

## Towards an Intelligent Hybrid Recommendation System for E-Learning Platforms Using Data Mining

<https://doi.org/10.3991/ijet.v12i06.6610>

Marouane El Mabrouk

Abdelmalek Essaadi University, Tangier, Morocco  
elmabroukmarouane@gmail.com

Salma Gaou

Abdelmalek Essaadi University, Tetuan, Morocco  
salma.gaou@gmail.com

Mohamed Kamal Rtili

Abdelmalek Essaadi University, Tetuan, Morocco  
rtili.kamal@gmail.com

**Abstract**—Nowadays, Information and communications technology (ICT) becomes a very important thing in human life in different fields. They are used in many fields as information systems (software, middleware) using various telecommunication media to give users the ability to manipulate digital data. In addition, with new technology development, a new concept appeared in the late 90s and early millennium, which is distance learning through e-Learning platform. Recommendation systems become increasingly used in information systems and especially in e-learning platform. These systems are used to propose and recommend content of these platforms to users according to needs of the latter in order to allow them to have the maximum information for learning.

In this paper, we present an intelligent hybrid recommendation system based on data mining. This system has four parts, the first for data collection and for center of interest construction by two modes: explicit data collection, which based on users and what they filled in their profiles, and implicit and automatic data collection by proposing a survey to users in order to gather information about their interest. A second part for processing information already collected in the previous part and for creating the learning model, classifying users who posted the content and classifying content also in order to send the results to the recommendation module. The third part is for making the similarity between learners and content and doing the recommendation for learners and the final part is for creating a log file of recommendation by learner, which will be used in the upcoming recommendation. According to results already done, we noticed that our proposition is satisfactory and the system is well optimized in terms of accuracy, response and processing time compared to the standard recommendation.

**Keywords**—e-learning, recommendation systems, filtering based content, collaborative filtering, hybrid filtering, utility matrix, data mining, decision trees

## **1 Introduction**

The new world of information has changed our lives. However, this impact is modest compared to what we were expecting in the coming years especially in education. Information technologies have changed the way we work. Better, they will change the fundamental aspects of life: health, shopping, education, teaching, etc.

Since the Internet has been adopted and developed as a means of communication by the educational institutions in the 1970s, scholars have had knowledge of its huge potential as a learning tool. Over the past few years, the governments of the countries are more and more excited by the possibilities of online learning for make it profitable and easily accessible education for all ages and social backgrounds, regardless of time and geography.

The researchers have worked hard to provide a remote environment for real-time instantaneous interaction and exchange between persons in order to facilitate learning. The research made have led to the creation of a remote virtual environment for learning called the intelligent tutoring system (ITS).

The intelligent tutoring system (ITS) is a computing environment designed for promoting human learning and constructing the knowledge among a learner [1]. The ITS concerns several themes, among these themes is the e-Learning.

The e-Learning is considered by many as the only reliable solution to the problem of offering and providing the necessary resources to facilitate and improve the learning and education. The e-Learning (Electronic Learning) means, as indicated in the "e" in its name, learning by electronic ways.

According to the European Commission [2]: "e-learning is the use of new multimedia technologies and the Internet to improve the quality of learning by facilitating access to resources and services as well as remote exchanges and collaboration". This clear, synthetic complete definition puts the emphasis on the notion of quality in stating that the objective of the e-Learning is to improve and facilitate the quality of learning. ITS covers several systems with several types of training [3], distance education (EAD), distance training (FAD) and the Open and Distance Learning (ODL).

Considering the importance of the e-learning in our present life in regards to learning among all the world, recommendation systems must be present in the e-Learning platforms to facilitate, propose and recommend articles, books, presentations and courses that will be consulted and followed, as well as valuing the content in the platform. Several works have been made to implement the recommendation systems in e-Learning platforms by using the data mining approach. Our proposed recommendation system is based on data mining and classification techniques, especially the decision trees and other proposals, which will be presented in the next sections.

The rest of the paper is organized as follows:

Section 2 introduces the recommendation systems proposed by other researchers in the literature. Data mining and classification methods as well as the application data mining in the recommendations systems are presented in section 3. Section 4 presents the recommendation systems and their types. Section 5 introduces and presents our new intelligent recommendation system. Section 6 presents the design and the imple-

mentation of the proposed recommendation system. The results are presented in section 7, finally the Conclusion, and the future work in Section 8.

## 2 Related Work

The distance training evolved since the 1970s and the 1980s. Turoff Murray and Hiltz Roxanne Starr [4] have published online courses. The emergence of the Internet has accelerated the evolution of distance training and it has transformed into an electronic training called e-Learning. The Web 2.0 has allowed the evolution of the e-Learning to 2.0 version and allowed considerably to expand the potential of collaborative learning, as well as increasing the number of connected learners.

In [5], the authors presented a recommendation system to recommend to users the adequate e-Learning platforms for their cases, which is based on decision trees using ID3algorithm. The system takes into consideration several attributes namely, the type of questions of the test required for a course or content, the images as a Boolean variable that gives the information to the system that the user will use the images or not. In addition, we find the complexity of the interface for the use of platforms, the price of the platform, the knowledge of the users on the platforms e-Learning and finally the type of files to import. They have presented also the generated decision tree where they have concluded a lot of rules that can be used by the system. As well, they have presented the final interface of their proposed system.

In [6], the authors presented the recommendation system architecture in the field of inter-library services. This system has three layers; the first layer is intended to acquire the data for the database (data for the inter-library database, data concerning the user profiles for user profiles database and data for bibliographies database). The second layer is for the recommendation system processing which contains two modules, one for the pre-processing of the data from the databases and a module of data mining for extraction of the recommendation rules. Finally a presentation layer as a user interface where there are two module; one for processing of the rules extracted in order to recommend and one other module for the feedback of the processing of data.

In [7], the authors have proposed a decisions trees improvement by proposing a method called "**Most Popular Sampling (MPS)**" to improve the construction time of the trees. This improvement is based on verification of the nodes associated to users who have a popular reputation instead of of check all nodes. The second improvement is based on the proposal of a new algorithm for the creation of the decision trees called "**Factorized Decision Trees (FDT)**" which uses the factorization of the matrix to predict the assessments of the nodes of the tree. The authors have exploited the Netflix database to achieve their experiments. According to the authors, the results have been well satisfactory, either for the MPS method that improves the response time of the trees construction without influencing the accuracy or for the Algorithm **FDT**, which improves the precision of the evaluations of the forecast.

