A Novel Algorithm for Course Learning Object Recommendation Based on Student Learning Styles

Shaimaa M. Nafea, François Siewe, Ying He

Abstract—Explosive growth of e-learning in the recent years has faced difficulty of locating appropriate learning resources to match the students learning styles. Recommender system is a promising technology in e-learning environments to present personalised offers and convey appropriate learning objects that match student inclinations. This paper, proposes a novel and effective recommender algorithm that recommends personalised learning objects based on the student learning styles. Various similarity metrics are considered in an experimental study to investigate the best similarity metrics to use in a recommender system for learning objects. The approach is based on the Felder and Silverman learning style model which is used to represent both the student learning styles and the learning object profiles. It was found that the K-means clustering algorithm, the cosine similarity metrics and the Pearson correlation coefficient are effective tools for implementing learning object recommender systems. The accuracy of the recommendations are measured using traditional evaluation metrics, namely the Mean Absolute Error and the Root Mean Squared Error.

Index Terms—Recommender system, learning object profile, student profile, algorithms, Learning style model, similarity metric, K-means clustering.

1 Introduction

E-learning recommender systems aim to recommend a sequence of learning objects to students, that is to suggest the most efficient or effective paths through a plethora of learning resources to achieve a certain competence [1] [2]. However, over specification and excessive searching in e-learning recommender systems result in information overload. Students do not have enough time to deal with these massive recommendations. In addition, the adaptability and diversity of recommendations are desirable in e-learning recommender systems, because students preferences and abilities keep changing, and also because the functionality of some learning resources for active students keeps changing. The diverse and adaptive recommendations should be presented to motivate learning potential of students and ensure a long-term learning experience.

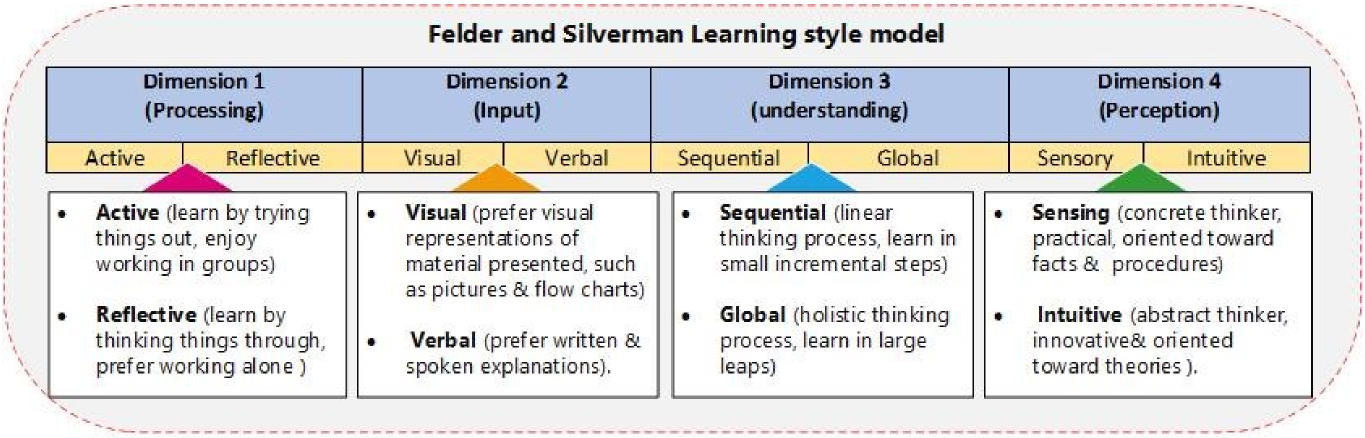
Many authors stress that personalised learning improves students achievements and increases learning efficiency [3] [4] [5]. Therefore, explosive growth of e-learning has caused difficulty of locating the most appropriate learning objects to achieve positive educational experiences that fits learning styles of their students. Learning styles describe the preferences a student has for how material is presented, how to work with material and how to internalise information [6]. Knowing a students learning styles can help in several ways to improve the learning process. For example, personalising content to the learning styles of students has been found to be beneficial to learning in several ways such as improving satisfaction [7], learning outcomes, and reducing the time needed to learn [8]. Students may also be enlightened by understanding their own learning styles [6].

In this study, the Felder and Silverman Learning Style Model (FSLSM) [9] is used since it is the most widely used in educational systems thanks to its ability to quantify students' learning style [10] [11]. FSLSM is a learning style model that describes learning styles in detail and is therefore highly appropriate for providing adaptively in learning systems. Furthermore, FSLSM 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 [12], [13], [14] as well as, it is easy to implement. Furthermore the learning styles can be obtained by means of the Index of Learning Styles (ILS) instrument, a questionnaire based on 44 items to which each student responds according to their learning preferences [6] as shown in Fig. 1.

The results obtained from the ILS distinguish the students preferences according to the four dimensions (active/reflective, sensing/intuitive, visual/verbal, and sequential/global) of the model, and they allow for describing trends about stronger and weaker preferences by means of a numeric scale. For example Active students prefer doing tasks or talking about concepts, while reflective students are likely to manipulate and examine the information introspectively [9]. The success of e-learning has created huge amounts of learning objects which makes locating suitable ones a real big challenge. Moreover, selecting appropriate learning resources to compose a suitable learning path for e-learning students is a complex task [15] [16]. One way to address this issue is to use recommender system techniques

- *Shaimaa M. Nafea with the Department of management information system, Arab Academy For Science Technology and Maritime, Egypt.*
François Siewe and Ying He with the Department of Computer Science and Informatics, De Montfort University, Leicester, UK.
E-mail: P15017421@myemail.dmu.ac.uk, fsiewe@dmu.ac.uk, ying.he@dmu.ac.uk
-

Manuscript received April 19, 2005; revised August 26, 2015.



- **Active** (learn by trying things out, enjoy working in groups)
- **Reflective** (learn by thinking things through, prefer working alone)

- **Visual** (prefer visual representations of material presented, such as pictures & flow charts)
- **Verbal** (prefer written & spoken explanations).

- **Sequential** (linear thinking process, learn in small incremental steps)
- **Global** (holistic thinking process, learn in large leaps)

- **Sensing** (concrete thinker, practical, oriented toward facts & procedures)
- **Intuitive** (abstract thinker, innovative& oriented toward theories).

Fig. 1: FSLSM learning style model

to personalise learning process according to the interests and goals of each student. They recommend Learning Object (LO) based on explicit or implicit students preferences. They help in alleviating the problem of information overload, showing users the most interesting learning objects according to some criteria, offering relevance and diversity [17].

