Trace Learners Clustering to Improve Learning Object Recommendation in Online Education Platforms

Zriaa Rajae[1]
Informatics and Applications Laboratory (IA), Faculty of Sciences
Moulay Ismail University of Meknes
Morocco

Amali Said[2]
Informatics and Applications Laboratory (IA), FSJES
Moulay Ismail University of Meknes
Morocco

El Faddouli Nour-eddine[3]
RIME Team, MASI Laboratory
E3S Research Center, EMI
Mohammed V University
Morocco

***Abstract*—E-learning platforms propose pedagogical pathways where learners are invited to mobilize their autonomy to achieve the learning objectives. However, some learners face a set of cognitive barriers that require additional learning objects to progress in the course. A mediating recommendation system is one of the efficient solutions to reinforce the resilience of online platforms, while suggesting learning objects that will be interesting for them according to their needs. The objective of this contribution is to design a new mediator recommendation model for e-learning platforms to suggest learning objects to the learner based on collaborative filtering. To this end, the proposed system relies on the implicit behaviors estimation function as an underlying technique to convert tacit traces into explicit preferences allowing to compute the similarity between learners.**

Keywords—e-learning; recommendation system; learning objects; tacit behaviors

I. INTRODUCTION

E-learning has become, in recent years, the fundamental pillar of any educational system [1], as it allows everyone to learn easily, at any time, from any place and through any tool (laptops, smart phones, ...). In addition, several universities, institutes and schools have started using e-learning platforms to evolve their educational systems under any circumstances.

E-learning is considered as a process by which a set of educational activities and resources are delivered through digital devices to help learners achieve their learning objectives in the best possible conditions [2]. Moreover, these platforms are based on two fundamental aspects: the technological aspect (platform infrastructure) and the pedagogical aspect (learning content and its exploitation) [3], moreover, they are generally designed for heterogeneous learners with diverse pedagogical characteristics, including those related to experience levels, preferences, learning styles, etc. To this end, it is necessary to take full advantage of new technologies to improve the context of the pedagogical tool and also to adapt the learning strategies according to the learners' profiles.

Today, the application of recommender systems in e-learning has become an important field of research, as learning platforms have grown considerably, resulting in a massive increase in online digital resources. As a result, learners face great difficulty in choosing the most relevant and useful learning resources. Recommender systems are promising new technologies in online learning environments, as they can mitigate the problem of information overload [4-5], while highlighting what is most relevant and interesting based on the learner's profile. The recommendation of personalized learning resources is based on the different types of knowledge identified in the learner's profile, such as: preferences, learning styles and contextual information [6]. Several works have been proposed in the application of recommender systems, which help users to get the desired information through some filtering processes, such as: recommending movies on Netflix, videos on Youtube, articles on Amazon and courses on Byju and Gooru.

This paper presents a recommendation system based on Tacit Learner Preferences (TLP), supported by a new methodological approach for extracting their tacit preferences, in order to provide learners with learning resources that perfectly match their preferences without the tutor's intervention. The recommendation system is based on a learner model that gathers all personal information (age, education level, language, ...), pedagogical characteristics (learning styles, ...) and competency profiles (prerequisites, performance, expertise level, ...). All this information is extracted from the learner's interactions within the online platform. However, the problem is the difficulty of extracting tacit traces from the learning platform.

The objective of this paper is to convert the tacit traces into explicit ratings, in order to estimate the learner's preferences for a resource in the learner model of the recommender system.

The rest of the paper is organized as follows. In Section II, we highlight the main techniques proposed in the literature. In Section III, we represent related research work. In Section IV, we describe our proposed approach: Model for Converting Tacit Behaviors into Explicit Behaviors and in Section V, we conclude the paper and suggest possible future work.

II. TYPES FOR RECOMMENDER SYSTEMS

Recommender system is defined as a strategy that helps users make decisions in complex and evolving information spaces [7-8]. The recommender system suggests items for the user to evaluate based on their profile and the target domain. In addition, recommender systems address the problem of information overload and lack of domain knowledge, which users usually encounter, by providing personalized and exclusive content and service recommendations [9]. Recommender systems are classified into three main categories namely: Content-Based Filtering (CBF), Collaborative Filtering (CF), Hybrid Filtering (HF).

A. Content-Based Filtering

Content-based filtering (CBF) is used to suggest articles similar to those previously liked by the user [10]. For example, the recommendation of research articles is based on the content-based approach, where the procedure uses the descriptive content of the articles and the users' needs without considering the ratings of other users [11]. The main problem with the content-based filtering recommender method is serendipity

1) This problem of serendipity is more known in content-based recommender systems, as these systems only recommend items that match the user's profile. At this point, the user has no chance to receive unexpected recommendations, which leads to a certain weariness with the proposed recommendations.

B. Collaborative Filtering

Collaborative filtering (CF) is the most common and effective technique in recommender systems, as it compares users' ratings with other users' ratings to find the users who are "most similar" based on a similarity criterion and to recommend the articles that similar users have previously liked [12]. In 2016, QusaiShambour et al. developed a personalized recommendation system based on collaborative multi-criteria filtering of articles, while exploiting multi-criteria ratings and semantic information of articles, to overcome the problem of data sparsity and cold start of articles [13]. Collaborative filtering is considered the most popular and widespread method in recommender systems. They have been massively exploited in companies and universities. Some of these systems include PHOAKS [14], which helps users find accurate and relevant information on the web, GroupLens [15], BellcoreVideoRecommender [16], etc.

The main problem with the collaborative filtering recommendation method is data sparsity and the cold start problem:

1) The cold start problem [17-18]: is caused by the lack of data on new items or new users. Indeed, a new item cannot be recommended until a user has evaluated it. Similarly, for a new user, we cannot predict his preferences without knowing his item evaluation history.

2) The sparsity problem [19-20-21]: is generated when the number of items rated by users is very small compared to the total number of items available in the system. Parity results in a very low density of the matrix (items/user). This affects the ability of the system to recommend less accurate items.