In [8], the authors have proposed an algorithm for recommendation, which is based on "**Repeated Incremental Pruning**" method to reduce the rate error produced during the processing, and on "**Predictive Rule Mining**" method to analyze the users in

order to extract the rules of learning. The authors conducted a comparative study with the traditional approaches in citing the advantages and disadvantages of each. The authors have used the data set of the assessments of the movielens platform made by the different users of the platform. According to the authors, the results have been well satisfactory and their proposal is much better than the other approaches in the literature.

### **3 Datamining and Classification Approach**

#### **3.1 Datamining concept**

Data mining is an approach for exploration, exploitation and analysis of data of large volume (Big Data), which organizes the data in a way more understandable. On the other hand, it serves to discover the rules of organization, classification and prediction to assist decision makers in the decision support process. [9].

Data mining is used to increase the capacity of storage and processing of data, and is used in the collection of data [10]. Data mining allows also exploiting data to increase the profitability of the institutions namely companies, government etc ..., and increase the return on investment of their information systems.

#### **3.2 Classification approach**

Classification is a data analysis approach. It serves to facilitate the study and processing of data with large volumes [11], this approach aims to consolidate data into groups to categorize data. It is a method among the methods used in data mining for the processing, analysis and exploitation of important data. The data is grouped into several classes such a way that the data of the same class are as similar as possible and the classes are the most distinct possible.

There are many classification approaches in data mining namely neural networks, Bayesian networks and decision trees.

**Decision trees:** Decision tree is a data structure of statistical machine learning. Its operation is based on heuristics that provide outstanding results in practice. Its appearance makes it very clear and easy regarding reading to be exploited by humans [12].

Decision tree models an hierarchy of tests on values of a set of variables called attributes. After these tests, the predictor produces a numeric value or selects an item in a discrete set of conclusions. We speak of regression in the first case and classification in the second [13].

There was a lot of construction algorithms of decision trees but the most significant works are the CART algorithm (Classification And Regression Trees) proposed by Breiman et al. in 1984 [14], the ID3 algorithm proposed by R. Quinlan in 1986 [15] and C4.5 algorithm which is an improvement of the ID3 algorithm also proposed by R. Quinlan in 1993 [16].

## 4 Recommendation Systems

Recommendation systems have become a very important tool in computer systems namely e-commerce sites, social networks or e-Learning platforms. The aim of these systems is to recommend articles, content or media according to several criteria as applicable for users to facilitate them the searching of information.

Pradel [17] defines the recommendation systems as tools for filtering information (movies, music, books, news, images, Web pages, etc.) to be suggested to users. The recommendation system analyzes user information and its profile as well as its activities to predict and recommend information that suits him.

The purpose of these systems is to do similarity between user and information, by exploiting a set of criteria and information collected about this user and other information as well as information to recommend in order being able to recommend this information to users. Recommendation systems differ depending on the way of filtering information and data. This difference produces several types of recommender systems namely:

- **Filtering based content** relies on the content of articles or information in order to compare a user profile [18]. The system compares the representation of information and articles on the platform with the user's profile to recommend it. An article is recommended based on its similarity to the user profile.
- **Collaborative filtering** relies on the feedback and votes made by users of the platform on the articles in question. These assessments are translated into numerical values that can be exploited in the recommendation. There are two main approaches for collaborative filtering. The approach that refers to users [19]. It consists in comparing the users with each other and finding things in common in order to recommend. The second approach refers to Articles [20]. It consists in comparing the items evaluated by the same users and recommending based on the closest of those they have already evaluated items.
- **Hybrid filtering** exploits the evaluations made by users and information about the contents. These systems may use other sources of information [21]. several methods to make the hybrid filter can be used to combine sources or models. It is possible for example applying separately collaborative filtering and other type of filtering to recommend, and combine these sets of recommendations by weighting, cascading, switching, etc. to produce final recommendations for users [22].

## 5 Proposed Intelligent Hybrid Recommendation System

In this section, we will present our proposed recommendation system. Our recommendation system aims to recommend the most appropriate content for a learner of an e-learning platform. This system helps to build the center of interest for users by facilitating access to content. Most recommendation systems are based on the processing of learner profiles only, however, our system designed to take into consideration several aspects when recommending namely:

- The consideration of center of interest if it exists
- The consideration of evaluations on the content and the number of visits by users of the platform as well as the user's reputation who put the contents
- Construction of the center of interest of learners using surveys proposed by the platform for learners who have not fulfilled or not completed their center of interest to fill it or to evolve it
- Using data collection with dual method, explicit method and implicit method
- Using an hybrid filter that combines content filtering and collaborative filtering in order to recommend
- Using the data mining and especially the decision trees in order to provide recommendations and to classify users and content
- The construction of file log and historical data by user in order to be used during the next recommendation

Our proposed system is divided into four main parts:

- The first part is for the data collection about the learner profile, the content to recommend, the evaluations of the learner, the number of visiting a content and on requests sent into the platform.
- The second part is for processing of the information already collected in the previous section and for creating the learning model, classification of users who have submitted the contents and for classification of content in order to send the results to the recommendation part
- The third part is for making the similarity between learners and content and performing the recommendation
- The fourth part is for creating file log and historical data by user in order to be used during the next recommendation

We have tried to include the recommendation based on the historical data, cookies, and the requests sent on the platform order to turn our system into a system able to recommend for the users who are not registered on the platform.

### 5.1 Data collection and construction of centers of interest

Recommendation system must be able to make predictions about the interests of users. It must collect a number of data of these users in order to be able to build a profile for each user. Our system is based on two steps to make the collection of data needed to build the user profile.

**Explicit collection:** It relies on data and interest entered by the user in the center of interest and on the activities made by the learner on the platform (comments, likes and dislikes, favorites and ratings). The rating uses the standard scale of rating ranging from one star to five stars which is converted into numerical value used in the recommendation module.

**Implicit collection:** It relies on the analysis of requests sent and the analysis of the historical data of the learner on the platform. To do that, we proposed a survey that will appear periodically to users in order to build its center of interest and its profile.

This proposition allows the system to fill automatically the interest of learners for users who have not fulfilled their center of interest or evolved it implicitly. The historical data of the learner is collected by the last part of the proposed system. This historical data is in the form of set of elements consulted on the platform (articles, videos, images, etc ...), consultation frequency of these contents and finally the time spent on such content. The system also analyzes the browser historical data by consulting browser cookies etc ... to be able to recommend to learners.

The survey is in form of questions displayed on the platform like flying messages asking for answers from the learner. The system exploits the answers in the form of binary decision tree to be able to provide the results of questions.

There are several algorithms to draw decision trees; there are algorithms that browse the trees from top to down and vice versa. In 1986, Ross Quinlan proposed an algorithm that made a revolution in the field of machine learning for the decision trees: ID3 algorithm [15].