Recommender systems are divided in four different groups:

1) **Collaborative filtering:** A technique for generating recommendations in which the similarity of opinions on a set of LOs is used to predict their similarity of opinions on other LOs.
2) **Content-based:** A technique for creating recommendations based on the contents of the LOs and the profiles of the students preferences.
3) **Knowledge-based:** A technique for providing recommendations using knowledge models about the students and the LOs to reason about which LOs meet students requirements.
4) **Hybrid:** A technique to generate recommendations that combines two or more of the previous techniques to build its recommendations.

Fig. 2 shows the general e-learning recommendation process: first the student model is extracted from student profiles, then the adaptive module builds the adaptive concept selection rules or content selection rules through analysing the relationships between the students and learning objects models. The recommendation module executes the adaptive matching rules and provides the recommendations.

1.1 Current Challenges

E-learning recommendation systems still face the following problems:

1) The majority of the traditional recommendation algorithms have been developed for e-commerce applications, which are unable to cover the entire requirements of learning environments. They do not consider the learning process in their recommendation approach.
2) Most of existing personalised recommendation systems which are based on clustering approach suffer from low accuracy level because the risk of creating clusters that include data points that are actually not too close, the risk of getting different clusters when running the same clustering algorithm again, meaning that the clustering algorithm does not always group the nearest data points.
3) Usually, e-learning systems follow a one-size-fits all approach in which a same learning sequence or same learning materials are provided to all students by considering only common aspects of those students. However, these students vary in their learning styles, knowledge levels, background and goals [18].
4) It is necessary to consider the attributes of object profile such as learning styles in order to provide a good recommendation in the learning environment. Therefore, in order to improve the quality and accuracy of recommendations in the learning environment, this research takes the multidimensional attributes of learning object into account in order to personalise learning path by selecting and sequencing the most appropriate object profile according to students learning styles.

1.2 Contribution

Dynamically adaptive e-learning systems that continuously adapt course learning objects to the learners according to their learning styles during on-line sessions are becoming popular in the research community. This paper, proposes a novel and effective recommender algorithm that recommends personalised learning objects based on the student learning styles (Sect. 3). Various similarity metrics are considered in an experimental study to investigate the best similarity metrics to use in a recommender system for learning objects (Sect. 4). The approach is based on the Felder and Silverman learning style model which is used to represent both the student learning styles and the learning object profiles. It was found that the K-means clustering algorithm, the cosine similarity metrics and the Pearson correlation coefficient are effective tools for implementing learning object recommender systems (Sect. 4.4). The accuracy of the recommendations are measured using traditional

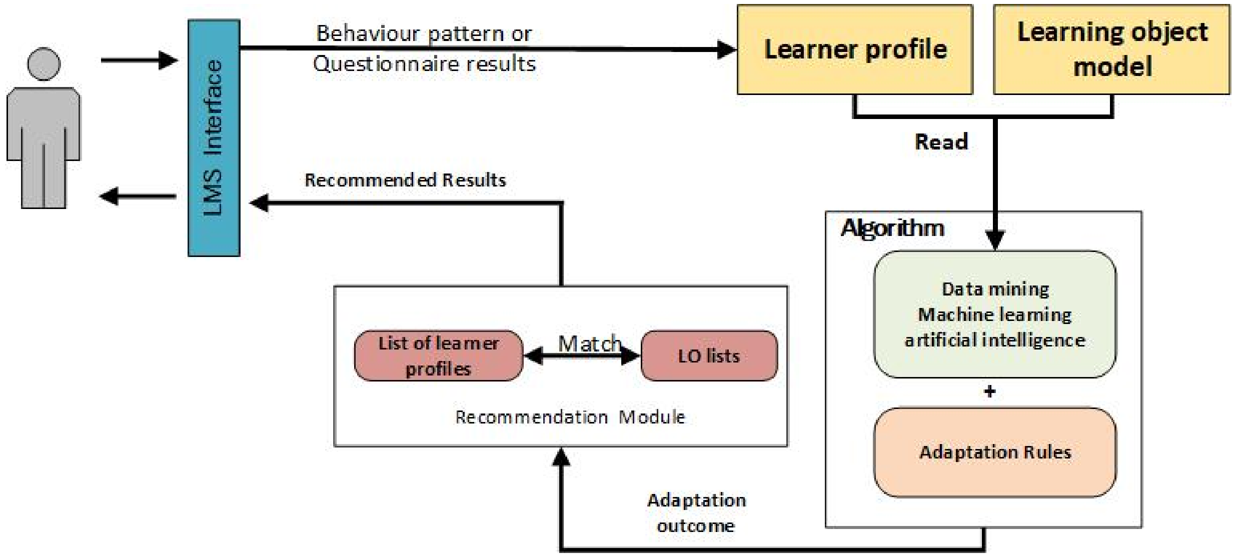


Fig. 2: General e-learning recommendation process

evaluation metrics, namely the Mean Absolute Error (MAE) and the Root Mean Squared Error (RMSE).

The remainder of the paper is organised as follows. The next section defines the main concepts used in the proposed approach. Sect. 3 discusses the proposed recommender algorithm. Sect. 4 presents the experimental results and analysis. Related works are discussed in Sect. 5; and Sect. 6 concludes the paper and points to future directions.

2 PRELIMINARIES

This section describes the similarity metrics commonly used in recommender systems. These will be analysed later to improve the accuracy of recommendations. An overview of the K-means clustering algorithm is also presented.

2.1 Similarity Metrics

Recommender systems contain many similarity metrics that come from machine learning. They are used to measure the closeness (or the distance) between two features vectors. As such they are very important in recommender systems in order to determine which items are more likely to interest a particular user. In order to identify the appropriate similarity metrics to be used in the recommender system proposed in this paper, it was important to review the common similarity metrics used in e-learning recommendation systems as shown in Table 1.

Let $x = (x_1, \cdots, x_n)$ and $y = (y_1, \cdots, y_n)$ be two vectors in an n-dimensional real vector space, for some integer $n > 0$.

2.1.1 Euclidean Distance

The Euclidean distance of two vectors x and y is given by Eq. (1) [25]:

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

The Euclidean distance is always greater than or equal to 0. The measurement is 0 for identical vectors and high for vectors that show little similarity. It is the most commonly used for clustering continuous data in algorithms such as K-means and Fuzzy C-means [26].