C. Hybrid Filtering

Hybrid filtering (HF): aims at combining the strengths of the previously explained recommendation approaches in order to benefit from their complementary advantages and to overcome the problems identified before. Several techniques have been proposed to combine the basic techniques to create a new hybrid system. In 2018, R. Shanthi and colleagues proposed a hybrid recommender system to recommend products to users based on users' opinions and ratings [22]. In 2002, Burke describes a taxonomy that proposes seven ways of hybridization: weighted, switching, mixed, feature combination, cascade, feature augmentation, and meta-level [23].

III. Related Work

Several recommender systems have been developed for online learning, such as: In 2018, Feng Zhang et al. proposed a recommender system based on the collaborative filtering approach to recommend learning resources that are valued by learners most similar to the active learner [24]. In 2018, Hayder Murad et al. designed a recommender system that detects students' profiles and knowledge levels, with the aim of automatically recommending online video learning materials that are perfectly suited to students' needs [25]. In 2017, Tarus et al. propose a hybrid ontology-based recommender system with sequential pattern mining to recommend online learning resources to learners [26]. In 2015, Bokde et al. develop an academic recommender system, which provides engineering school students with recommendations that meet their past preferences, based on a hybrid technique that combines article-based multi-criteria collaborative filtering with a dimensionality reduction approach [27]. In addition, a tutoring system based on a recommendation engine Protus (ProgrammmingTUtorting System) [28] was designed to recommend materials of interest to learners, while taking into account their pedagogical differences such as: preferences, knowledge, learning goals and learner progress, etc. The initial recommendation in Portus is based on the default sequence of lessons and the surveys previously assigned to the lessons.

Most of the works proposed in e-learning context are based on collaborative filtering, with the aim of recommending educational resources based on the profile and explicit evaluations of similar learners. These recommendation systems are designed to provide a personalized list of suggestions (educational resources, learning activities or videos). However, these systems ignore the importance of assisting learners in their learning journey, through the recommendation of additional resources to help them overcome cognitive difficulties and also to maintain their perseverance throughout the online training. Moreover, these recommender systems only exploit explicit user ratings to make recommendations.

Explicit evaluations are sometimes poorly expressed or ignored by users, which leads to a decrease in the performance of recommender systems [29]. In other words, implicit user feedback is poorly considered in most existing recommender systems compared to explicit feedbacks, knowing that implicit behaviors can serve as a means to overcome several problems that plague recommender systems, such as: data sparsity and cold start.

Moreover, to improve the accuracy of recommendations, several works have been done to better capture and know users' preferences based on their behaviors in the system; In 2009, Rendle et al. proposed an article recommendation system based on users' implicit comments to predict a personalized ranking on a set of articles [30]. In 2008, Hu et al. developed a recommender system based mainly on implicit user interactions which are considered as user preferences, the proposed approach is based on converting expressed trust into user preference [31]. In 2006, Zigoris et al. proposed a Bayesian model technique to combine implicit and explicit user behaviors [32]. In 2005, Adomavicius et al. focused on exploiting implicit user interactions to make relevant recommendations [33].

Designing a recommender model based on a preference matrix that generates the different implicit interactions and explicit evaluations of learners is one of the challenges. Indeed, implicit feedbacks represent an objectivity towards a learning resource, while explicit feedbacks indicate a subjectivity, through the learners' rating of the learning objects. This combination allows for a more personalized learning environment that is more appropriate to the learners' expectations.

IV. Proposed Model

The model proposed illustrates the general recommendation process that is based on the tacit behaviors (objective preferences) and explicit evaluations (subjective preferences) of learners in e-learning, in order to build a preference matrix that represents the set of learning objects with their ratings. In addition, this model also allows to suggest learning objects that help learners to overcome cognitive obstacles and to progress easily in their training path, without any personal experience required in the search of alternatives (see Fig. 1).

First, the implicit user data is extracted from the log files. Then, this data is processed in such a way that it is combined with the explicit evaluations stored in the database of the LMS used. Subsequently, the prepared data are used in a learner preference matrix for learning objects "LPLO" that combines objective and subjective learner preferences. Then, a classification of learners into virtual communities of similar interest is done, using the clustering approach. Finally, the recommended learning objects are presented to the learner in order to help him in his learning path. In this way, the learner can interact with the recommended learning objects and start to exploit the ones that match his cognitive level and preferences.

A. Model of Conversion from Tacit to Explicit Behaviors

Our proposal is based on the exploration of the log files and the database of the used LMS, which record the learners' traces during their interactions with the e-learning system. Furthermore, the preference matrix is generated based on these interactions, in order to select the most relevant learning objects. Tacit behavioral indicators and explicit ratings are used to build a model of the learner in an e-learning system by analyzing the learner's interactions with the learning objects implicitly or explicitly.

In order to maintain interoperability of CEHL, a set of standards have been considered in the literature to model learners [34] such as:

- The PAPI Learner Model (Public and Private Information for Learner) is a standard developed by the IEEE P1484.2 Learner Model Working Group and is one of the first proposals to model the learner. Various information is presented in this model: personal or demographic information, relational information, information on preferences, information on the learner's history and learning progress, etc.
- The IMS-LIP Learner Model ((IMS Learner Information Package), is a standard that allows the necessary characteristics of a learner to be modeled using XML technology in order to ensure interoperability between CEHL. The characteristics presented in this model are the learner's personal information, acquired skills, information related to qualifications, information about the learner's preferences and interests, etc.

Our system is based on the PAPI model for modeling the learner in the learning system, because, the objective of this contribution is to collect and exploit the information about the learner's history in the recommendation process (see Fig. 2). Thus, we used the demographic information as initial information to start the learner profiling process, and the behavioral information to describe the objective preferences in his profile [35].

