Building an e-learning recommender system using Association Rules techniques and R environment

Karim DAHDOUH
Engineering Sciences Laboratory, FPT
Sidi Mohamed Ben Abdellah University
Taza, Morocco
karim.dahdoh@gmail.com

Lahcen OUGHDIR
Engineering Sciences Laboratory, FPT
Sidi Mohamed Ben Abdellah University
Taza, Morocco
lahcen.oughdir@usmba.ac.ma

Ahmed DAKKAK
Engineering Sciences Laboratory, FPT
Sidi Mohamed Ben Abdellah University
Taza, Morocco
ahmed.dakkak@usmba.ac.ma

Abdelali IBRIZ
High School of Technology
Sidi Mohamed Ben Abdellah University
Fez, Morocco
a.ibriz@gmail.com

***Abstract*—To help the human mind in its process of selecting and filtering information, several recommendation systems have been developed in multiple application domains, such as e-commerce and e-tourism. Recommender systems are built to guide end users to easily find the most proper items on huge amount of data. In other words, this type of system tries to suggest items that best meet the preferences and needs of a user. It can be represented as a unit for predicting the behavior of a user by anticipating his next actions.**

In this article, we develop a courses recommender system dedicated to online learning environment. It aims to discover relationships between student's courses activities using association rules method in order to help the student to choose the more appropriate learning materials. We also focus on the analysis of past historical data of the courses enrollments or log data. The article discusses particularly the frequent itemsets concept to determine the interesting rules in the transaction database. Then, we use the extracted rules to find the catalog of more suitable courses according to the learner's behaviors and preferences. Next, we implement our system using the FP-growth algorithm and R programming language. Finally, the experimental results prove the effectiveness and reliability of the proposed system to increase the quality of student's decision and orientate them during the learning process by providing most relevant pedagogical resources.

***Keywords*—online learning; e-learning; recommendation engine, course recommender system; association rules; FP-growth algorithm; R language; frequent itemset**

I. INTRODUCTION

The computing environment for human learning is changing rapidly, due to the emergence of new information and communication technology such as big data and cloud computing[1]. Furthermore, the learning methods are changing every day. Therefore, e-learning systems need to develop more techniques and tools to meet the increased needs of millions of learners around the world.

Generally, there are three approaches to build a recommendation system including content-based filtering, collaborative filtering, and the hybrid model combining the first two types of filtering[2].

In a content-based recommendation system, the choice of items to recommend is based on a comparison of the themes and features discussed in these items against those defined in the user profile. The recommendation mechanism for this type of system is performed by identifying the resources similar to those appreciated by a user according to their content. This recommendation technique consists of analyzing the content of the resources to be recommended or descriptions of them in order to determine which resources are likely to be the most useful for a given user. Unlike the content-based filtering approach that focuses on a single user, collaborative filtering recommendations use the preferences of multiple users. This method tries to form a group of users with the same preferences. In a collaborative recommendation system, the idea is either to focus specifically on the new item that would be likely to please the user but to look at which items have appreciated users close to the current user. Thus, there is no analysis of the subject or content of the objects to recommend. This type of system is very effective in case the content of objects is complex or difficult to define (eg educational resources, films). The third approach to developing recommender engines is the hybrid approach. This can be defined as a combination of two or more different methods. For example, such a system can use both collaborative filtering techniques and content-based filtering techniques, and so on.

In addition, over the past decade, we have witnessed the development of several new families of information filtering systems, such as geographic location-based systems, ontology-based information filtering systems, collaborative annotations, social network-based information filtering systems, etc.