If the coordinates of the vectors x and y range from 0 to 1 (which is the case for the feature vectors used in this paper to represent the student learning styles and the learning object profiles, see Sect. 3), i.e. $0 \leq x_i, y_i \leq 1$ for all $1 \leq i \leq n$, then the Euclidean distance can be normalised as follows.

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

Note that $0 \leq d(x, y) \leq 1$ and the greater the value of $d(x, y)$ the more dissimilar x and y are. Therefore a similarity metric can be derived from Eq. (2) as follows.

$$Sim_d(x, y) = 1 - d(x, y) \tag{3}$$

The value of $Sim_d(x, y)$ is also between 0 and 1; however, a greater value indicates a stronger similarity between x and y.

2.1.2 Manhattan Distance

The Manhattan distance of two vectors x and y in an n-dimensional real vector space is the sum of the absolute differences of their Cartesian coordinates, as defined in Eq. (4).

$$M(x, y) = \sum_{i=1}^{n} |x_i - y_i| \tag{4}$$

It follows that $M(x, y) = 0$ if x and y are identical, otherwise $M(x, y) > 0$. Like for the Euclidean distance, the Manhattan distance can be normalised to fall between 0 and 1 for vectors whose coordinates are non-negative real numbers less than or equal to 1, as defined in Eq. (5).

$$m(x, y) = \frac{1}{n} \sum_{i=1}^{n} |x_i - y_i| \tag{5}$$

Likewise, a similarity metric can be derived from Eq. (5) as follows.

$$Sim_m(x, y) = 1 - m(x, y) \tag{6}$$

TABLE 1: A Summary of Related Work

✓: indicates fully supported; 0: indicates not supported

Study	Adaptation	Euclidean	Pearsons	Cosine	Jaccard	Manhattan
[19]	Learning objects adaptation	0	0	✓	0	0
[20]	Similarity degree between learners	0	✓	✓	0	0
[21]	Measure similarity between the rating vectors of learners	0	✓	0	0	0
[22]	Compute the similarity between learners	✓	0	0	0	0
[23]	Compute the similarity between learners	0	0	0	✓	0
[24]	Compute the similarity between students	0	0	0	0	✓

2.1.3 *Pearson Correlation*

The Pearson correlation coefficient is a measure of the linear dependence between two variables (real-valued vectors). The Pearson correlation coefficient 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) [27] and it can be equivalently defined by Eq. (7).

$$P(x,y) = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}} \tag{7}$$

where $\overline{x}$ and $\overline{y}$ 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 variables. 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.

2.1.4 *Cosine Similarity*

The cosine similarity is one of the most popular similarity metrics, which measures the angle between two vectors [28] and is calculated in Eq. (8) as the ratio of the scalar product by the product of the magnitudes.

$$c(x,y) = \frac{x.y}{||x||.||y||} \tag{8}$$

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.

2.2 K-means Clustering Algorithm

Clustering 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 objects belonging to different clusters have a high degree of dissimilarity. The K-means algorithm [29] is one of the most popular clustering algorithms due to its simplicity and intuitive interpretations. The algorithm is depicted by the flowchart in Fig. 3 and can be described by the following steps.

Step 1: Select K random points from the data-set as initial cluster centroids.

Step 2: Create K cluster by associating each data point with their closest cluster centroid according to the Euclidean distance defined Eq. (1).

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 centroids no longer change.

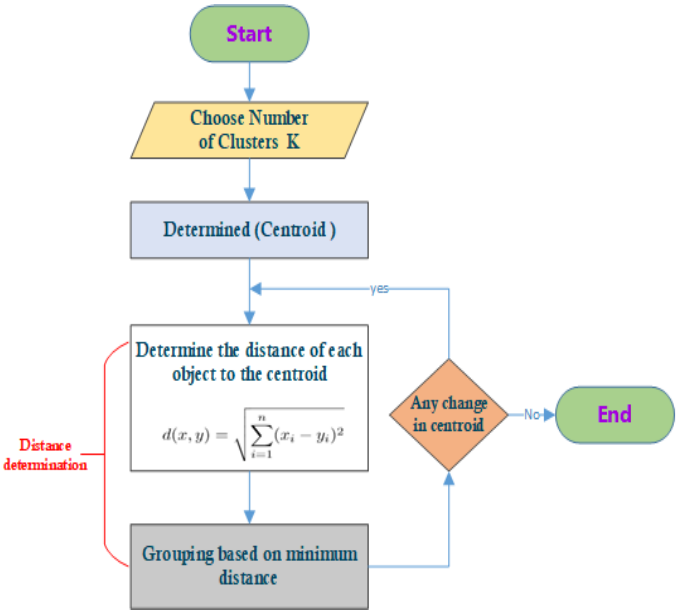

Start
Choose Number of Clusters K
Determined (Centroid)
yes
Determine the distance of each object to the centroid
$d(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$
Distance determination
Any change in centroid
No
End
Grouping based on minimum distance

Fig. 3: K-means clustering algorithm

Clustering improves the efficiency of recommender systems because it can be used off-line to partition the dataset such that similar data are grouped in the same cluster and dissimilar data are put in different clusters. Then the recommendation algorithm is applied only to the cluster with highest similarity to the active user.

2.3 The Felder-Silverman Learning Style Model

Learning style is the manner in which one learns best. It is based on individual characteristics and preferences. Theories of learning styles suggest that people differ in the ways they perceive, process and receive information. Each learner has their own learning style which if identified accurately could be used to provide effective and personalised teaching to significantly boost their performance in learning. The

Fielder-Silverman learning style model (FSLSM) [9] is considered the most stable learning style model and comprises four dimensions as depicted in Fig. 1.

The *information processing* dimension (active/reflective) tells how you prefer to process information. An active learner prefers to try things out, working with others in groups; while a reflective learner prefers thinking things through, working alone or with familiar partner.

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

The *information understanding* dimension (sequential/global) determines how you prefer to organise and progress toward understanding information. A sequential learner prefers linear thinking, orderly, and learns in small incremental steps. In the contrary, a global learner prefers holistic thinking, systems thinkers, and learns in large leaps.

The *information perception* dimension (sensing/intuitive) states how you prefer to perceive or take in information. A sensing learner prefers concrete thinking, practical, concerned with facts and procedures; while an intuitive learner prefers conceptual thinking, innovative, and concerned with theories and meanings.

Note 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 that 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 and A Soloman [30] developed the Index of Learning Styles (ILS) questionnaire that comprises 44 questions 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 [30], [31].

The next section presents a novel algorithm for recommending learning objects based on the student learning style.