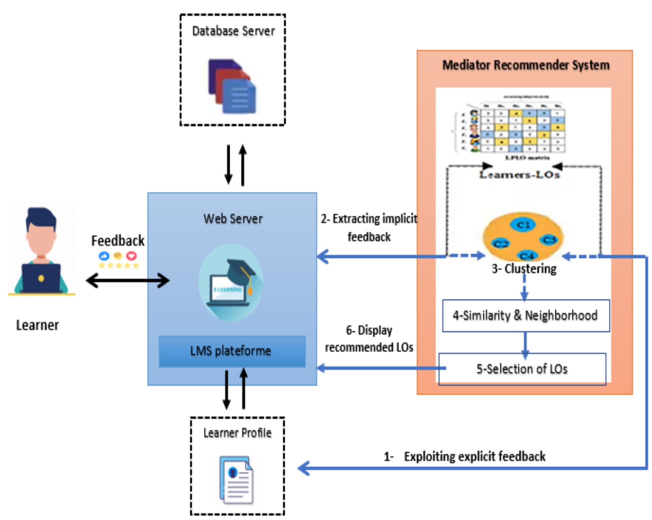

Fig. 1. Recommender System Model.

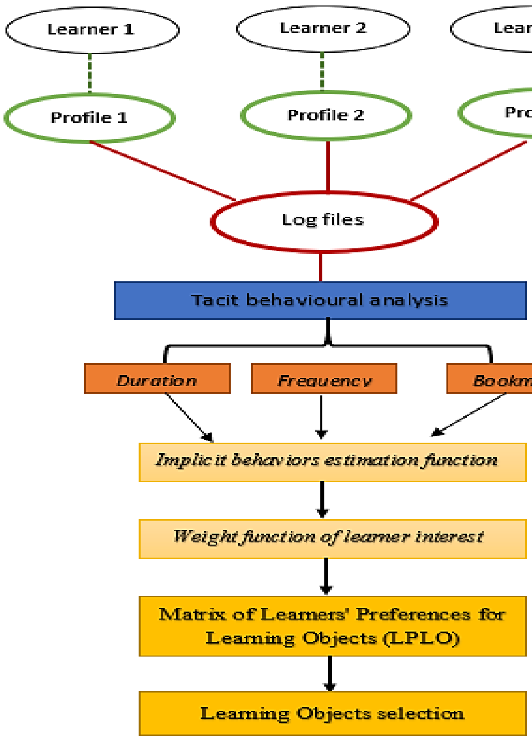

Fig. 2. Behavioural Analysis for LOs Selection.

1) Learner demographic information: name, gender, age, nationality, languages, education level, etc. This type of information is used during the learner's first interactions with the system, in order to overcome the cold start problem. Demographic filtering is used to begin the process of building the learner's profile.

2) Behavioral information: duration of use of a learning object, frequency of access to a learning object, learning object marked as important to the learner. This type of information is used to transform tacit behaviors into explicit preferences.

3) Estimation function for implicit behaviors: The learner's browsing history is recorded in a log file, hence the need to prepare the log file (their extraction and analysis), in order to generate a scoring matrix based on the following indicators:

- Duration (D): indicates the time a learner spends when operating a learning object.
- Frequency (F): indicates how often the learner requests the learning object.
- Bookmark(s): indicates the learning objects that are marked by the learner as important.

The implicit interest estimation function (1) allows to define the implicit score for each learning object, visited or consulted by the learner through the use of the indicators mentioned above; In this respect, we have relied on the "Page InterestEstimator" formula, to compute the implicit scores defined by Philip K. Chan in 2003, in an e-learning context, which uses the user traces to define the interest of a user for an article (web page) [36-37]. In our context, the implicit interest estimation function is defined as follows:

$$Imp(l_{a,} lo_k) = D(l_{a,} lo_k) + F(l_{a,} lo_k) + B(l_{a,} lo_k) \qquad (1)$$

Where**:**

l_a is active learner, lo_k is a learning Object, $D(l_a,lo_k)$ indicate the duration, F (l_a,lo_k) the frequency and B (l_a,lo_k) denotes the bookmarks.

4) Learner interest weighting function: The learner interest weighting function (Wli) (2) is used to combine tacit behavioral indicators and variant ratings between 1 and 5 (see Table I).

TABLE I. THE LEARNING OBJECT RATING SCALE

Linking	**Rating scale**
Very like	5
Like	4
Normal	3
Not like	2
Do not like	1

On the other hand, the value 0 indicates that the learning object is not evaluated by the learner.

The learner interest weights function (2) is defined as follows:

$$Wli(l_{a,} lo_k) = 1/2\ (Imp(l_{a,} lo_k) + Exp(l_{a,} lo_k)) \qquad (2)$$

Where:

Exp(l_a ,lo_k)) l_a is the current learner and lo_k is the learner's explicit score, where k is the number of learning object, the rating is normalized according to the scale [1-5], in case the lo_k is no longer rated by the learner, the value assigned to the Exp(l_a ,lo_k) function is '0'.

5) Matrix of learner preferences for learning objects (LPLO): After the normalization of ratings, the basic matrix LRLO (Learners' Ratings for Learning Objects) contains the explicit ratings of the learners, this matrix is transformed into matrix LPLO (Learners' Preferences for Learning Objects) contains the implicit and explicit preferences of the learners, where the rows represent the learners L {l1,l2,l3,...}. And the columns represent the learning objects LO {lo1,lo2,lo3,...}. Moreover, the unknown notations are defined by the function $Wli(l_{a,} lo_k)$, which is computed based on the implicit interest estimation function (1), with the aim of defining the implicit score for each learning object, visited or accessed by the learner through the exploitation of the following indicators: Duration, frequency and bookmark (see Fig. 3).