In this work, we chose to use the collaborative filtering approach in the development of the proposed course recommendation system. The choice of a recommendation method depends in the first place on the available data on the elements to be recommended and the user profiles. A recommendation system must adapt to the data on which it will rely to offer relevant recommendations to a user. In fact, collaborative filtering is the basis of the recommendation, the methods of filtering content are rather related to information retrieval systems used by search engines like Google. In addition, recommendation systems using content filtering are quite rare and very old. In fact, most current recommendation engines use collaborative filtering. For example, the Amazon e-commerce giant [3] offers a recommendation system for a wide range of products (books, electronic equipment, etc.) on its website through collaborative methods. In addition, a content-based recommendation system has a higher level of complexity than the collaborative filtering technique. This complexity depends mainly on problems related to the extraction and analysis of semantic data on the items. Indeed, if it proves difficult to release some properties on the items automatically, the algorithms of recommendation by the content become useless. For example, the quality and diversity of multimedia data (image, video, audio or web page) of a learning platform may represent important information for some learners, this information is difficult to extract automatically from an online course. Furthermore, collaborative filtering is based on the analysis of the interaction history of a community of users. To produce recommendations, it is sufficient to look for the similarity between actions and human-machine interactions, from the point of view of preferences, choices and user behaviors, in order to produce relevant items adapted to the profile and needs of each user.

This article exposes a smart recommender system applied in an online learning environment in order to be able to provide personalized courses and guide students to select more suitable courses. For example, emailing or sending notifications through the user interface of distance learning platform, to students who follow courses in a specific field and recommend the suitable educational resources that are likely to be interesting for them. Also, learners can be guided to enroll in the latest courses in their interest areas based on historical data of all users over a large dataset of courses enrollments. In this article, we are interested in improving learning platforms through a recommender system. Our system uses association rules for extracting more interesting relationships between learners' behaviors. Especially to find similarities in courses enrollments in the transaction database. Thus, discovering association rules enables us to target students who learn two or more courses together, i.e. finding a list of frequent courses enrollments to determine those that are more likely chosen by the learners. So, based on the discovered patterns, we can guide students to take specific courses. The pedagogical team can also improve the quality of non-frequent courses or create new ones.

The rest of the article is organized as follows: In section 2, we present a state of the art of recommender system for e-learning environment. In section 3, we introduce the basic concepts of the association rules technique and then we give a detailed description of the FP-growth algorithm. In section 4, we implement the course recommender system using data from ESTenLigne platform and R Studio as an integrated development environment (IDE). For making the experimental results clear and easy to understand, we use arulesViz[4] package which provides a rich set of powerful data visualization techniques for association rules.

II. Related Work

Many research works have conducted in the field of distance learning in higher education using big data techniques and including data mining and machine learning methods. There are a lot of applications of these techniques. Particularly, recommendation engine which is used in several areas such as basket analysis (Amazon), social networks (linkedIn), government, education, etc.

Mihai Gabroveanu[5] proposed the prototype of a recommender system based on association rules for the distributed learning management system. The article uses distributed data mining algorithms and data obtained from Learning Management Systems (LMS) database in order to identify strong correlations between sets of courses followed by students. It also gives a brief description of the architecture and methodology of the course recommender system without providing an implementation of the proposed architecture.

Zhou et al.[6] implemented an alternating least squares (ALS) algorithm by utilizing the collaborative filtering approach for the Netflix Prize. ALS is a simple parallel algorithm which aims to tackle the scalability issue with very large datasets. It used for building a large-scale movies recommender system for predicting user ratings. The results achieved show a performance improvement of 5.91%.

Jiahui et al.[7] developed a large-scale recommender system in order to offer personalized news based on past click behavior of users. Their work combines both content-based recommendation and collaborative filtering methods to suggest most relevant news articles.The experiment results showed that the hybrid approach enhances the quality of news recommendations by attracting more readers to visit Google website news.

Many recommender systems for social networks have been developed using user interactions and behaviors. An approach is suggested in [8] to analyze the users' interest of Twitter platform. They combined sentiment analysis and classification of tweets by analyzing the topics discussed by the users. The implementation of their work gives encouraging results.

To help learners to find most proper pedagogical resources in computing environments for human learning dedicated to e-learning platform, we propose a smart courses recommender system using association rules model and the advanced packages of R programming language. Moreover, the system presented in this article uses many data visualization techniques which aim to provide more understanding of the hidden information over a large dataset of historical courses enrollments.

[1] ESTenLigne project is supported by the EST Network of Morocco and the Eomed association (http://www.eomed.org).

III. METHOD

In this section, we focus on the basic concepts of association rules used for developing the proposed recommender system. Also, we present the FP-growth algorithm which is considered as one of the most commonly used algorithms for extracting the frequent itemsets, without candidate generation, from historical learners' enrollments during the learning process.