3 Proposed algorithm for recommending personalised learning objects

In this section a study is carried out to devise an effective algorithm for recommending personalised learning objects in e-learning systems based on student learning styles. The FSLSM learning style model described above is adopted to represent both the student learning preferences and the learning object profiles. First a skeleton algorithm is presented (Sect. 3.2) that contains two abstract similarity metrics as place holders. Then an experimental study is done by replacing these abstract similarity metrics by the actual similarity metrics defined in Sect. 2.1 to find out which combination produces the best accuracy for the recommender algorithm. The best performing algorithm is then retained for the recommender system.

TABLE 2: 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

TABLE 3: Examples of learning object profile

	act	**ref**	**vis**	**ver**	**seq**	**glo**	**sen**	**int**
$\mathbf{OP_1}$	0.7	0.3	0.5	0.5	1	0	0.3	0.7
$\mathbf{OP_2}$	0.2	0.8	0.8	0.2	1	0	0.4	0.6
$\mathbf{OP_3}$	0.5	0.5	0.2	0.8	0.1	0.9	0.6	0.4
$\mathbf{OP_4}$	0.5	0.5	0.5	0.5	0.5	0.5	0.3	0.7
$\mathbf{OP_5}$	0.9	0.1	0.7	0.3	0.3	0.7	0.1	0.9
$\mathbf{OP_6}$	0.1	0.9	0.9	0.1	0.2	0.8	0.9	0.1
$\mathbf{OP_7}$	0.2	0.8	0.1	0.9	0.1	0.9	0.8	0.2
$\mathbf{OP_8}$	0.3	0.7	0.6	0.4	0.6	0.4	0.7	0.3
$\mathbf{OP_9}$	0.4	0.6	0.5	0.5	0.5	0.5	0.5	0.5
$\mathbf{OP_{10}}$	0.8	0.2	0.7	0.3	1	0	0.2	0.8
$\mathbf{OP_{11}}$	1	0	0.7	0.3	0.5	0.5	0.3	0.7
$\mathbf{OP_{12}}$	0.2	0.8	0.6	0.4	0.8	0.2	0.8	0.2
$\mathbf{OP_{13}}$	0.9	0.1	0.4	0.6	0.2	0.8	0.5	0.5
$\mathbf{OP_{14}}$	0.5	0.5	0.2	0.8	0.4	0.6	0.8	0.2
$\mathbf{OP_{15}}$	0.7	0.3	0.9	0.1	0.3	0.7	0.1	0.9
$\mathbf{OP_{16}}$	0.8	0.2	0.1	0.9	0.9	0.1	0.1	0.9
$\mathbf{OP_{17}}$	0.9	0.1	0.2	0.8	0.1	0.9	0.8	0.2
$\mathbf{OP_{18}}$	0	1	0.8	0.2	0.7	0.3	0.2	0.8
$\mathbf{OP_{19}}$	0.5	0.5	0.6	0.4	0.5	0.5	0.5	0.5
$\mathbf{OP_{20}}$	0.3	0.7	0.5	0.5	0.5	0.5	0.5	0.5

3.1 Assumptions

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 follows, where the prefixes of learning style attributes are used as place holders.

$$LS = (act, ref, vis, ver, seq, glo, sen, int) \tag{9}$$

Some examples of student learning style vectors are given in Table 2. As stated in Sect. 2.3, the learning style vector can be calculated using e.g. the learner's responses to the ILS questionnaire.

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 learner will prefer the opposite. Therefore, a learning object profile (OP) can be represented by a FSLSM learning style vector indicating the category of learners that this learning object is suitable to as in Eq. (10).

$$OP = (act, ref, vis, ver, seq, glo, sen, int) \tag{10}$$

Unlike the student learning styles that are calculated through the ILS questionnaire, 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 3 for illustration.

Finally, it is assumed that for each lesson the learning objects are clustered off-line to partition them into groups of similar profiles using the K-means clustering algorithm described in Sect.2.2. This process is repeated each time there is a change to the database of learning objects. These clusters are used as input to the proposed recommender algorithm presented in the following section.

3.2 Skeleton of the Recommender Algorithm

Let LS be the learning style vector of the active student. The recommender algorithm is described by the following 5 steps where Sim_1 and Sim_2 denote any of the similarity metrics defined in Sect. 2.1.

Step 1 *– Calculate C, the nearest learning object cluster to the active student learning style LS using the similarity metric* Sim_1: This is done by calculating the similarity degree between LS and the centroid of each cluster and choose the cluster that produces the highest similarity degree.

Step 2 *– Calculate the similarity degree between* LS *and each learning object in* C *using the similarity metric* Sim_2: For all $OP \in C$, calculate $Sim_2(LS, OP)$.

Step 3 *– Select the top-n learning objects most similar to* LS: The number of learning objects to be selected can be a chosen constant or determined using a similarity threshold.

Step 4 *– Predict the Student's ratings of the top-n learning objects*: A 5-level Likert scale is considered, with 1 be the lowest score and 5 the highest score. The learner ratings of the learning objects are predicted using Eq. (11).

$$\tilde{r}(LS, OP_i) = int(0.5 + Sim_2(LS, OP_i) \times 5), \quad 1 \leq i \leq n \tag{11}$$

where $\tilde{r}(LS, OP_i)$ is the predicted rating of the learning object (profile) OP_i by the active student LS and $int(x)$ denotes the closest integer to the real value x; e.g. $int(2.3) = 2$ and $int(2.5) = 3$.

Step 5 *– Rank learning objects in descending order of predicted ratings*: This constitutes the list of learning objects recommended to the active student.

The following section illustrates how the algorithm works using an example.