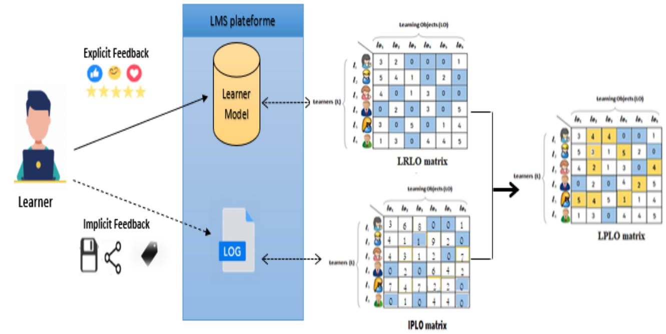

Fig. 3. Matrix Transformations.

The LPLO matrix is less sparse compared to the basic LRLO matrix , the saprsity level (3) is calculated as the ratio of the number of unevaluated learning objects (empty entries) to the total number of learning objects in the matrix (LxLO) (Matrix Size).

$$levelofsparsity = \frac{UnratedLOsnumbers}{MatrixSize} \quad (3)$$

The number of missing ratings for the LRLO matrix is 14, while for the LPLO matrix is 7, moreover the size of both matrices is 36. The level of sparsity for the LRLO matrix is 39%, however for the LRLO matrix is 19%. Also, when the matrix is less sparsity data, i.e. the number of missing ratings is low; the recommendations will be more accurate.

6) Preference matrix weighted by educational criteria; Learning objects are tagged with a set of pedagogical criteria, which are defined by the course designer. These criteria are transcribed by values and are used to personalize the recommendations according to the learner's level of involvement in the learning system and his preferences expressed in his profile.

The Table II illustrates the different educational criteria with their values.

So, for each learner 'cl', there exists in his profile a set of learning objects $LO_{cl}=\{lo^{cl}_i \in LO, i=1\ldots m_{cl}\}$, where lo^{cl}_i are the learning objects consulted by the current learner 'cl', the values of the pedagogical criteria are described by v^{cl}_i and the notations generated on the basis of the function Wli (2) for the pair {learner, learning object} are expressed by r^{cl}_i. (see Table III).

B. Classification of Learners into Clusters

1) Pearson correlation coefficient (PCC): is one of several measures such as: Manhattan distance, Jaccard similarity, Cosine similarity and Euclidean distance [38], all these measures are used to calculate the degree of similarity between learners based on a set of criteria, where similar learners are assigned to the same cluster; each of these measures has been applied to millions of clustering applications in the measure of creating virtual communalities of similar interest.

TABLE II. EDUCATIONAL CRITERIA AND THEIR ASSOCIATED VALUES

Function	***Criteria***	***Values***
Content f(c)	***Duration (D)***	*5,10, 15, … min*
	Difficulty level (DL)	
	LOT(Lower Order Thinking)	*1*
	MOT (Middle Order Thinking)	*2*
	HOT (Higher Order Thinking)	*3*
	Objective Level (OL)	
	Remember	*1*
	Comprehension	*2*
	Application	*3*
	Analysis	*4*
	Evaluation	*5*
Presentation f(p)	***Preferences(P)***	
	Theory	*1*
	Exercises	*2*
	Examples	*3*
	Real-life applications/ Simulation	*4*
	case study	*5*
	Demonstration	*6*
	Assessment tests	*7*
Media f(m)	***Learning material (LM)***	
	Text	*1*
	Image	*2*
	Audio	*3*
	Video	*4*

TABLE III. LEARNING OBJECTS-PEDAGOGICAL CRITERIA-RATING MATRIX BY LEARNER

	c_1	c_2	c_3	..	..	c_p	r_{cl}
lo_1	v^1_1	v^1_2	v^1_3			v^1_p	r^{cl}_1
lo_2	v^2_1	v^2_2	v^2_3			v^2_p	r^{cl}_2
lo_3	v^3_1	v^3_2	v^3_3			v^3_p	r^{cl}_3
lo_4	v^4_1	v^4_2	v^4_3			v^4_p	r^{cl}_4
..	..	..	..			..	..
..	..	..	..			..	..
lo_m	v^m_1	v^m_2	v^m_3			v^m_p	r^{cl}_m

In our context, we use the Pearson correlation; given that the variables of the matrix are associated in a linear way, i.e. when a change is made to one variable, a proportional change is made to the other variable. Moreover, since the values of the variables are quantitative and have a Gaussian distribution, the use of the Pearson correlation is adequate to define the degree of similarity between the learners. On the other hand, the De Jaccard similarity coefficient is sufficiently appropriate for use in documents or word similarity measurement.

Pearson Correlation Coefficient (PCC) known as Pearson Product-Moment Correlation Coefficient "r", PCC is one of the most popular coefficients for measuring the dependence of two variables, i.e. the relationship between two quantitative variables and the degree of similarity between these two variables.

- Dependency Coefficient (DC): is calculated through the Pearson Correlation Coefficient in order to identify the degree of association between the ratings provided by the learner for all the learning objects consulted.
- Pearson Correlation Coefficient (PCC) (8): is a statistical measure of the linear relationship between two variables and it ranges between [-1,+1]; the positive correlation indicates that the variables increase or decrease in parallel, however a negative correlation means that one variable increases while the other decreases. Furthermore, correlation (4) is a measure of effect where the strength of the correlation is described as follows, with -1.0≤ r ≤+1.0 (see Table IV).

