* Berlin, Germany: Springer, Jul. 2006, pp. 10–24.
- [69] Z. Yu, Y. Nakamura, S. Jang, S. Kajita, and K. Mase, "Ontology-based semantic recommendation for context-aware e-learning," in *Proc. Int. Conf. Ubiquitous Intell. Comput.* Berlin, Germany: Springer, Jul. 2007, pp. 898–907.
- [70] H. Tan, J. Guo, and Y. Li, "E-learning recommendation system," in *Proc. Int. Conf. Comput. Sci. Softw. Eng.*, vol. 5, Dec. 2008, pp. 430–433.
- [71] K. I. B. Ghauth and N. A. Abdullah, "Building an e-learning recommender system using vector space model and good learners average rating," in *Proc. 9th IEEE Int. Conf. Adv. Learn. Technol.*, Jul. 2009, pp. 194–196.
- [72] J. L. Rodgers and W. A. Nicewander, "Thirteen ways to look at the correlation coefficient," *Amer. Statist.*, vol. 42, no. 1, pp. 59–66, 1988.
- [73] J. B. Schafer, "Application of data mining to recommender systems," in *Encyclopedia of Data Warehousing and Mining*. Hershey, PA, USA: IGI Global, 2005, pp. 44–48.
- [74] M. K. Khribi, M. Jenni, and O. Nasraoui, "Automatic recommendations for e-learning personalization based on Web usage mining techniques and information retrieval," in *Proc. 8th IEEE Int. Conf. Adv. Learn. Technol.*, Jul. 2008, pp. 241–245.
- [75] M. Salehi and I. N. Kmalabadi, "A hybrid attribute-based recommender system for e-learning material recommendation," *IERI Procedia*, vol. 2, pp. 565–570, Jan. 2012.
- [76] O. Bourkoukou, E. El Bachari, and M. El Adnani, "A recommender model in e-learning environment," *Arabian J. Sci. Eng.*, vol. 42, no. 2, pp. 607–617, 2017.
- [77] J. Golbeck and J. Hendler, "Inferring binary trust relationships in Web-based social networks," *ACM Trans. Internet Technol.*, vol. 6, no. 4, pp. 497–529, 2006.
- [78] Y. Ar and E. Bostanci, "A genetic algorithm solution to the collaborative filtering problem," *Expert Syst. Appl.*, vol. 61, pp. 122–128, Nov. 2016.
- [79] K. Hammouda and M. Kamel, "Collaborative document clustering," in *Proc. SIAM Int. Conf. Data Mining*, Apr. 2006, pp. 453–463.
- [80] J. Bobadilla, F. Serradilla, and A. Hernando, "Collaborative filtering adapted to recommender systems of e-learning," *Knowl.-Based Syst.*, vol. 22, no. 4, pp. 261–265, 2009.
- [81] W. Chen, Z. Niu, X. Zhao, and Y. Li, "A hybrid recommendation algorithm adapted in e-learning environments," *World Wide Web*, vol. 17, no. 2, pp. 271–284, 2014.
- [82] E. Kurilovas, I. Zilinskiene, and V. Dagiene, "Recommending suitable learning scenarios according to learners' preferences: An improved swarm based approach," *Comput. Hum. Behav.*, vol. 30, pp. 550–557, Jan. 2014.
- [83] R. A. W. Tortorella and S. Graf, "Considering learning styles and context-awareness for mobile adaptive learning," *Educ. Inf. Technol.*, vol. 22, no. 1, pp. 297–315, 2017.
- [84] B. Vesin, M. Ivanović, A. Klačnja-Milićević, and Z. Budimac, "Protus 2.0: Ontology-based semantic recommendation in programming tutoring system," *Expert Syst. Appl.*, vol. 39, no. 15, pp. 12229–12246, 2012.
- [85] Ö. Özyurt, H. Özyurt, A. Baki, B. Güven, and H. Karal, "Evaluation of an adaptive and intelligent educational hypermedia for enhanced individual learning of mathematics: A qualitative study," *Expert Syst. Appl.*, vol. 39, no. 15, pp. 12092–12104, 2012.
- [86] B. C. L. Christudas, E. Kirubakaran, and P. R. J. Thangaiah, "An evolutionary approach for personalization of content delivery in e-learning systems based on learner behavior forcing compatibility of learning materials," *Telematics Informat.*, vol. 35, no. 3, pp. 520–533, 2018.
- [87] S. Schiaffino, P. Garcia, and A. Amandi, "eTeacher: Providing personalized assistance to e-learning students," *Comput. Educ.*, vol. 51, no. 4, pp. 1744–1754, 2008.
- [88] A. Latham, K. Crockett, D. McLean, and B. Edmonds, "A conversational intelligent tutoring system to automatically predict learning styles," *Comput. Educ.*, vol. 59, no. 1, pp. 95–109, 2012.
- [89] G. Guo, "Improving the performance of recommender systems by alleviating the data sparsity and cold start problems," in *Proc. 23rd Int. Joint Conf. Artif. Intell.*, Jun. 2013.
- [90] P. Honey and A. Mumford, *The Manual of Learning Styles*. 1992.
- [91] M. P. P. Liyanage, K. S. L. Gunawardena, and M. Hirakawa, "Detecting learning styles in learning management systems using data mining," *J. Inf. Process.*, vol. 24, no. 4, pp. 740–749, 2016.
- [92] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, "GroupLens: An open architecture for collaborative filtering of netnews," in *Proc. ACM Conf. Comput. Supported Cooperat. Work*, Oct. 1994, pp. 175–186.
- [93] M. Erdt, A. Fernández, and C. Rensing, "Evaluating recommender systems for technology enhanced learning: A quantitative survey," *IEEE Trans. Learn. Technol.*, vol. 8, no. 4, pp. 326–344, Oct./Dec. 2015.
- [94] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," *Inf. Process. Manage.*, vol. 24, no. 5, pp. 513–523, 1988.
- [95] H. Drachsler, K. Verbert, O. C. Santos, and N. Manouselis, "Panorama of recommender systems to support learning," in *Recommender Systems Handbook*. Boston, MA, USA: Springer, 2015, pp. 421–451.