ID3 built the decision tree recursively. At each step of recursion, it calculates among the remaining attributes of the current branch, the one that maximizes the information gain. I.e., the attribute that makes it easier to classify samples at this level of this branch of the tree.

This algorithm uses two actions to choose the best attribute, measurement of the entropy and measurement of the frequency. We will be interested in measurements of Shannon entropy because it is the best in terms of performance. The following formula represents the Shannon entropy  $H(S)$ :

$$H(S) = - \sum_{x \in X} p(x) \log_2 p(x) \quad (1)$$

The following algorithm is proposed to select the best attribute and perform the procedure of processing and searching:

---

### ID3 (Examples, Target\_Attribute, Attributes)

---

#### Begin

Create a root node for the tree

**If** all examples are positive, **Return** the single-node tree Root, with label = +.

**If** all examples are negative, **Return** the single-node tree Root, with label = -.

**If** number of predicting attributes is empty, **Then Return** the single node tree Root, with label = most common value of the target attribute in the examples.

**Otherwise Begin**

A ← The Attribute that best classifies examples.

Decision Tree attribute for Root = A.

**For each** possible value,  $v_i$ , of A,

    Add a new tree branch below Root, corresponding to the test  $A = v_i$ .

    Let  $\text{Examples}(v_i)$  be the subset of examples that have the value  $v_i$  for A

**If**  $\text{Examples}(v_i)$  is empty

**Then** below this new branch add a leaf node with label = most common target value in the examples

**Else** below this new branch add the subtree ID3 ( $\text{Examples}(v_i)$ , Target\_Attribute, Attributes – {A})

**End For Each**

**End**

**Return** Root

**End**

---

We will present in coming section an example of the functioning of our system and we will illustrate the functioning of our decision tree and how the survey works to fill the center of interest of the learner.

Table 1 shows an example of rules that can be taken by the system to discover the learners' interest center.

When the decision is positive, the system saves all information regarding the center of interest of the learner, the field, the subject, the author and books.

Table 2 presents the list of decisions to be taken by the system to build the center of interest of the learner.

Figure 1 shows an excerpt of the tree generated for the following example by the system.

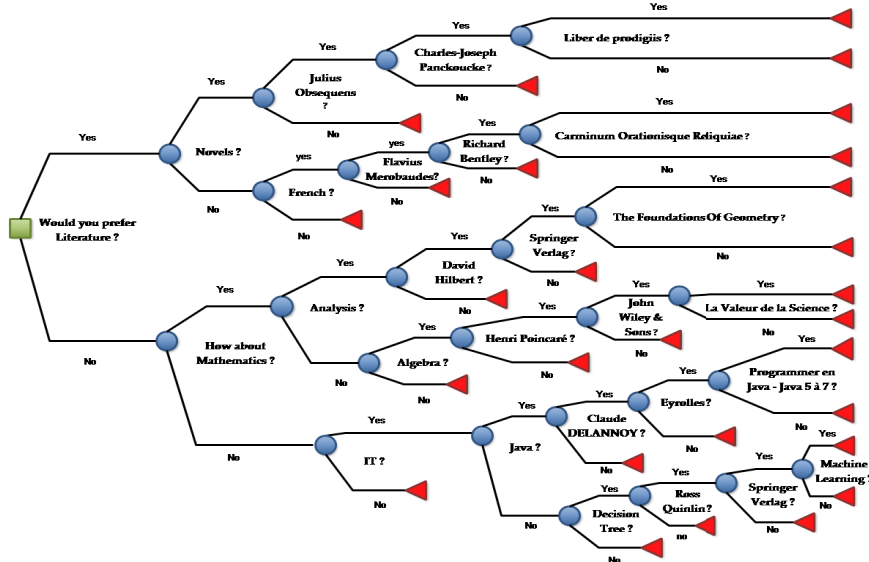


**Table 1.** Rules examples for the decision tree

Attribute	Possible values
Field	Literature, Mathematics, Computer Science
Subject	Roman, French, Analysis, Algebra, Java, Decision Tree
Author	Julius Obsequens, Flavius Merobaudes, David Hilbert, Henri Poincaré, Claude DELANNOY, Ross Quinlin
Publishing house	Charles-Joseph Panckoucke, Richard Bentley, Eyrolles, John Wiley & Sons, Larousse, Springer Verlag
Book	Liber de prodigiis, Carminum Orationisque Reliquiae, The Foundations Of Geometry, La Valeur de la Science, Programmer en Java - Java 5 à 7, Machine Learning
Decision	Yes or No

**Table 2.** List of decisions for the decision tree

Field	Subject	Author	Publishing house	Book	Decision
Literature	Roman	Julius Obsequens	Charles-Joseph Panckoucke	Liber de prodigiis	Yes
Literature	French	Flavius Merobaudes	Richard Bentley	Carminum Orationisque Reliquiae	Yes
Mathematics	Analysis	David Hilbert	Springer Verlag	The Foundations Of Geometry	Yes
Mathematics	Algebra	Henri Poincaré	John Wiley & Sons	La Valeur de la Science	Yes
Computer Science	Java	Claude DELANNOY	Eyrolles	Automat und Mensch: Über menschliche und maschinelle Intelligenz	Yes
Computer Science	Decision Tree	Ross Quinlin	Springer Verlag	Machine Learning	Yes



**Fig. 1.** Generated decision tree by our system

### 5.2 Classification of users that posted the contents

Our system classifies the users of the platform in three types, new users, the least reputed users and well reputed users. We proposed a formula to calculate the margin in order to indicate the interval for each class type. The formula proposed:

$$Range = \left| \frac{\sum(R_L) + \sum(RC_S)}{3} \right| \tag{2}$$

Where  $R_L$  is the total number of evaluations on all learners and  $RC_S$  is the total number of evaluations of content. Evaluations are in the form of number of stars of a scale ranging between zero and five star rating. The system calculates intervals and then for each user, its evaluation is added to the evaluation of such content to put him in a category. Here is the formula calculated for each user:

$$Rating\ U_i = \sum(R_{L_i}) + \sum(RC_{S_i}) \tag{3}$$

Where  $RL_i$  represents, the total number of evaluations of a learner and  $RCS_i$  represents the total number of evaluations of the contents of this learner.