TABLE IV. INTERPRETATION OF R VALUES

r value	*Interprétation*
+.80 to +1.0	Very strong positive relationship
+.60 to +.79	Strong positive relationship
+.40 to +.59	Moderate positive relationship
+.20 to +.39	Weak positive relationship
+.01 to+.19	Negligible relationship
0	No relationship
-.01 to -.19	Negligible relationship
-.20 to -.39	Weak negative relationship
-.40 to -.59	Moderate negative relationship
-.60 to -.79	Strong negative relationship
-.80 to -1.0	Very strong negative relationship

The correlation value 'r' is calculated as follows:

$$r_{x,y} = \frac{\delta xy}{\delta x \ \delta y} \quad (4)$$

Where:

$$\delta xy = \frac{1}{n}\sum_{i=1}^{n} x_i y_i + \bar{x}\bar{y} \quad (5)$$

$$\delta x = \sqrt{\sum_{i=1}^{n} \frac{x_i^2}{n} - \bar{x}^2} \quad (6)$$

$$\delta y = \sqrt{\sum_{i=1}^{n} \frac{y_i^2}{n} - \bar{y}^2} \quad (7)$$

Based on the formula proposed by Manuel J. Barranco and Luis Martinez, the Pearson correlation coefficient is associated with two variables (see Table V):

- r_{cl} : The ratings assigned by the current learner 'cl' for the learning objects.
- v_{cl} : The pedagogical criteria of the learning objects consulted by the current learner.

$$PCC_{\ cl\ j} = \frac{\sum_i r_i^{cl} v_{ij}^{cl} - \frac{\sum_i r_i^{cl} \sum_i v_{ij}^{cl}}{n_{cl}}}{\sqrt{\left(\sum_i \left(r_i^{cl}\right)^2 - \frac{\left(\sum_i r_i^{cl}\right)^2}{n_{cl}}\right)}\sqrt{\left(\sum_i \left(v_{ij}^{cl}\right)^2 - \frac{\left(\sum_i v_{ij}^{cl}\right)^2}{n_{cl}}\right)}} \quad (8)$$

TABLE V. DETAILS OF NOTATIONS USED IN PCC METRIC

Notation	*Description*
r_i^{cl}	Score assigned by the learner 'cl' for the learning object ‘lo_i’
v_{ij}^{cl}	Pedagogical criterion 'vj' for the learning object ' lo_i' consulted by the learner 'cl'
n_{cl}	Number of learning objects consulted by the current learner 'cl'

7) K-means clustering: The main goal of clustering is to divide learners into groups based on similarity characteristics with the aim of recommending learning objects that the active learner has never visited [39]. In a first step, learners with similar interests will be grouped in the same cluster, in order to recommend them appropriate learning objects, for this purpose, we opt for the K-means algorithm which refers to the preference matrix "LPLO" (learner-learning object) generated in the previous step.

The K-means algorithm is well known for its efficiency and power in clustering a large data set compared to other k-nearest neighbors’ algorithm [40]. Moreover, it is considered one of the most popular clustering algorithms for unsupervised learning.

The procedure for assigning learners into clusters via the k-means algorithm is done according to the following pseudo code:

Input: k // Number of desired clusters
L= {l_1,l_2,l_3,……...L_n) // Set of learners

Output: a set of k clusters

Process:

Arbitrarily select k learners form L as the initial cluster centers;

Repeat:

1-(re) assign each learner to the clusters with the most similar interests based on the Pearson correlation coefficient 'PCC' of the current learner and the mean value of the learners in the cluster
2-Update the cluster means; calculate new mean value of learners for each cluster;

Until no change;

8) Selecting learning objects: The learning objects recommended to the current learner is made, by the association between the profile of the current learner and the learning objects repository. Moreover, the learning objects that are well evaluated by the closest neighbors’ to the active learner and having a score higher than three will be recommended to the current user (see Fig. 4).

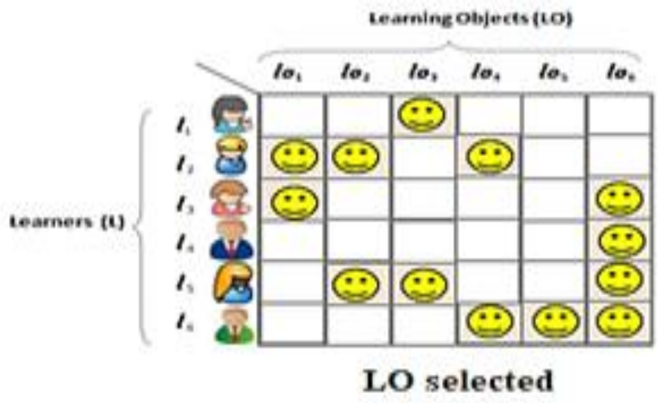

Fig. 4. The Selected Learning Objects.

Rate [lo20] = 4 ✓ Rate [lo6] = 1

Rate [lo12] = 2 Rate [lo15] = 5 ✓

Rate [lo29] = 6 ✓ Rate [lo80] = 2

V. Experimentation

This section presents a performance analysis of the proposed methodology. The performance of this work is evaluated against the recommendation based only on the explicit preferences of the users.

The participants in this experiment are 100 learners from a high school in the delegation of Chefchaouen, Morocco. However, the learners had to study four modules in the computer science subject, namely: "Generalities of Computer Systems", "Software", "Algorithms and Programming" and "Networks and the Internet"; each module consists of a set of lessons, which are well defined in the pedagogical guidelines of computer science in high school. Our first experiment is based on the first module "Generalities of Computer Systems", which contains three lessons: lesson 1 "Basic Definitions and Vocabulary", lesson 2 "Basic Structure of a Computer" and lesson 3 "Software and Application Areas of Computer Science" (see Table VI).

The Table VII expresses the degree of depth of the concepts for each notion.

Our approach is tested on learning objects stored in the LMS database. The first module "General Computer Systems" provides an assessment dataset of 50 learning objects containing 36 ratings from 26 learners. The ratings are integers ranging from 1 to 5. The experiments are performed on an HP computer with CORE i5 processors.

The Sum Squared Error (SSE) (8) is used to find an appropriate k by plotting the number of clusters against the SSE, while evaluating SSE for different values of k.

$$SSE = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \quad (9)$$

Where:

i is test set, y_i is predicted value and $\hat{y}_i$ is actual value.