3.3 An Example

Consider the learning objects described in Table 3, where each learning object profile is represented by a learning style vector indicating the kind of learners this learning object is suitable to. The K-means clustering algorithm (see Sect. 2.2) is applied with K = 3 to partition these learning objects into 3 clusters as depicted in Fig. 4, where CC_i denotes the centroid of the cluster C_i, $1 \leq i \leq 3$. The centroids of the clusters are described in Table 4. In this example, the recommender algorithm presented in Sect. 3.2 is instantiated by taking the similarity metric Sim_1 to be the cosine similarity defined in Eq. (8) and the similarity metric Sim_2 to be the Pearson correlation coefficient defined in Eq. (7). Hence, the recommender algorithm works as follows for the active student Fatima whose learning style vector LS is given in Table 2, i.e.

$$LS = (0.7, 0.3, 0.2, 0.8, 0.5, 0.5, 0.6, 0.4).$$

Step 1 *– Calculate C, the nearest learning object cluster to the active student learning style LS using the similarity metric* Sim_1:

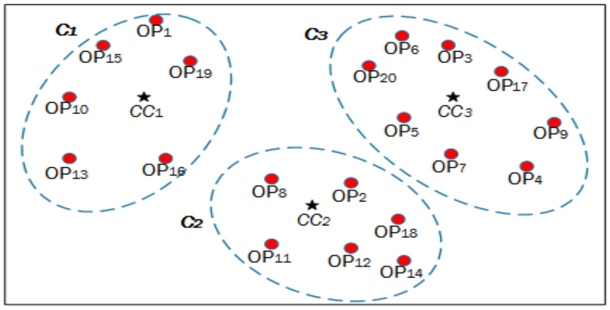


Fig. 4: Learning objects clustering

TABLE 4: Description of the clusters' centroids

	act	ref	vis	ver	seq	glo	sen	int
CC_1	0.7	0.3	0.5	0.5	0.65	0.35	0.3	0.7
CC_2	0.4	0.6	0.6	0.4	0.75	0.25	0.5	0.5
CC_3	0.5	0.5	0.45	0.55	0.3	0.7	0.6	0.4

- $Sim_1(LS, CC_1) = c(LS, CC_1) = 0.90$
- $Sim_1(LS, CC_2) = c(LS, CC_2) = 0.84$
- $Sim_1(LS, CC_3) = c(LS, CC_3) = 0.93$

Therefore, the nearest cluster is $C = C_3$.

Step 2 *– Calculate the similarity measurement between* LS *and each learning object in* C *using the similarity metric* Sim_2:

- $Sim_2(LS, OP_3) = P(LS, OP_3) = 0.52$
- $Sim_2(LS, OP_4) = P(LS, OP_4) = -0.27$
- $Sim_2(LS, OP_5) = P(LS, OP_5) = -0.08$
- $Sim_2(LS, OP_6) = P(LS, OP_6) = -0.57$
- $Sim_2(LS, OP_7) = P(LS, OP_7) = 0.34$
- $Sim_2(LS, OP_9) = P(LS, OP_9) = -0.53$
- $Sim_2(LS, OP_{17}) = P(LS, OP_{17}) = 0.76$
- $Sim_2(LS, OP_{20}) = P(LS, OP_{20}) = -0.53$

Step 3 *– Select the top-n learning objects most similar to* LS: With a similarity threshold of 0.25, the top-n learning objects are: OP_3, OP_7, and OP_{17}.

Step 4 *– Predict the Student's ratings of the top-n learning objects*:

- $\tilde{r}(LS, OP_3) = int(0.5 + 0.52 \times 5) = 3$
- $\tilde{r}(LS, OP_7) = int(0.5 + 0.34 \times 5) = 2$
- $\tilde{r}(LS, OP_{17}) = int(0.5 + 0.76 \times 5) = 4$

Step 5 *– Rank learning objects in descending order of predicted ratings*: The list of recommended leaning objects in descending order of predicted ratings is OP_{17}, OP_7 and OP_3.

Note that if other similarity metrics are used for Sim_1 and Sim_2 then the prediction of the student ratings of the learning objects may be different. Therefore, an important question is "what are the similarity metrics that provide the best prediction of the student ratings of the learning objects using the proposed recommender algorithm?" An experimental study is carried out in an attempt to find the answer to that question.

4 Experimental Study

An experimental study was carried out at the Arab Academy for Science Technology and Maritime (AAST) in

Cairo in Egypt, to determine the most effective metrics to be used for the recommendation of learning objects in e-learning systems. The skeleton recommender algorithm proposed in Sect. 3.2 was used as the basis of the study, aiming to find the best values for Sim_1 and Sim_2.

4.1 Dataset

The dataset of MOODLE log-file at AAST is used in this study for the fall and spring semesters in 2016/2017 and 2017/2018 in the school of business. 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 were a minimum of 15 learning objects for each topic. The experimental set up consisted of 30 students whose learning styles were identified using the ILS questionnaire as explained in [32]. During the course, the students were asked to rate each learning object using a 5-level Likert scale; with 1 be "not at all useful" and 5 be "very useful" to their learning.

4.2 Evaluation Metrics

To evaluation the quality of the recommender algorithm, 2 metrics were used to calculate the accuracy of the recommendations: the Mean Absolute Error (MAE) and the Root Mean Squared Error (RMSE). These are the most commonly used metrics to measure accuracy in recommender systems. Both metrics measure the average magnitude of the errors in a set of predictions, without considering their direction. They range from 0 to ∞ and the smaller their value, the greater the accuracy. They are defined by Eq. (12) and Eq. (13), where r_i is the actual student rating of the learning object i and $\tilde{r}_i$ is the predicted student rating for that learning object, $1 \leq i \leq n$.

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|r_i - \tilde{r}_i| \tag{12}$$

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(r_i - \tilde{r}_i)^2} \tag{13}$$

RMSE is useful to detect large errors in the prediction that may not be observed through MAE.

4.3 Experimental Platform

The skeleton algorithm was implemented in C++ using Visual studio and Windows Presentation Foundation (WPF) to design the GUI (graphical user interface). All the experiments are run on a Windows based PC with Intel core i5 processor having a speed of 2.40 GHz and 16GB of RAM. The GUI which allows to select various combinations of similarity metrics is depicted in Fig. 5.

4.4 Experimental Result and Analysis

It was found that the similarity metric Sim_1 used to determine the nearest learning object cluster to the active student learning style has no significant effect on the accuracy of the recommender algorithm. Indeed, Fig. 6 shows the distributions across K=3 clusters of a random sample of 30 students using the similarity metrics Euclidean (Eq. (3)), Manhattan (Eq. (6)), Person correlation coefficient (Eq. (7)), and cosine (Eq. (8)), respectively. It can be seen that there is no significant difference between these distributions. This means that any of this similarity metrics can be used for Sim_1 in skeleton recommender algorithm proposed in Sect. 3.2.

Now that it is clear what metric (i.e. Sim_1) can be used to calculate the nearest learning object cluster to a student learning style, the next step is to find the most effective metric (i.e. Sim_2) to select the top-n nearest learning objects to the student learning style within a cluster of learning objects. Experimental results show that the skeleton recommender algorithm has the best accuracy when the Pearson correlation coefficient (Eq. (7)) is substituted to Sim_2, as it can be seen in Fig. 7 and Fig. 8 according to the evaluation metrics MAE and RMSE respectively.

Based on this study the final recommender algorithm is presented in the following section.