Therefore, the system classifies users who have posted content using decision trees as follows:

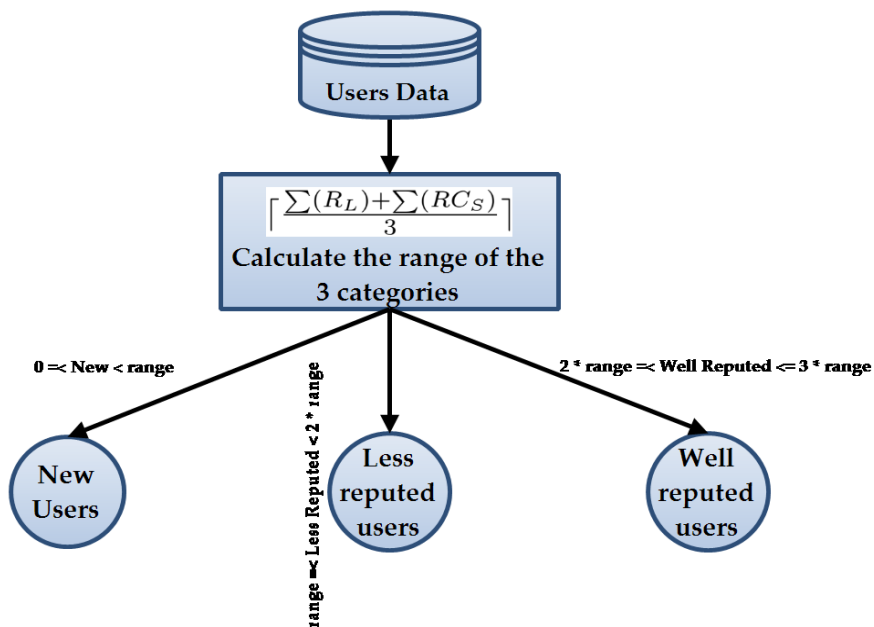


Fig. 2. Classification of users who submitted the contents

### 5.3 Classification of the well reputed users that posted the contents according to the ratings and content category

Our proposed system performs the content classification in stages, classification on category for put the content in its category in order to facilitate the association of the content according to the learners' interest center. The second classification is based on the ratings.

**Classification on category:** In this stage, the system tries to associate each content in its category to facilitate the matching task with the interests of learners in the recommendation. Each content has a category field in the database to use it in classification. The system groups items by category using decision trees. Below is an illustration of how the system works to do this classification:

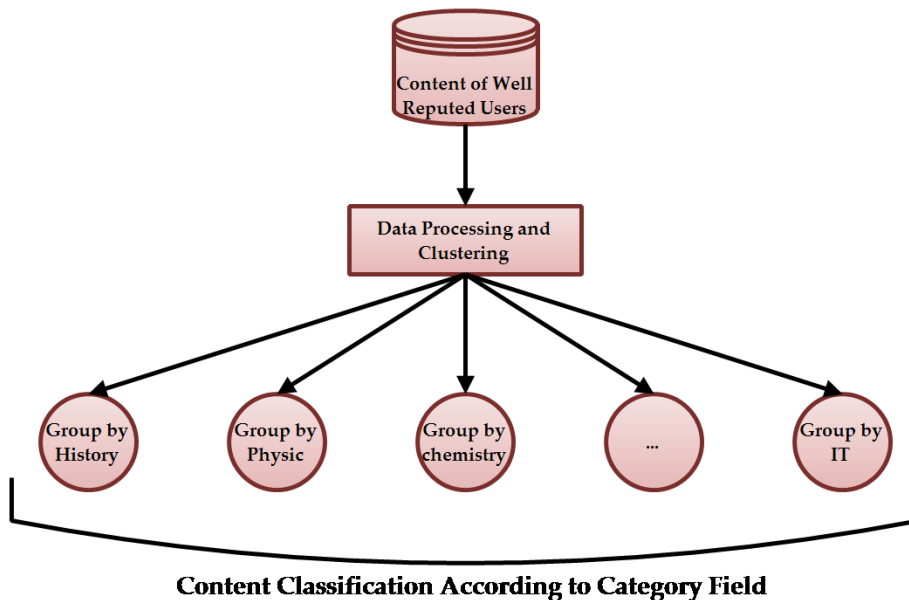


Fig. 3. Content Classification according to Category Field

**Classification based on ratings:** In this stage, our proposed system classifies each category according to evaluations made on the contents. We designed the system to take into consideration the evaluation on the content i.e., the number of stars voted for this content, user reputation and the number of likes and dislikes in order to evaluate the content . We proposed a formula to group all these criteria into a single value calculated by our proposed formula. Here is the proposed formula:

$$CGR_i = \frac{Rating_{Content_i} + DSL_i + Rating_{Reputation_i}}{3} \tag{4}$$

Where:

$CGR_i$  is Content General Rating

$Rating_{Content_i}$  are evaluation and votes of learners who have followed or who have read the contents. This evaluation is in the form of scale from zero stars to five stars as follows:

- Null = 0
- Poor = 1
- Medium = 2
- Good = 3
- Very good = 4
- Excellent = 5

$DSL_i$  (**DiSlikeLike**) are likes and dislikes that are made by learners to express their appreciations or their disgruntled towards the content posted on the platform. This evaluation is converted and calculated by the following proposition:

- Likes = 1
- Dislikes = 0

$Rating_{Reputation_i}$  is the user reputation that posted the content. Learners make their votes and ratings on users that posted the content based on its activities on the platform, its relevant content as well as his help and explanations for learners. This evaluation is a scale from zero stars to five stars and is divided into six categories as follows:

- Null = 0
- Poor = 1
- Medium = 2
- Good = 3
- Very good = 4
- Excellent = 5

#### 5.4 Construction of the proposed personalized matrix utility

There are many methods for prediction the content that interest learners. All these methods need information regarding learners and content in order to be exploited and analyzed by the recommendation system. This information is gathered into a matrix called matrix utility.

[23] defined the recommendation by a set of all learners and a set of all content that can be recommended.  $R$  is a set of learners voting and rating on content posted on the platforms. Based on this matrix, the system tries to recommend content that has the highest rating and the highest number of learners. In the previous section, we proposed a formula that includes all the criteria into a single value calculated by our proposed formula.

Consider learners  $L$  and  $C$  content,  $R$  is the set of votes and ratings where  $L \times C \rightarrow R$ .