A. Association Rules Method

Association rules is an unsupervised learning method that is widely used in many fields including recommendation engines, retail analysis of the transaction, and clickstream analysis across web pages[9]. It aims to discover hidden patterns in large amounts of data, in the form of interesting rules. In other words, this method consists in detecting associations between data stored in a giant database. It is a set of powerful exploratory techniques widely used in different sectors but also for scientific research purposes. The most popular application using the association rules is the one concerning the analysis of consumption habits. The power of the association rule method lies in its ability to extract hidden structures in a massive amount of data.

In the field of e-commerce, these techniques make it possible to know the relation between the choices of the customers, as the customers who bought the product A also purchased Product B or Product C most often. The term Association rules is often referred to as Market Basket Analysis application. Because the first time used was in 1993 by Agrawal , et al.[10] in order to find useful relationships between items through a large database of customer transactions. Each transaction consists of items purchased by a customer. In order to discover all significant connections between items bought by a customer over a period of time not necessarily consist of items bought together at the same time.

In general, the commendation systems consist of three principle steps; first, collect data from large transaction database; second, find similarities between users behavior's, according to more frequent item set, and finally, recommend more suitable items for users.

Considering $C = \{c_1, c_2, c_3, \dots, c_n\}$ a set of all items or courses enrollments stored in database and $L = \{l_1, l_2, l_3, \dots, l_n\}$ a set of learner profiles. Each learner l_i enrolls into k courses, where k is a subset of courses chosen from set of items C. In association rules, we define a rule as an implication of $X \Longrightarrow Y$, Where $X, Y \subseteq C$ (X and Y are sets of courses) and $X \cap Y = \emptyset$, which means when course X followed by the learner l_i, course Y is likely followed as well with a high probability. The set of attributes X is called antecedent or left-hand-side (LHS) of the rule; the set of attributes Y is called consequent or right-hand-side (RHS) of the rule[10].

Generally, The association rules technique produce a large number of rules $X \Longrightarrow Y$, but to select interesting rules from the set of all generated rules, there are two important measures to determine the quality of an association rule, the most known are minimum thresholds of support and confidence.

The support is the percentage (%) of transactions in the dataset that contain the itemset X while confidence is defined as the percentage (%) of transactions that contain X, which also contain Y. The formal definition of the confidence is: $\mathrm{conf}(X \Longrightarrow Y) = \frac{\mathrm{supp}(X \cup Y)}{\mathrm{supp}(X)}$. Therefore, a strength association rule $X \Longrightarrow Y$ should satisfy: $\mathrm{supp}(X \cup Y) \geq \sigma$ and $\mathrm{conf}(X \Longrightarrow Y) \geq \delta$, where σ and δ are the minimum support and minimum confidence, respectively.

In the context of our research, we apply association rules technique in the online learning. Accordingly, a transaction in our case is represented by the student's profile. Similarly, items are replaced by courses followed by a given student during the learning process. So, we can define the support and confidence respectively as follows:

$$supp(X \Longrightarrow Y) = \frac{number\ of\ learners\ following\ X\ and\ Y\ courses}{total\ number\ of\ learner\ enrollments\ in\ database}$$

$$conf(X \Longrightarrow Y) = \frac{number\ of\ learners\ following\ X\ and\ Y\ courses}{number\ of\ learners\ profiles\ following\ X\ courses}$$

Support is the first one which is defined as the percentage of transactions that contain X, It means, support is an indication of how frequently the itemset appears in the database. On the other hand, confidence is defined as the percentage (%) of transactions (students profiles) that follow X, which also follow Y.

B. FP-growth Algorithm

There are several algorithms for implementing association rules method to determine the most interesting relationships between variables or items in a large database, through finding frequent itemsets. FP-growth (Frequent Pattern Growth) [11] is an efficient and scalable algorithm for extract items that more likely appear together in a large transaction database. We find other algorithms such as Apriori, MAFIA (Maximal Frequent Itemset Algorithm) and Eclat. But FP-growth is the fastest one because it allows frequent itemset discovery without candidate item set generation which is more expensive in both memory and time. Moreover, candidate generation and test require multiple database scans. In fact, by using FP-growth the number of database scan is reduced to two. The first scan counts the support of each item; the infrequent items are deleted while the frequent items are sorted in decreasing support counts as a list of frequent items (L). And in the second scan, FP-growth constructs FP-tree. Those operations form the first step of the algorithm. On the other hand, the second step aims to extract frequent itemsets from the constructed FP-tree.