4.5 Final Recommender Algorithm

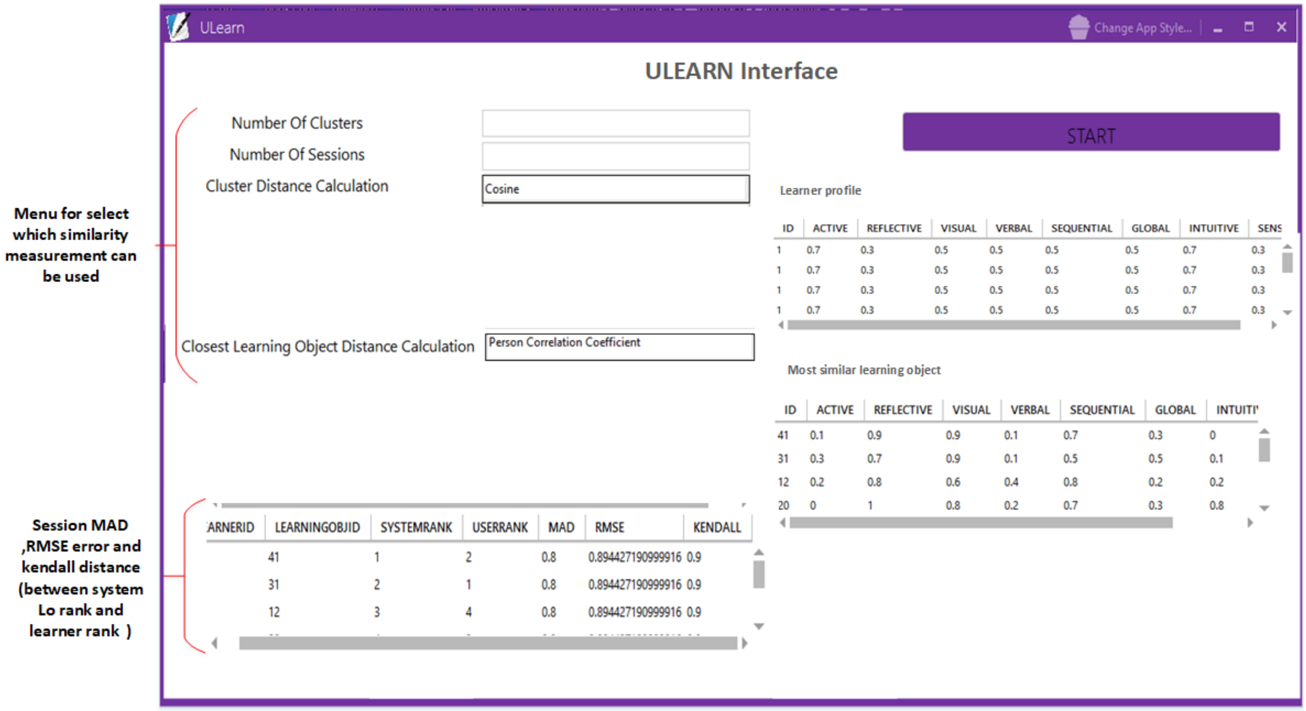
The final recommender algorithm is given by Algorithm 1 where the cosine similarity metric is used to determine the nearest learning object cluster to the active student learning style and the Pearson correlation coefficient is used to select the top-n learning objects of the nearest cluster that are the most similar to the active student learning style. It is understood that the function $quickSort(L, i)$ sorts a list of tuples L along the i^{th} dimension. In the case of Algorithm 1, L is a list of tuples of the form $< x, y >$, where x is a learning object profile (OP) and y the corresponding predicted rating for the active student LS.

5 Related Work

In recent years, recommendation technologies have been studied extensively by researchers across different disciplines. This section presents a review of some of the relevant works on e-learning recommendation systems approaches. A comparative review of these works is summarised in Table 5. In e-learning environment students are still being provided with static and predefined patterns of learning courses, tasks, materials, objects inspite of the fact that learner differ in characteristics such as learning interests, objectives, needs, skills and personalities. Taking all these differences into account it has become essential personalising learning items for individual learners using recommendation systems. However, most of these recommendation systems suffer from several drawbacks, e.g. some are based only on the users rating for generating the recommendations; others assume all on-line student would follow a consistent path, the path put out in the design and materialised by some hyperlinks.

6 Conclusion and Future Work

This paper presented a novel algorithm for recommending learning objects in e-learning systems based on the student learning styles. Initially a skeleton recommender algorithm was devised using two abstract similarity metrics Sim_1 and Sim_2 as place holders. Then an experimental study was carried out to find the best actual similarity metrics for



Menu for select which similarity measurement can be used

Number Of Clusters:
Number Of Sessions:
Cluster Distance Calculation: Cosine
Closest Learning Object Distance Calculation: Person Correlation Coefficient

Learner profile

ID	ACTIVE	REFLECTIVE	VISUAL	VERBAL	SEQUENTIAL	GLOBAL	INTUITIVE	SENS
1	0.7	0.3	0.5	0.5	0.5	0.5	0.7	0.3
1	0.7	0.3	0.5	0.5	0.5	0.5	0.7	0.3
1	0.7	0.3	0.5	0.5	0.5	0.5	0.7	0.3
1	0.7	0.3	0.5	0.5	0.5	0.5	0.7	0.3

Most similar learning object

ID	ACTIVE	REFLECTIVE	VISUAL	VERBAL	SEQUENTIAL	GLOBAL	INTUITI'
41	0.1	0.9	0.9	0.1	0.7	0.3	0
31	0.3	0.7	0.9	0.1	0.5	0.5	0.1
12	0.2	0.8	0.6	0.4	0.8	0.2	0.2
20	0	1	0.8	0.2	0.7	0.3	0.8

Session MAD ,RMSE error and kendall distance (between system Lo rank and learner rank)

ARNERID	LEARNINGOBJID	SYSTEMRANK	USERRANK	MAD	RMSE	KENDALL
	41	1	2	0.8	0.894427190999916	0.9
	31	2	1	0.8	0.894427190999916	0.9
	12	3	4	0.8	0.894427190999916	0.9