For each learner  $l \in L$ , there is a content  $c' \in C$  that is recommended by the system and maximizes reputation of  $l$

$$\forall l \in L, c'_l = \arg \max_{c \in C} r(c, l) \tag{5}$$

Consider the following table that contains an example of the information used during the creation of the matrix:

**Table 3.** Content 1

	RatingContent	DSL	RatingReputation	CGR1
<i>L1</i>	5	1	4	3,33333333
<i>L2</i>	4	1	4	3
<i>L3</i>	2	0	4	2
<i>L4</i>	5	1	4	3,33333333
<i>L5</i>	0	0	4	1,33333333
<i>L6</i>	3	1	4	2,66666667

**Table 4.** Content 2

	RatingContent	DSL	RatingReputation	CGR2
<i>L1</i>	1	0	2	1
<i>L2</i>	0	0	2	0,66666667
<i>L3</i>	2	0	2	1,33333333
<i>L4</i>	0	0	2	0,66666667
<i>L5</i>	5	1	2	2,66666667
<i>L6</i>	3	1	2	2

**Table 5.** Content 3

	RatingContent	DSL	RatingReputation	CGR3
<i>L1</i>	5	1	5	3,66666667
<i>L2</i>	5	1	5	3,66666667
<i>L3</i>	4	1	5	3,33333333
<i>L4</i>	5	1	5	3,66666667
<i>L5</i>	5	1	5	3,66666667
<i>L6</i>	4	1	5	3,33333333

**Table 6.** Content 4

	RatingContent	DSL	RatingReputation	CGR4
<i>L1</i>	0	0	1	0,33333333
<i>L2</i>	0	0	1	0,33333333
<i>L3</i>	2	0	1	1
<i>L4</i>	4	1	1	2
<i>L5</i>	0	0	1	0,33333333
<i>L6</i>	1	0	1	0,66666667

Now, we will present the utility matrix according to our proposition:

**Table 7.** Utility Matrix

	CGR1	CGR2	CGR3	CGR4
<i>L1</i>	3,33333333	1	3,66666667	0,33333333
<i>L2</i>	3	0,66666667	3,66666667	0,33333333
<i>L3</i>	2	1,33333333	3,33333333	1
<i>L4</i>	3,33333333	0,66666667	3,66666667	2
<i>L5</i>	1,33333333	2,66666667	3,66666667	0,33333333
<i>L6</i>	2,66666667	2	3,33333333	0,66666667

### 5.5 Recommendation of content to learners based on similarity Learners - Contents

In order to do the similarity between learners and content posted on the platform, you have to find some kind of similarity between the rows and columns of the utility matrix. The choice of approach depends on information contained in the e-Learning platform. There are many approaches and methods to measure the similarity between learners and content. In our case, we have chosen the cosine similarity [24, 25, 26] to recommend to learners.

The cosine similarity can be calculated by the following formula:

$$S_{\text{cosine}(a,b)} = \frac{\sum_{x \in E_a \cap E_b} u(a, x) \times u(b, x)}{\sqrt{\sum_{x \in E_a \cap E_b} u(a, x)^2 \sum_{x \in E_a \cap E_b} u(b, x)^2}} \quad (6)$$

Where *a* and *b* are two learners or two contents and  $E_a$  is a set of contents measured by the learner *a* and  $E_b$  is a set of contents measured by the learner *b*.

In the section coming we will present the interpretation of information of the utility matrix by the system for recommendation.

**Recommendation based on similarity Learner – Learner:** Consider the example of the following utility matrix presented in the previous section:

**Table 8.** Utility Matrix

	CGR1	CGR2	CGR3	CGR4
<i>L1</i>	3,33333333	1	3,66666667	0,33333333
<i>L2</i>	3	0,66666667	3,66666667	0,33333333
<i>L3</i>	2	1,33333333	3,33333333	1
<i>L4</i>	3,33333333	0,66666667	3,66666667	2
<i>L5</i>	1,33333333	2,66666667	3,66666667	0,33333333
<i>L6</i>	2,66666667	2	3,33333333	0,66666667

When the system recommends for learner  $L_1$ , it calculate the cosine similarity of  $L_1$  with all learners in its category as follows to determine its similarity with the other learners.

Iteration 1: Cosine Similarity of  $L_1$  and  $L_2$

$$S_{\cosine(L_1, L_2)} = \frac{3,3333333 \times 3 + 1 \times 0,6666667 + 3,6666667 \times 3,6666667 + 0,3333333 \times 0,3333333}{\sqrt{3,3333333^2 + 1^2 + 3,6666667^2 + 0,3333333^2} \times \sqrt{3^2 + 0,6666667^2 + 3,6666667^2 + 0,3333333^2}}$$

$$S_{\cosine(L_1, L_2)} = 0,996931528369$$

Iteration 2: Cosine Similarity of  $L_1$  and  $L_3$

$$S_{\cosine(L_1, L_3)} = 0,958610229606$$

Iteration 3: Cosine Similarity of  $L_1$  and  $L_4$

$$S_{\cosine(L_1, L_4)} = 0,892476565758$$

Iteration 4: Cosine Similarity of  $L_1$  and  $L_5$

$$S_{\cosine(L_1, L_5)} = 0,860879919727$$

Iteration 5: Cosine Similarity of  $L_1$  and  $L_6$

$$S_{\cosine(L_1, L_6)} = 0,967347818696$$

Since a large value of cosine (positive) means a smaller angle so a smaller distance, thus these iterations gives us the following result:

Ranking of learners according to similarity with learning L1
L2
L6
L3
L4
L5

We proposed to give the possibility the administrators of the e-Learning platform to configure a proposed parameter called K that aims to specify the TOP K, which is the number of learners to recommend when displaying the recommendation result. On the above example, we have five learners that are classified according to similarity

with the learner  $L_1$ , so if for example  $K = 2$ , so the system recommends content published or viewed by learners  $L_2$  and  $L_6$  and if  $K = 3$  then learners  $L_2, L_6$  and  $L_4 \dots$  etc.

### 5.6 Classification of content for unregistered users (Passive Collaborative Filtering)

Our system performs an analysis of learner’s behavior in the "background" using a proposed survey in the module of data collection, exploiting the requests sent on the platform, cookies of the platform and log files information of the learner on the browser or on the platform to deduce its center of interest in order to use it in the recommendation. The proposed survey as already said is a survey with several questions displayed on the platform like flying messages asking for answers from the learner. The system exploits these answers in the form of binary decision tree to be able to deduce the result of the issues. After construction of the center of interest, the system use it to classify contents according to the categories of center of interest. We proposed the passive recommendation solution using the recommendation based on similarity Content - Content.

#### Recommendation based on similarity Content - Content

Consider the example of the following utility matrix presented in the previous section:

Table 9. Utility Matrix

	CGR1	CGR2	CGR3	CGR4
L1	3,33333333	1	3,66666667	0,33333333
L2	3	0,66666667	3,66666667	0,33333333
L3	2	1,33333333	3,33333333	1
L4	3,33333333	0,66666667	3,66666667	2
L5	1,33333333	2,66666667	3,66666667	0,33333333
L6	2,66666667	2	3,33333333	0,66666667

When the system recommends content to a learner, it calculate the cosine similarity of content already visited by this learner with all content in its category as follows to recommend the best among them. We assumed in our example that the content already visited is the content  $C_1$  content, so we worked with the  $CGR_1$ .