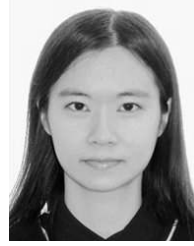


SHAIMAA M. NAFAA received the B.S. degree in business informatics and the M.S. degree in information system mechanical engineering from Arab Academy for Science, Technology & Maritime Transport, Egypt, in 2009 and 2013, respectively. She is currently pursuing the Ph.D. degree in computer science with De Montfort University, U.K. Her main research interests include in adaptation and personalization of e-learning environment, learning styles, algorithms, recommender

systems and collaborative filtering, artificial intelligence, and machine learning.



FRANÇOIS SIEWE received the B.Sc. degree in mathematics and computer science, the M.Sc. and Doctorat de Troisième Cycle degrees in computer science from the University of Yaounde I, Cameroon, and the Ph.D. degree in computer science from De Montfort University (DMU), U.K. He is a Reader in computer science and the Head of the Software Technology Research Laboratory (STRL), School of Computer Science and Informatics, De Montfort University (DMU). His research interests include software engineering, formal methods, cyber security, context-aware and pervasive computing, and the Internet of Things (IoT).



YING HE received the Ph.D. degree in computer science from Glasgow University, U.K. She is currently a Senior Lecturer of computer science with the School of Computer Science and Informatics, De Montfort University, U.K. Her research interests include cyber threat intelligence, security risk management, decision-making, business analytics, control systems architecture, and human's aspects of security. She also looks at how security management frameworks can be applied in different industries, such as healthcare and finance.

• • •