Fig. 5: ULearn interface

Algorithm 1: Recommender Algorithm

```
input  : $K$: Number of clusters, $K \geq 1$
         $C$: Set of K clusters $C_1, \ldots, C_k$
         $CC$: Set of the K clusters' centroids
         $LS$: Active student learning style
         $T$: Similarity degree threshold
output: $TopN$: List of recommended learning
         objects
$j = 1$
$x = c(LS, CC_1)$
/* $c$ is the cosine similarity */
for $i = 2$ to $K$ do
    if $c(LS, CC_i) > x$ then
        $x = c(LS, CC_i)$
        $j = i$
    end
end
/* $C_j$ is the nearest cluster to $LS$ */
$TopN = []$ /* empty list */
foreach $OP \in C_j$ do
    $S = P(LS, OP)$
    /* $P$ is the Pearson similarity */
    if $S \geq T$ then
        $TopN.append(< OP, int(0.5 + S \times 5) >)$
    end
end
$quickSort(TopN, 2)$
return $TopN$
```

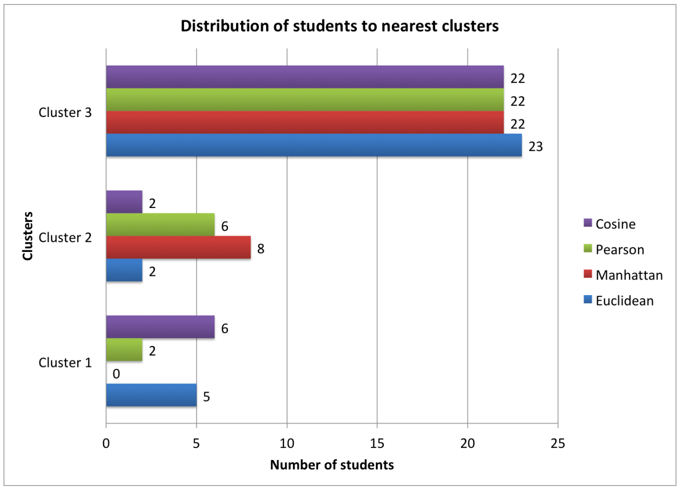


Fig. 6: Distribution of students to nearest clusters

Sim_1 and Sim_2 from the pool of the common similarity metrics used in recommender systems. After applying the K-means clustering algorithm to create K clusters of learning objects based on their profile, the similarity metric Sim_1 is used to find the nearest learning object cluster to the active student learning style. Then Sim_2 is used to determine the top-n learning objects in that cluster which are the most similar to the the active student learning style. Finally, the predicted rating of learning objects for the active student are calculated using Sim_2. It was found that the algorithm performs best for Sim_1 be the cosine similarity and Sim_2 the Pearson correlation coefficient.

In future work, the algorithm will be extended to improve the accuracy of recommendations by taking into account the ratings of learners who have similar learning

TABLE 5: A comparison between the different E-learning recommendation systems

		Input Data				
Refs.	Function	Users	Explicit	Implicit	techniques	Evaluation
[33]	Measure Similarity value between words	Learner Domain expert	✓	✓	N-Gram Query Expansion Ontology	-Keyword: 75.11% -N-Grams : 89.85% -Hybrid : 95.25%
[22]	Recommend LO based on Neighborhood learners	Learner		✓	neighborhood based on Euclidean distance	association rules are evaluated based on (support and confidence.)
[19]	Recommend learning materials	Learner	✓	✓	Adjusted Cosine Similarity ontology	proposed algorithm accuracy using the mean absolute error
[34]	Recommended LO sequence based on learning style	Learner	✓	✓	Clustering Machine learning	Learners in the experimental group complete a course in less time than learners in the control group
[35]	LOs recommendation	Learner	✓	✓	Nearest neighborhood Preference matrix Genetic Algorithm	Improve recommendation accuracy and alleviate cold-start and sparsity problems
[36]	LOs recommendation based on collaborative filtering and sequential.	Learner domain expert		✓	K-means Bi-Directional based frequent closed sequence mining	proposed model showed higher accurately according to the real-time up dated contextual information

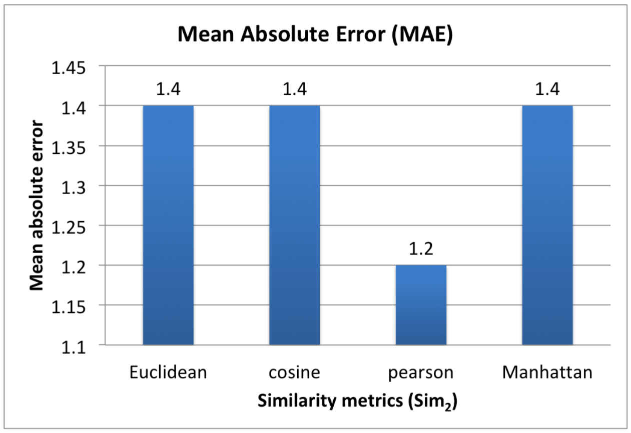


Fig. 7: Accuracy of the recommender algorithm using MAE

styles to the active learner.

References

[1] H. Drachsler, K. Verbert, O. C. Santos, and N. Manouselis, "Panorama of recommender systems to support learning," in *Recommender systems handbook*. Springer, 2015, pp. 421–451.

[2] J. Lu, D. Wu, M. Mao, W. Wang, and G. Zhang, "Recommender system application developments: a survey," *Decision Support Systems*, vol. 74, pp. 12–32, 2015.

[3] A. Bennane, "Adaptive educational software by applying reinforcement learning." *Informatics in Education*, vol. 12, no. 1, 2013.

[4] K. I. Ghauth and N. A. Abdullah, "Learning materials recommendation using good learners ratings and content-based filtering," *Educational technology research and development*, vol. 58, no. 6, pp. 711–727, 2010.

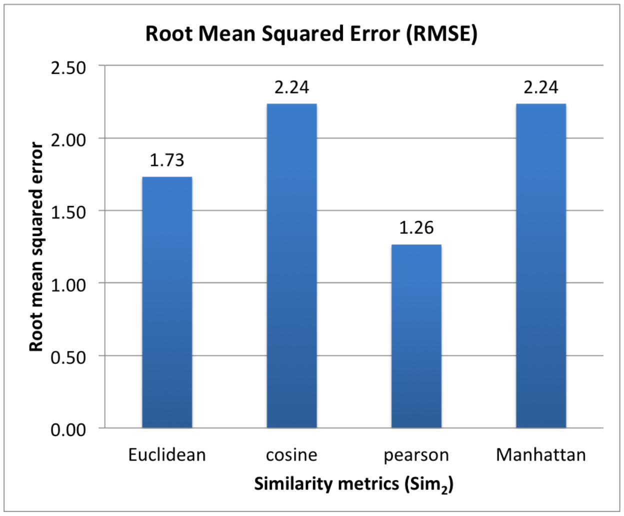


Fig. 8: Accuracy of the recommender algorithm using RMSE

[5] P. Vu, S. Fredrickson, and C. Moore, *Handbook of Research on Innovative Pedagogies and Technologies for Online Learning in Higher Education*. IGI Global, 2016.