The FP-growth algorithm is based on the "divide and conquer" strategy of breaking down a problem into subproblems. First, it compresses frequent itemsets represented in the database using a compact data structure called FP-Tree (frequent-pattern tree) whose branches contain the possible item associations. Each association can be divided into fragments (pattern fragment) which constitute the frequent itemsets. The FP-growth method transforms the problem of finding the longest frequent itemset by searching for the smaller one and its concatenation with the corresponding suffix (the last frequent item in the branch leading to the item in question). This reduces the cost of research.

In general, an FP-tree tree is composed of a root is a number of nodes. A node is composed by the name of the item, the number of occurrences of that item, and an inter-node link to other occurrences of the same item in other transaction sequences. A header table points to the first occurrence of each item. The second step of the FP-growth algorithm is to extract frequent models using the FP-tree structure. The mechanism of extraction of association rules is described in detail in the article by Jiawei Han et al[11].

IV. COURSES RECOMMENDER SYSTEM

The construction mechanism of our recommendation system is very simple. It is based on the analysis of the history of the enrollment activities of learners in the courses. To do this, he needs all the items (courses) available in the ESTenLigne platform database to be able to generate a catalog of relevant courses while analyzing the learning trances of the learners. Fig. 1 gives a simplified diagram describing the course recommendation process.

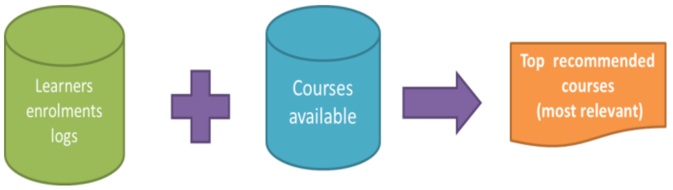

Fig. 1: Course recommendation mechanism

A. System Architecture

The proposed courses recommender system consists of two roles which are learners and teachers or pedagogical team. It has the following components: online learning platform and MySQL database server that stores and manages data produced by learner's interactions during the learning session including courses enrollments, learner's behavior, etc. There is also R environment to develop, execute, and test the association rule method. Furthermore, the architecture describes different steps including data discovery, modeling, processing, and data visualization in order to efficiently finding useful rules between all courses enrollments.

Our approach consists in finding hidden patterns from historical data of learners activities. So, the input of our system is courses enrollments of the students. Fig. 2 illustrates the global architecture of the implemented system. The historical data of courses enrollments are integrated into the R environment using RMySQL driver to connect with the database server. In general, our commendation system consists of the following steps.

- First, import data from ESTenLigne database;
- Second, the imported data are collected and prepared using several techniques
- Next, we choose the association rules technique as a model of our system.
- Then, we process courses enrollments data using the FP-growth algorithm.
- After, we can visualize the findings through the use of arulesViz[4] visualization techniques to have more understanding of the results.

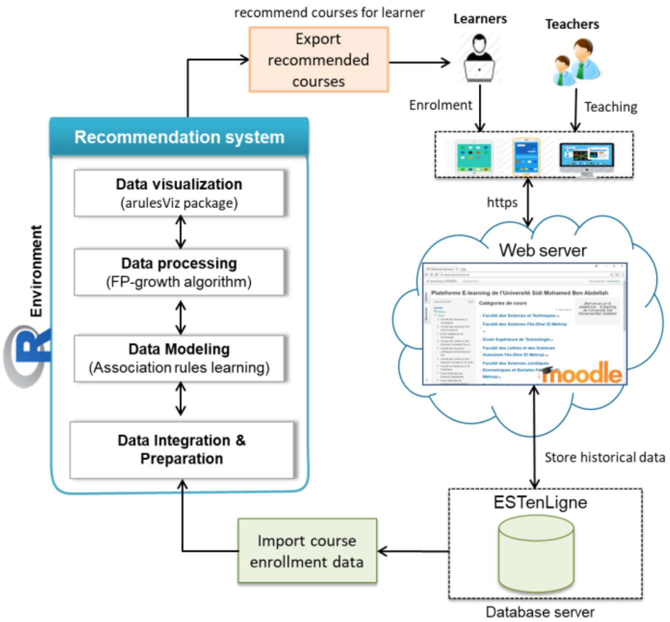

Fig. 2: Courses recommender system architecture

And the output is the catalog of recommended courses that match the learner interests. Then, the students can browse these recommended courses and start learning the relevant courses.