Iteration 1: Cosine Similarity of  $CGR_1$  and  $CGR_2$

$$S_{\cosine(CGR_1, CGR_2)} = \frac{3,33333333 \times 1 + 3 \times 0,66666667 + 2 \times 1,33333333 + 3,33333333 \times 0,66666667 + 1,33333333 \times 2,66666667 + 2,66666667 \times 2}{\sqrt{3,33333333^2 + 3^2 + 2^2 + 3,33333333^2 + 1,33333333^2 + 2,66666667^2} \times \sqrt{1^2 + 0,66666667^2 + 1,33333333^2 + 0,66666667^2 + 2,66666667^2 + 2^2}}$$

$$S_{\cosine(CGR_1, CGR_2)} = 0,748818358036$$



Iteration 2: Cosine Similarity of  $CGR_1$  and  $CGR_3$

$$S_{\cosine(CGR_1, CGR_3)} = 0,965246603853$$

Iteration 3: Cosine Similarity of  $CGR_1$  and  $CGR_4$

$$S_{\cosine(CGR_1, CGR_4)} = 0,813351635265$$

Since a large value of cosine (positive) means a smaller angle so a smaller distance, thus these iterations gives us the following result:

Ranking of contents according to similarity with the content CGR1
CGR3
CGR4
CGR2

The same as the recommendation of learners, we proposed to give the possibility the administrators of the e-Learning platform to configure a proposed parameter called K that aims to specify the TOP K, which is the number of contents to recommend when displaying the recommendation result. On the above example, we have three contents that are classified according to similarity with the content  $CGR_1$ , so if for example  $K = 2$ , so the system recommends the content  $CGR_3$  and  $CGR_4$  and if  $K = 3$  then learners  $CGR_3$ ,  $CGR_4$  and  $CGR_2$  etc...

### 5.7 Processing log file information of learners

The creation and processing of historical data and file log information serve to give a recommendation list at each connection or at each consultation of the e-Learning platform. This process is also used to recommend content already recommended to be consulted again and recommend new content that are similar to content already recommended.

There are two types of recommendation based on historical data and file log information proposed by our system:

- For learners registered on the platform: historical data of recommendations of the learner is saved by our system to be recommended after, ie, this historical data is used to recommend content already recommend and new content similar to the content already recommend.
- For non-registered learners on the platform: it is based on browser history, so the system creates a session and cookies to retrieve information consulted by the user. Also, to have a traceability of all requests or content consulted in the platform.

In both cases, our system is based on content-based recommendation to recommend similar new contents that have just been posted on the platform classified by category.

## 6 Design and Implementation of the Proposed Recommendation System on E-Learning Platform

In this part, we will introduce the e-Learning platform and our recommendation system based as already said on several modules to make the recommendation. The first module for data collection implicitly and explicitly. The second module for processing the collected data in order to create the learner profile, the classification of the users and the classification of the contents. The third module for making the recommendation based on the proposed model and the fourth module for the creation of log files and user historical data to be used in the upcoming recommendation processes.

For the data collection of the first module, as already mentioned, we have designed two methods: explicit and implicit. The first one is based on the information entered by the user when registering on the platform or during the modification of the profile to evolve the information and the center of interest. The next figures shows our proposed recommendation system:



Fig. 4. Result recommendation and construction of log files and historical data

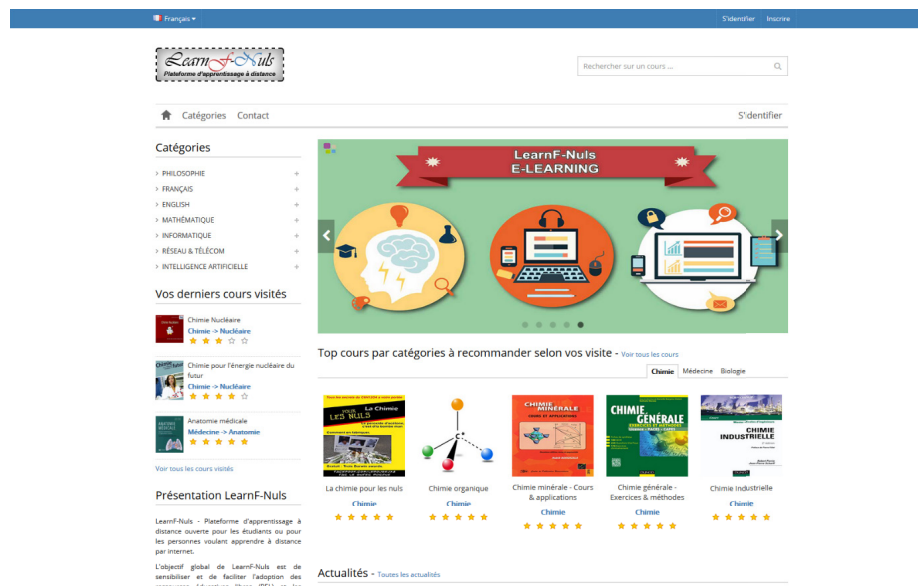


Fig. 5. Recommendation for unregistered users

## 7 Results and Discussion

To calculate the accuracy and the response time of our proposed system, we used a data set of an e-Learning platform that contains 700 users and 1000 content. This set comprises 70512 total of evaluations. As already mentioned before, the evaluation has scale that ranges from 0 to 5 stars where 0 to 2 is a poor evaluation, 3 is average and 4 to 5 star represents a good evaluation. In our results, we selected using the classification proposed in this paper only contents which their users who posted these contents have a good reputation on the platform and the contents are contents that have a relationship with literature. Figure 6 shows the set.

We also used a data set of reputations of users who posted these contents to be used in our proposition.

Therefore, after applying our proposition to get CGR according to the likes and dislikes and according to the reputation of users who posted these contents, we got our final matrix with which our system uses it to recommend content. Figure 7 shows the resulting matrix.

After getting the final matrix, the system executes the process of recommendation and displays the content list recommended according to the center of interest of learn-er as shown in Figure 8.