[6] R. M. Felder and J. Spurlin, "Applications, reliability and validity of the index of learning styles," *International journal of engineering education*, vol. 21, no. 1, pp. 103–112, 2005.

[7] E. Popescu, "Adaptation provisioning with respect to learning styles in a web-based educational system: an experimental study," *Journal of Computer Assisted Learning*, vol. 26, no. 6, pp. 243–257, 2010.

[8] S. Graf, "Adaptivity in learning management systems focussing on learning styles," 2007.

[9] R. M. Felder, L. K. Silverman *et al.*, "Learning and teaching styles

in engineering education," *Engineering education*, vol. 78, no. 7, pp. 674–681, 1988.

[10] C. A. Carver, R. A. Howard, and W. D. Lane, "Enhancing student learning through hypermedia courseware and incorporation of student learning styles," *IEEE transactions on Education*, vol. 42, no. 1, pp. 33–38, 1999.

[11] J. Kuljis and F. Liu, "A comparison of learning style theories on the suitability for elearning," *Web Technologies, Applications, and Services*, vol. 2005, pp. 191–197, 2005.

[12] F. Dağ and A. Geçer, "Relations between online learning and learning styles," *Procedia-Social and Behavioral Sciences*, vol. 1, no. 1, pp. 862–871, 2009.

[13] P. García, A. Amandi, S. Schiaffino, and M. Campo, "Evaluating bayesian networks precision for detecting students learning styles," *Computers & Education*, vol. 49, no. 3, pp. 794–808, 2007.

[14] C.-C. Hsu, K.-T. Wang, and Y.-M. Huang, "Modeling personalized learning styles in a web-based learning system," in *Transactions on edutainment IV*. Springer, 2010, pp. 12–21.

[15] A. R. Anaya, M. Luque, and T. García-Saiz, "Recommender system in collaborative learning environment using an influence diagram," *Expert Systems with Applications*, vol. 40, no. 18, pp. 7193–7202, 2013.

[16] 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 *Information Technology Based Higher Education and Training (ITHET), 2016 15th International Conference on*. IEEE, 2016, pp. 1–5.

[17] M. Robillard, R. Walker, and T. Zimmermann, "Recommendation systems for software engineering," *IEEE software*, vol. 27, no. 4, pp. 80–86, 2010.

[18] S. B. Aher and L. Lobo, "Combination of machine learning algorithms for recommendation of courses in e-learning system based on historical data," *Knowledge-Based Systems*, vol. 51, pp. 1–14, 2013.

[19] J. Tarus, Z. Niu, and B. Khadidja, "E-learning recommender system based on collaborative filtering and ontology," *World Academy of Science, Engineering and Technology, International Journal of Computer, Electrical, Automation, Control and Information Engineering*, vol. 11, no. 2, pp. 256–261, 2017.

[20] M. Salehi, I. N. Kmalabadi, and M. B. G. Ghoushchi, "A new recommendation approach based on implicit attributes of learning material," *Ieri Procedia*, vol. 2, pp. 571–576, 2012.

[21] M. Salehi and I. N. Kmalabadi, "Attribute-based recommender system for learning resource by learner preference tree," in *Computer and Knowledge Engineering (ICCKE), 2012 2nd International eConference on*. IEEE, 2012, pp. 133–138.

[22] H. Imran, M. Belghis-Zadeh, T.-W. Chang, S. Graf *et al.*, "Plors: a personalized learning object recommender system," *Vietnam Journal of Computer Science*, vol. 3, no. 1, pp. 3–13, 2016.

[23] M. Ravichandran and G. Kulanthaivel, "A multivariate lms with–hybrid recommender system using association rules for peer learners," in *International Conference on Advanced Computer Technology (ICACT)*. IJCA, 2011.

[24] A. Al-Badarenah and J. Alsakran, "An automated recommender system for course selection," *International Journal of Advanced Computer Science and Applications*, vol. 7, no. 3, pp. 1166–1175, 2016.

[25] P.-E. Danielsson, "Euclidean distance mapping," *Computer Graphics and image processing*, vol. 14, no. 3, pp. 227–248, 1980.

[26] M. Ji, F. Xie, and Y. Ping, "A dynamic fuzzy cluster algorithm for time series," in *Abstract and Applied Analysis*, vol. 2013. Hindawi, 2013.

[27] J. Lee Rodgers and W. A. Nicewander, "Thirteen ways to look at the correlation coefficient," *The American Statistician*, vol. 42, no. 1, pp. 59–66, 1988.

[28] G. Salton and M. J. McGill, *Introduction to Modern Information Retrieval*. New York, NY, USA: McGraw-Hill, Inc., 1986.

[29] 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 Transactions on Pattern Analysis & Machine Intelligence*, no. 7, pp. 881–892, 2002.

[30] M. R. Felder and B. A Soloman, "Index of learning styles questionnaire," in *https://www.webtools.ncsu.edu/learningstyles/ (accessed 05 October 2018)*, 1999.

[31] S. Nafea, F. Siewe, and Y. He, "Ulearn:personalised learner's profile based on dynamic learning style questionnaire," in *IEEE Technically Sponsored Intelligent Systems Conference (IntelliSys)*, 2018.

[32] ——, "Ulearn: Personalized course learning objects based on hybrid recommendation approach," in *7th International Conference on Software and Information Engineering (ICSIE 2018)*, 2018.

[33] Z. Gulzar, A. A. Leema, and G. Deepak, "Pcrs: Personalized course recommender system based on hybrid approach," *Procedia Computer Science*, vol. 125, pp. 518–524, 2018.

[34] A. Klašnja-Milićević, B. Vesin, M. Ivanović, and Z. Budimac, "E-learning personalization based on hybrid recommendation strategy and learning style identification," *Computers & Education*, vol. 56, no. 3, pp. 885–899, 2011.

[35] M. Salehi and I. N. Kamalabadi, "Hybrid recommendation approach for learning material based on sequential pattern of the accessed material and the learners preference tree," *Knowledge-Based Systems*, vol. 48, pp. 57–69, 2013.

[36] M. Salehi, "Application of implicit and explicit attribute based collaborative filtering and bide for learning resource recommendation," *Data & Knowledge Engineering*, vol. 87, pp. 130–145, 2013.