B. ESTenLigne Project

The present work is a part of the ESTenLigne project, which is the result of several years of experience for the development of e-learning in Sidi Mohamed Ben Abdellah University of Fez. It was started since 2012 by the EST network of Morocco, which aims the development of distance education based on new information and communication technologies through the implementation of open, adapted and free online learning platform, and taking into account the dimensions of exchange, sharing and mutualization of pedagogical resources[12], [13].

Several works have been done as part of this project including the training of experts across e-learning in the context of the Coselearn I project, the, and teacher training through Franco-Moroccan EST[2] and IUT[3] cooperation[13], [14]. Furthermore, there are some research's that have been done around this project such as the analysis of the use of educational resources where the objective was to analyze the use pedagogical resources in some courses namely the algorithmic course[15]. Also, a case study for collaboration analysis of online course based on activity theory[16].

However, in this platform, we have never worked on systems of recommendation, due to its important role as a support to orientate students. Especially, most of the learners in this platform are new graduates who have just come to integrate higher education and who need a system to help them to target the relevant courses to follow during their learning process.

In fact, the students have a lot of difficulties and are lost in the diversity of educational resources, particularly the large number of available courses. This requires the adaptation of the teaching to meet the needs of students. To solve these problems, we develop a course recommender system to promote learning to learners through creating a smart solution. It is able to generate the most appropriate courses automatically based on historical data of learner's activities.

C. Dataset structure

The ESTenLigne project is based on LMS Moodle [17]. Indeed, it is an open source learning management system. It uses a relational database which has around 250 tables. We focus only on student's enrollments into courses. Especially, we focus on four tables that represent the information we need to implement our recommender system based on association rules technique. The class diagram depicted in Fig. 3 shows the structure, the attributes, and the relationships between different entities in the database.

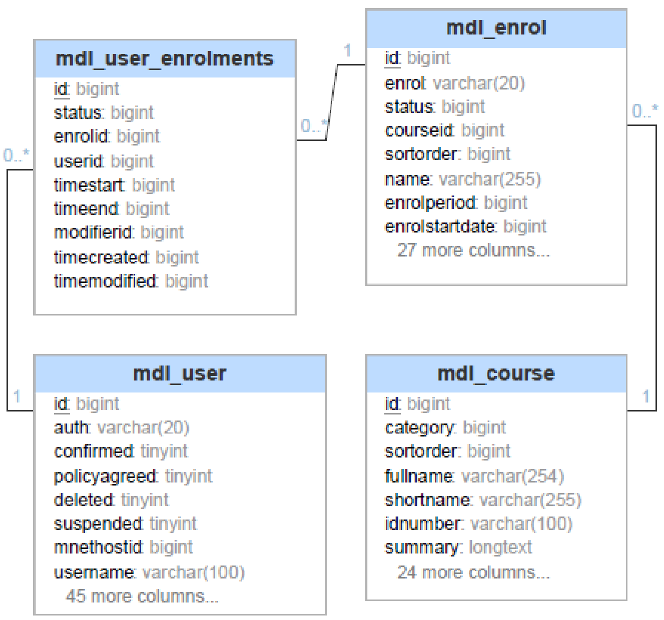

Fig. 3: Class diagram of ESTenLigne database for learner enrollment information

The class diagram, as seen in Fig. 3, describes four important tables for collecting data about historical data of learner's enrollments. First, there is mdl_user that gives information about the student's profile. Second, all user enrollments have recorded into mdl_user_enrollments table. Third, mdl_enrol table contains data of courses enrollments. For the same course, there can be different enrollment start and end dates. Some students may require a course for one period of time but other users may want to enroll in the same course for a different period of time. Fourth, mdl_course table stores all the details of the courses that are uploaded to the learning management system. It stores the names of the course, category, full name, short name, summary, time created, time modified etc.[18].