# Contents	Final Results
C <sub>9</sub>	6.5580389547141
C <sub>15</sub>	6.40467145127
C <sub>12</sub>	6.3976162943189
C <sub>10</sub>	6.3545321592154
C <sub>20</sub>	6.2711647387643
C <sub>29</sub>	6.1197757964326
C <sub>28</sub>	6.0796833339916
C <sub>33</sub>	6.0272575055675
C <sub>19</sub>	5.7215750159401
C <sub>30</sub>	5.6032878231148
C <sub>3</sub>	5.5996068558417
C <sub>34</sub>	5.5433236632896
C <sub>22</sub>	5.3645702255661
C <sub>25</sub>	5.3591407011913
C <sub>2</sub>	4.9300417251237
C <sub>32</sub>	4.9300417251237
C <sub>18</sub>	4.9164201396501
C <sub>17</sub>	4.804354692307
C <sub>6</sub>	4.4344658931827
C <sub>14</sub>	3.8113735485431
C <sub>4</sub>	3.73334838921
C <sub>27</sub>	3.6434631549884
C <sub>23</sub>	3.5898298231558
C <sub>5</sub>	3.2719485868036
C <sub>16</sub>	2.2816591088241
C <sub>13</sub>	2.195528381508
C <sub>31</sub>	2.181113572949
C <sub>26</sub>	2.0861514160746
C <sub>24</sub>	1.1547211257576

Fig. 8. Recommended Content List

According to the method of the cosine distance, content that has the highest value is the most appropriate content for the learner. In the figure, the recommendation is made for learner number one (LA). Notice that the contents number nine is the most suitable. After, it comes the content number 15 and so on.

To evaluate our proposal in terms of accuracy and response time needed to recommend, we conducted 20 experiments. We started with a matrix of 35 learners and 50 content. In each time, we increase the number of learners by 35 and the number of content by 50.

To evaluate the performance and accuracy of our proposed system for recommendation against the standard recommendation, we used the following equation [27]:

$$Accuracy = \frac{ITPR}{ITPR + IFPR} \tag{7}$$

ITPR: Item True Positive Recommendation. Case was positive i.e. the data and the rules were good and the recommendation is also positive.

IFPR: Item False Positive Recommendation. Case was positive i.e. the data and rules and all but unfortunately the recommendation was negative.

We made the experiments for our proposal and for the standard recommendation. In addition, we tried to compare experiences and we got the following results:

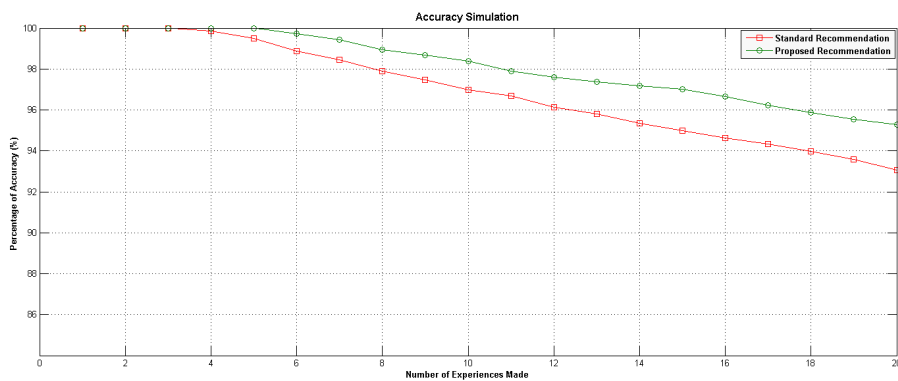


Fig. 9. Accuracy simulation

We noticed that the accuracy for our proposition and for the standard recommendation degrades when increasing matrix size (number of content and number of learners with the evaluations) in each experiment. At the end, we observed that our proposed system remains beyond the threshold that we set to satisfy the recommendation that is 90%, and degrades slowly unlike the standard recommendation that degrades very quickly with time at each increasing in volume of the matrix so that at certain volume, it will no longer satisfy the recommendation set to 90% with an uncertainty of 10% margin.

The second part of our results was for the comparison between our proposed recommendation system and the standard recommendation. To do it, we calculated the response time for both systems in each experiment and we got the result shown in Figure 10.

During the first two experiments, we noticed that the standard recommendation is faster than our proposal, but starting from the third experiment our proposition starts to be faster than the standard recommendation. It was also noted that when the volume of the matrix increases, the standard recommendation performance decreases, so it influence on its response time unlike our proposed system that manages very well the processing and response time and keeps its processing and response performance even if the volume of the matrix increases.

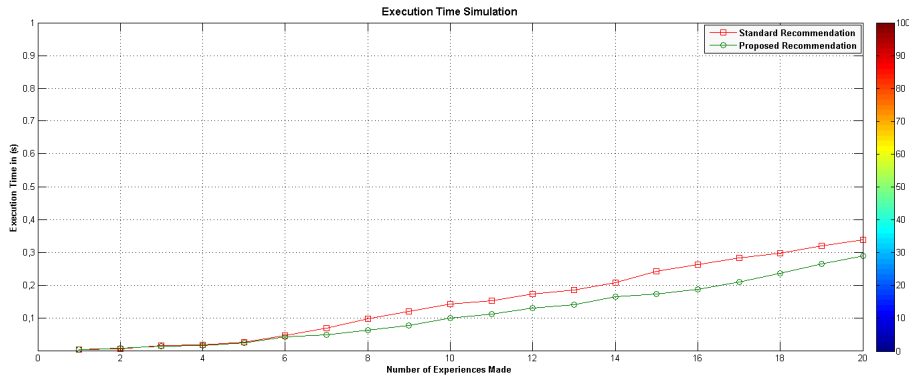


Fig. 10. Execution Time simulation

## 8 Conclusion and Perspectives

In this paper, we presented a hybrid intelligent recommendation system for e-Learning platforms based on data mining. The recommendation system aims to recommend the most appropriate contents for learners of e-learning platform. It assists in build the center of interest for users by facilitating access to content.

The system is divided into four main parts:

- The first part for data collection
- The second part for processing of information already collected in the previous section
- The third part for the similarity between learners and content and make the recommendation
- The fourth part for creating a historical data of recommendation by learner.

For data collection, and construction of the learner profile we used the explicit and implicit collection. In the recommendation, we proposed several modules, a module for classification of users who posted contents according to their reputations, a module to classify these users according to content category and according to the evaluations made by learners of the platform. A module for the construction of the proposed utility matrix and the recommendation module that is based on learner profile and content. The last part for the recommendation based on retrieved and recorded historical for both types of learners registered and non-registered on the platform in order to recommend content not only already recommended but also the new similar content by category. The recommendation used in this case is the one based on the content.

For future work, we will try to work with multi-agent systems to manage the modules of our proposed systems to improve the performance in terms of processing and response time because we will use the distributed processing, which reduces the response time. In addition, multi-agent systems will ensure the accuracy of our proposed system in terms of recommendation. Moreover, we will transform our recommendation system into a generic module that works and that installs easily on the

entire well known e-Learning platform under free license like (Chamilo, Claroline, eFront, Moodle, SAKAI, Etc.).