TABLE VI. Module No. 1: Generalities On Computer Systems

		Common Core			
Content	**Schedule**	**Letter & Arts**	**Original**	**Science**	**Technologies**
Definition and basic Vocabulary	***2h***				
Definition of information		**2**	**2**	**2**	**2**
Definition of treatment		**2**	**2**	**2**	**2**
Definition of computer Science		**2**	**2**	**2**	**2**
Definition of the computer system		**2**	**2**	**2**	**2**
Basic structure of a computer	***4h***				
Functional diagram of a Computer		**2**	**2**	**3**	**3**
Peripherals		**2**	**2**	**3**	**3**
Central processing unit		**2**	**2**	**3**	**3**
Types of software	***1h***				
Basic software		**2**	**2**	**2**	**2**
Application software		**2**	**2**	**2**	**2**
Fields of application	***1h***	***2***	***2***	***2***	***2***

TABLE VII. The Degree of Depth

Degree of depth	**Descriptor**
1	Initiation
2	Appropriation
3	Master

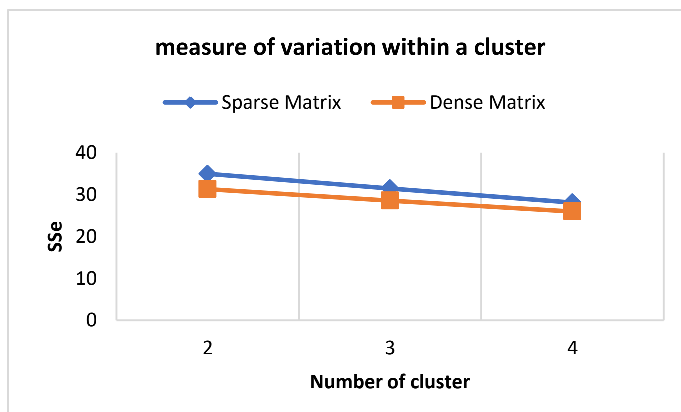

Fig. 5. The Appropriate Number of Clusters in respect to the Matrices.

From the above graph (see Fig. 5), we notice that the SSE value is high in the sparse matrix unlike the dense matrix. For both matrices, when the number of cluster increases, the SES value decreases. This means that the number of clusters is independent to the type of matrix. For this distribution of 100 learners, the most appropriate number of clusters for both matrix types (sparse or dense) is 4 clusters.

The silhouette is used to determine the degree of separation between clusters. The silhouette plot in (Fig. 6), illustrates the proximity of learners in neighboring clusters using a measure of [-1 , +1]. A value of +1 indicates that the learners are far apart, 0 indicates that the observations are very close, and -1 indicates that the learners can be assigned to the wrong cluster.

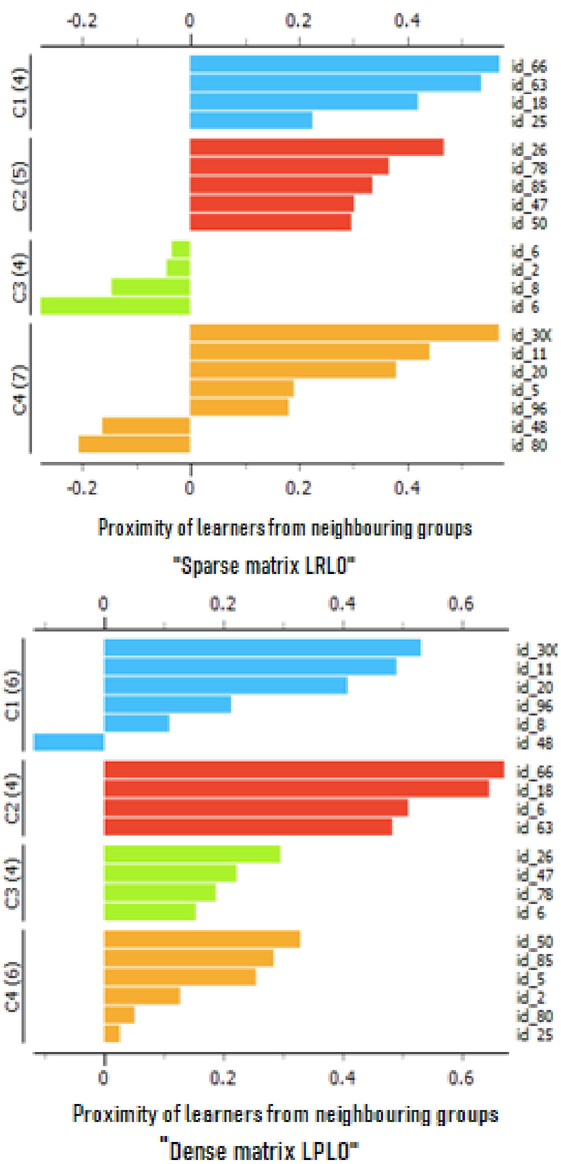

Fig. 6. Silhouette Analysis for Sparse and Dense Matrix.

For the sparse matrix LRLO "Learners Ratings for Learning Objects", we find that some learners are assigned to the wrong cluster, due to lack of ratings of learning objects in this Matrix. Since, some learners do not rate all these objects, their assignments to some clusters may be inappropriate. However, for the dense matrix LPLO "Learners Preferences for Learning Objects", the majority of learners are classified in the appropriate clusters, thanks to the elicitation of preferences through the combination of explicit ratings and implicit interactions of learners with the learning objects.

The mean absolute error (MAE) "(10)" is a measure of prediction accuracy. It indicates the absolute value of the difference between the predicted value and the actual value.

This measure is used to indicate the effectiveness of our recommendation system based on implicit and explicit learner feedback and recommendation systems based on explicit feedback only. The lower the MAE value, the smaller the magnitude of the error.

$$MAE = \frac{\sum_{(u,i)\in test} |prediction_{u,i} - real_{u,i}|}{n_{test}} \quad (10)$$

Where:

n is the total number of ratings-prediction pairs in the test set, $prediction_{u,i}$ is the predicted rating for learner u on learning object j, and $real_{u,i}$ is the actual rating in the real dataset (see Table VIII).