D. Flow chart of courses recommendation system

The Flow chart illustrated in Fig. 4, shows the dynamic aspects of our system. It describes a series of actions or activities performed by different actors.

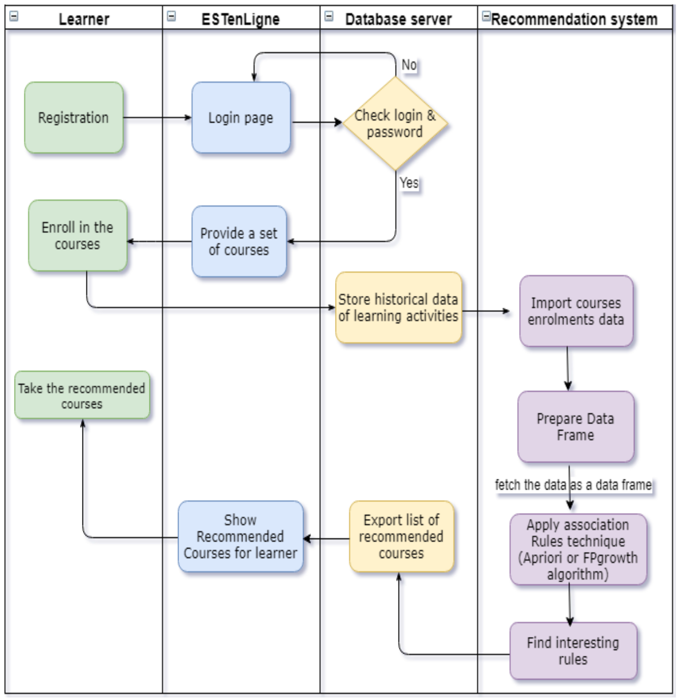

Fig. 4: Flow chart of courses recommendation system

In the beginning, learners should create an account through page registration. After the login into ESTenLigne platform, learners login name and password is verified. If it is valid then the student can access a set of courses available in the dataset. After browsing these courses, Learners will choose the courses in which he is interested. The data of learner enrollments, during the learning session, are stored in ESTenLigne database. Next, we collect appropriate data from historical data. Then, we prepare those data in the form of data frame. Indeed, the courses enrollments that belong to each learner are aggregated into a single vector as an array of enrollments. Afterward, we use R environment to apply Association rules techniques. The result FP-growth algorithm, according to support and confidence specified values, is a catalog of interesting correlations between courses. The next step consists of exporting the list of recommended courses to ESTenLigne database. Finally, we can display the result of recommendation system to the learner in order to guide and suggest them more relevant courses.

[3] http://www.est-usmba.ac.ma
[4] http://www.iut.fr

V. EXPERIMENTS & RESULT

A. Experiments

To evaluate the effectiveness of the courses recommender system presented in this article, we have conducted many experiments on 1218 students from the High School of Technology of Fez. 859 of them enroll into at least one course and 359 have never followed any course. So, we concentrate just on students who have course enrollments. The proposed system recommends the suitable courses for students among 153 courses proposed by teachers.

In order to test and validate the proposed system, we choose R Studio as an integrated development environment (IDE). It is a powerful big data tool for the advanced statistical analysis. It provides a rich set of high-level packages for statistical purpose. It also allows flexible graphical options that enable different data visualization techniques.

To prepare the required data, before launching the execution of our system, we collect data from ESTenLigne database. To do that, we develop a SQL query to extract the list of courses followed by all learners using database structure in Fig. 3. Indeed, we focus essentially on four tables includes: *mdl_user*, *mdl_user_enrolments*, *mdl_enrol* and *mdl_course*. To do this task efficiently, we integrated RMySQL package in R Studio to easily connect and execute SQL statements.

Then, we need to record courses by student id (user_id), so the individual courses followed by a given learner are aggregated into a single record as a vector of courses. Next, we convert the grouped data into an optimized object (S4) for running the FP-growth algorithm. After we execute SQL query though RMySQL tool and create the group of courses for each student, we obtain a collection of courses order by learner identifier (user_id) which represent the input data of FP-growth algorithm.