## 9 References

- [1] Tchounikine, P. "Pour une ingénierie des Environnements Informatiques pour l'Apprentissage Humain". *Revue I3 Information-Interaction-Intelligence* 2(1). 2002.
- [2] Commission européenne. Direction générale de l'éducation et de la culture. *E-Learning: faire mieux pour l'Europe*. Office des publications officielles des Communautés européennes, 2003
- [3] Murray, T . "Expanding the knowledge acquisition bottleneck for intelligent tutoring systems". *International J. of Artificial Intelligence in Education*. Vol.8, No.3-4, pp.222-232.
- [4] Karacapilidis, Nikos. "Novel Developments in Web-Based Learning Technologies: Tools for Modern Teaching." IGI Global, 2010. 1-430 <https://doi.org/10.4018/978-1-60566-938-0>
- [5] V. Lopez, M. Santos, P. Esteve, D. Roldán and J. C. deMendoza, "Practical applications of Intelligent Systems: An E-learning platforms recommender," 2010 International Conference on Intelligent Systems and Knowledge Engineering (ISKE), Hangzhou, 2010, pp. 628-633. <https://doi.org/10.1109/ISKE.2010.5680766>
- [6] A. Jalalimanesh, M. Mansoury and H. Gandomi, "Recommender system based on data mining: Interlibrary case study," 20th Iranian Conference on Electrical Engineering (ICEE2012), Tehran, 2012, pp. 806-809. <https://doi.org/10.1109/IranianCEE.2012.6292464>
- [7] R. Karimi, M. Wistuba, A. Nanopoulos and L. Schmidt-Thieme, "Factorized Decision Trees for Active Learning in Recommender Systems," 2013 IEEE 25th International Conference on Tools with Artificial Intelligence, Herndon, VA, 2013, pp. 404-411. <https://doi.org/10.1109/ICTAI.2013.67>
- [8] A. Pujahari and V. Padmanabhan, "An Approach to Content Based Recommender Systems Using Decision List Based Classification with k-DNF Rule Set," 2014 International Conference on Information Technology (ICIT), Bhubaneswar, 2014, pp. 260-263. <https://doi.org/10.1109/ICIT.2014.13>
- [9] Daniel T-Larose. *Des données à la connaissance, une introduction au data-mining*. Vuibert, Paris, 2005. Traduction de *An introduction to data-mining*, New-York, 2005. Inclus: une version d'évaluation de Clémentine (SPSS).
- [10] Michel Jambu. *Introduction au Data Mining, Analyse intelligente des données*. Eyrolles, 1999.
- [11] Stéphane Tuffery. *Data mining et statistique décisionnelle*. Editions Technip, Août 2005.
- [12] Laurent Hyafil and R. L. Rivest. Constructing Optimal Binary Decision Trees is NP-complete. *Information Processing Letters*, 5(1):15–17, 1976. [https://doi.org/10.1016/0020-0190\(76\)90095-8](https://doi.org/10.1016/0020-0190(76)90095-8)
- [13] Leo Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001. <https://doi.org/10.1023/A:1010933404324>
- [14] Leo Breiman, Jerome Friedman, Charles J. Stone, and R. A. Olshen. *Classification and Regression Trees*. Chapman and Hall/CRC, 1 edition, January 1984.
- [15] J. Ross Quinlan. Induction of decision trees. *Machine Learning*, 1(1):81–106, 1986. <https://doi.org/10.1007/BF00116251>
- [16] J. Ross Quinlan. 1993. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.



- [17] Bruno Pradel. Evaluation des systèmes de recommandation à partir d'historiques de données, 2013. 108 pages.
- [18] Pazzani M., Billsus D., " Content-Based Recommendation Systems ", p. 325-341, 2007. [https://doi.org/10.1007/978-3-540-72079-9\\_10](https://doi.org/10.1007/978-3-540-72079-9_10)
- [19] Resnick P., Iacovou N., Suchak M., Bergstrom P., Riedl J., « GroupLens : an open architecture for collaborative filtering of netnews », CSCW '94 : Proceedings of the 1994 ACM conference on Computer supported cooperative work, ACM, New York, NY, USA, p. 175-186, 1994. <https://doi.org/10.1145/192844.192905>
- [20] Sarwar B., Karypis G., Konstan J., Reidl J., « Item-based collaborative filtering recommendation algorithms », WWW '01 : Proceedings of the 10th international conference on World Wide Web, ACM, New York, NY, USA, p. 285-295, 2001. <https://doi.org/10.1145/371920.372071>
- [21] Pazzani M. J., « A Framework for Collaborative, Content-Based and Demographic Filtering », Artif. Intell. Rev., vol. 13, n° 5-6, p. 393-408, 1999. <https://doi.org/10.1023/A:1006544522159>
- [22] Burke R., « Hybrid Web Recommender Systems », in P. Brusilovsky, A. Kobsa, W. Nejdl (eds), The Adaptive Web, vol. 4321 of Lecture Notes in Computer Science, Springer, chapter 12, p. 377-408, 2007. [https://doi.org/10.1007/978-3-540-72079-9\\_12](https://doi.org/10.1007/978-3-540-72079-9_12)
- [23] J. A. Konstan, B. N. Miller, D. Maltz, J. L. Herlocker, L. R. Gordon, and J. Riedl. GroupLens : Applying Collaborative Filtering to Usenet News. Commun. ACM, 1997
- [24] J. Breese, D. Heckerman, and C. Kadie, Empirical analysis of predictive algorithms for collaborative filtering, in Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence, 1998
- [25] B. Sarwar, G. Karypis, J. Konstan, and J. Reidl. Item-Based Collaborative Filtering Recommendation Algorithms. Proceedings of the 10th international conference on World Wide Web, 2001
- [26] Pang-Ning Tan, Michael Steinbach, and Vipin Kumar. 2005. Introduction to Data Mining, (First Edition). Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.
- [27] Makhoul, J.; Kubala, F.; Schwartz, R. & Weischedel, R. (1999), ' Performance Measures For Information Extraction' Proceedings of the DARPA Broadcast News Workshop.

## 10 Authors

**Marouane EL MABROUK** is a doctor and researcher in the Mathematics and Application Laboratory, IEEE Member and holder of PhD from the Faculty of Sciences and Technologies, Abdelmalek Essaadi University, Tangier, Morocco.

**Salma GAOU** is a doctor and researcher in LIROSA Laboratory, IEEE Member and holder of PhD from the Faculty of sciences, Abdelmalek Esaâdi University, Tetuan, Morocco.

**Mohamed Kamal RTILI** is a PhD student in LIROSA Laboratory of the Faculty of sciences, Abdelmalek Esaâdi University, Tetuan, Morocco.

Article submitted 02 January 2017. Published as resubmitted by the authors 09 March 2017.