In the graph (see Fig. 7), we see that with the variation of the number of clusters, the magnitude of the error is gradually decreased for the dense matrix "LPLO", which is based on the implicit and explicit preferences of the learners; contrary to the LRLO matrix where the magnitude of the error gradually increases. This is due to the insufficient ratings explicitly expressed by the learner, which leads to a loss of accuracy when assigning learners to clusters. This is also shown in Fig. 6, where we can infer that the less sparse the matrix is, the better the learners are assigned to clusters with similar learning needs. Therefore, the recommendations will be more appropriate.

TABLE VIII. MAE VALUE ACCORDING TO THE TYPE OF MATRIX

Number of cluster	**Mean Absolute Error**	
	Dense Matrix LPLO	Sparse Matrix LRLO
2	0.18386	1.7154
3	0.08716	2.4421
4	0.06501	2.6225

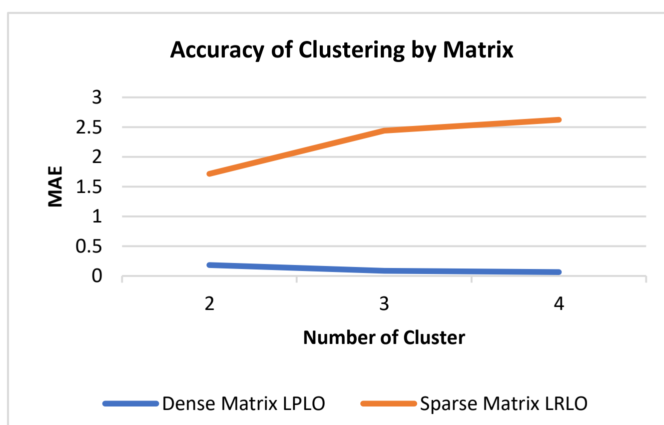

Fig. 7. The Accuracy of Clustering the Learners by Matrix.

VI. Conclusion and Future Work

Recommender systems have become a promising solution for improving the efficiency of e-learning systems. Preference matrices are the basic inputs of recommender systems, because based on these matrices, systems can suggest personalized learning objects and adapt them to the learner's profile. On the other hand, the lack of ratings in the preference matrices can handicap their functioning and consequently the recommendations will be less personalized.

To this end, we have proposed a model of a mediating recommendation system based on a matrix that combines the tacit and explicit behaviors of the learner. Through the exploration of the log file, while exploiting some indicators such as duration, frequency and bookmarks in the preference matrix, in order to obtain better performances.

The k-means clustering algorithm is used to group learners with similar preferences into clusters, in order to recommend to the active learner new learning objects that have not yet been viewed or visited by him and that have been well rated by his closest neighbors.

The results obtained in the experimentation phase illustrate the importance of hybridizing implicit traces and explicit ratings of learners to improve the accuracy of assignment of learners to appropriate clusters.

In our future work, we plan to study the recommendation of sequences of objects and learning activities, based on the dynamic prediction of the learning strategy adapted to the active learner, through evolutionary meta-heuristic algorithms such as: Ant Colony Optimization (ACO) and Particle Swarm Optimization (PSO). This will allow recommending the learner a personalized learning path (scenario).

References

[1] Bezovski, Z., & Poorani, S. (2016, March). The evolution of e-learning and new trends. In Information and Knowledge Management (Vol. 6, No. 3, pp. 50-57). IISTE.

[2] Clark, R. C., & Mayer, R. E. (2016). E-learning and the science of instruction: Proven guidelines for consumers and designers of multimedia learning. john Wiley & sons.

[3] Devedžic, V. (2006). Semantic web and education (Vol. 12). Springer Science & Business Media.

[4] Maes, P. (1995). Agents that reduce work and information overload. In Readings in human–computer interaction (pp. 811-821). Morgan Kaufmann.

[5] Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. (2000, October). Analysis of recommendation algorithms for e-commerce. In Proceedings of the 2nd ACM Conference on Electronic Commerce (pp. 158-167).

[6] Joshi, N., & Gupta, R. May (2020). "A personalized web based e-learning recommendation system to enhance and user learning experience", International Journal of Recent Technology and Engineering (IJRTE) ISSN 2277-3878, 9(1),1186-1195.

[7] Rashid, A. M., Albert, I., Cosley, D., Lam, S. K., McNee, S. M., Konstan, J. A., & Riedl, J. (2002, January). Getting to know you: learning new user preferences in recommender systems. In Proceedings of the 7th international conference on Intelligent user interfaces (pp. 127-134).

[8] Nafea, S. M., Siewe, F., & He, Y. (2019). On Recommendation of Learning Objects using Felder-Silverman Learning Style Model. IEEE Access, 7, 163034-163048.

[9] Lalitha, T. B., & Sreeja, P. S. (2020). Personalised Self-Directed Learning Recommendation System. Procedia Computer Science, 171, 583-592.

[10] Javed, U., Shaukat, K., Hameed, I. A., Iqbal, F., Alam, T. M., & Luo, S. (2021). A Review of Content-Based and Context-Based Recommendation Systems. International Journal of Emerging Technologies in Learning (iJET), 16(3), 274-306.

[11] Philip, S., Shola, P., &Ovye, A. (2014). Application of content-based approach in research paper recommendation system for a digital library. International Journal of Advanced Computer Science and Applications, 5(10).

[12] Shi, Y., & Yang, X. (2020). A Personalized Matching System for Management Teaching Resources Based on Collaborative Filtering Algorithm. International Journal of Emerging Technologies in Learning (iJET), 15(13), 207-220.

[13] Shambour, Q., Hourani, M., & Fraihat, S. (2016). An item-based multi-criteria collaborative filtering algorithm for personalized recommender systems. International Journal of Advanced Computer Science and Applications, 7(8), 274-279.

[14] Felfernig A., Friedrich G., Jannach D., and Zanker M.,(2006) An Integrated Environment for the Development of Knowledge-Based Recommender Applications, International Journal of Electronic Commerce,, 11(2), 11-34.

[15] Resnick P., Iacovou N., Suchak M., Bergstrom P., and Riedl J., (1994), GroupLens: An Open Architecture for Collaborative Filtering of Netnews, In Proceedings of ACM Conference on Computer Supported Cooperative Work, Chapel Hill, NC, USA, 175-186.

[16] Shih Y. and Liu D.,2005, Hybrid Recommendation Approaches: Collaborative Filtering via Valuable Content Information, In the Proceedings of the 38th Annual Hawaii International Conference HICSS '05, 217-223.

[17] Adomavicius G, Tuzhilin A, 2005 June, Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. Knowledge and Data Engineering, IEEE Transactions on.;17(6):734–749.

[18] Schein AI, Popescul A, Ungar LH, et al. 2002, Methods and metrics for cold-start recommendations. In: SIGIR '02: Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval. New York, NY, USA: ACM;. p. 253–260.

[19] Da Silva, J. F. G., de Moura Junior, N. N., &Caloba, L. P. (2018, July). Effects of data sparsity on recommender systems based on collaborative filtering. In 2018 International Joint Conference on Neural Networks (IJCNN) (pp. 1-8). IEEE.

[20] F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, 2010, Recommender Systems Handbook. Springer.

[21] Arwar, B., Karypis, G., Konstan, J., & Riedl, J. (2000, October). Analysis of recommendation algorithms for e-commerce. In Proceedings of the 2nd ACM Conference on Electronic Commerce (pp. 158-167).

[22] Shanthi, R., & Rajagopalan, S. P. (2018). A Personalized Hybrid Recommendation Procedure for Internet Shopping Support. International Journal Of Advanced Computer Science And Applications, 9(12), 363-372.

[23] Burke, R. (2002). Hybrid recommender systems: Survey and experiments. User modeling and user-adapted interaction, 12(4), 331-370.

[24] Feng Zhang, Victor E. Lee, Ruoming Jin, Saurabh Garg, Chi Cheng, Privacy-aware smart city: A case study in collaborative, 2018.filtering recommender systems, in: Journal of Parallel and Distributed Computing.

[25] Murad, H., & Yang, L. (2018). Personalized e-learning recommender system using multimedia data. International Journal of Advanced Computer Science and Applications, 9(9), 565-567.

[26] Tarus, J. K., Niu, Z., & Yousif, A. (2017). A hybrid knowledge-based recommender system for e-learning based on ontology and sequential pattern mining. Future Generation Computer Systems, 72, 37-48.

[27] Bokde, D. K., Girase, S., & Mukhopadhyay, D. (2015, December). An approach to a university recommendation by multi-criteria collaborative filtering and dimensionality reduction techniques. In 2015 IEEE International Symposium on Nanoelectronic and Information Systems (pp. 231-236). IEEE.

[28] Vesin, B., Ivanović, M., Klašnja-Milićević, A., & Budimac, Z. (2012).

Protus 2.0: Ontology-based semantic recommendation in programming tutoring system. Expert Systems with Applications, 39(15), 12229-12246.

[29] Kulkarni, P. V., Rai, S., & Kale, R. (2020). Recommender system in elearning: a survey. In Proceeding of International Conference on Computational Science and Applications (pp. 119-126). Springer, Singapore.

[30] Rendle, S., Freudenthaler, C., Gantner, Z., & Schmidt-Thieme, L. (2012). BPR: Bayesian personalized ranking from implicit feedback. arXiv preprint arXiv:1205.2618.

[31] Hu, Y., Koren, Y., & Volinsky, C. (2008, December). Collaborative filtering for implicit feedback datasets. In 2008 Eighth IEEE International Conference on Data Mining (pp. 263-272). IEEE.

[32] Zigoris, P., & Zhang, Y. (2006, November). Bayesian adaptive user profiling with explicit & implicit feedback. In Proceedings of the 15th ACM international conference on Information and knowledge management (pp. 397-404).

[33] Adomavicius, G., & Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. IEEE transactions on knowledge and data engineering, 17(6), 734-749.

[34] Oubahssi, L., & Grandbastien, M. (2007, June). Un modèle pour améliorer l'interopérabilité des données pédagogiques apprenant. In 3ième conférence en Environnement Informatique pour l'Apprentissage Humain-EIAH 2007 (pp. 437-448). INRP.

[35] Zriaa, R., & Amali, S. (2020, December). Recommendation of Learning Objects Through Dynamic Learning Style Identification and k-means Clustering Algorithm. In International Conference on Advanced Intelligent Systems for Sustainable Development (pp. 730-748). Springer, Cham.

[36] Kim, H. R., & Chan, P. K. (2003, January). Learning implicit user interest hierarchy for context in personalization. In Proceedings of the 8th international conference on Intelligent user interfaces (pp. 101-108).

[37] Núñez-Valdez, E. R., Quintana, D., Crespo, R. G., Isasi, P., & Herrera-Viedma, E. (2018). A recommender system based on implicit feedback for selective dissemination of ebooks. Information Sciences, 467, 87-98.

[38] Egghe, L., & Leydesdorff, L. (2009). The relation between Pearson's correlation coefficient r and Salton's cosine measure. Journal of the American Society for information Science and Technology, 60(5), 1027-1036.

[39] DUBEY, Ankita and CHOUBEY, Asso Prof Dr Abha, " A Systematic Review on K-Means Clustering Techniques", International Journal of Scientific Research Engineering & Technology (IJSRET), ISSN, vol.6, 6,pp. 2278-0882,June 2017.

[40] Zriaa, R., & Amali, S. (2020, October). A Comparative Study Between K-Nearest Neighbors and K-Means Clustering Techniques of Collaborative Filtering in e-Learning Environment. In The Proceedings of the Third International Conference on Smart City Applications (pp. 268-282). Springer, Cham.