The next step consists of running the FP-growth algorithm on data captured in the data discovery phase. For this purpose, we use rCBA package [15] which implements the FP-growth algorithm based on the “Classification Based on Associations”. At first, we specify the minimum support and confidence threshold, respectively, to find more strong relationships between courses enrollments. The number of interesting association rules changes according to the value of support and confidence and the database size. In order to find more interesting roles, we use the minimum support threshold of 1% and we set 2% as a minimum confidence threshold. Finally, we can run the FP-growth algorithm as shown in Fig. 5.

```
> rCBA::fpgrowth(train, support=0.01, confidence=0.02, maxLength=2,
consequent="course",
+                          verbose = TRUE, parallel=FALSE)
2018-07-07 17:17:28 rCBA: initialized
2018-07-07 17:17:28 rCBA: data 2553x2
                took: 0  s
Jul 07, 2018 5:17:28 PM cz.jkuchar.rcba.fpg.FPGrowth run
INFO: FPG: start
Jul 07, 2018 5:17:28 PM cz.jkuchar.rcba.fpg.FPGrowth run
INFO: FPG: tree built (24)
2018-07-07 17:17:28 rCBA: rules 14
                took: 0.11  s
```

Fig. 5: The experimental results of FP-growth Algorithm

In order to present the items frequency visually, we use itemFrequencyPlot function of arules package. The relative distribution of the most frequent courses enrollments in our database transaction is illustrated in Fig. 6.

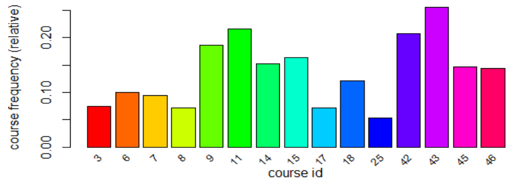

Fig.6: Relative frequency distribution of courses enrollments

B. RESULTS

The result of running FP-growth algorithm is a set of 14 rules which satisfy the minimum support and confidence thresholds. The top 10 useful rules ordered by the confidence measure are illustrated in Table 1.

TABLE 1: CONFIDENCE RESULTS

rule	lhs		rhs	support	confidence
[1]	{11,46}	=>	{45}	0.06519208	1.0000000
[2]	{11,45}	=>	{46}	0.06519208	0.9824561
[3]	{46}	=>	{45}	0.13736903	0.9593496
[4]	{15,46}	=>	{45}	0.05355064	0.9387755
[5]	{15,45}	=>	{46}	0.05355064	0.9387755
[6]	{45}	=>	{46}	0.13736903	0.9365079
[7]	{6,7}	=>	{18}	0.05355064	0.8518519
[8]	{6,18}	=>	{7}	0.05355064	0.8363636
[9]	{7}	=>	{18}	0.07566938	0.8024691
[10]	{7,18}	=>	{6}	0.05355064	0.7076923

According to the obtained results of support and confidence; as we can see from the experimental results in Table 1, the association between courses {11 and 46} and {45} has the highest confidence; also the association between courses {7 and 18} and {6} are the lowest. According to the more interesting rules extracted from transaction database and the calculated values of the confidence, it is clear which courses are more likely followed by learners and we can determine the suitable course to recommend for each learner. For example, the rule 1 {11, 46} => {45} has the highest confidence, so our system recommend course 45 to students who enroll into courses {11 and 46}. According to Table 1, the efficiency of rule 1 is 100% because there are 56 students who enroll in courses {11 and 46} where 56 among them enroll also into course {45}. For course rule 2, there are 57 students in historical data of learners enrollments who take both of courses {11 and 45}, 56 among them follow course {46} in subsequent courses, so the efficiency of rule 2 is 98%. So the system recommends the course {46} to students who enrolled in courses {11 and 45}. With regard to rule 3, there are 123 learners enroll into course {46}, because a high proportion

(118 learners) of them enroll also into course {45} in subsequent courses, the efficiency of rule 3 is 95%, so the system recommends the course {45} to students who enrolled in course {46}, and so on. For the top 10 rules, we notice that the values of confidence are between 0.707(70.7%) and 1.00 (100%), which prove that we have obtained good results. Thus, we can conclude that the proposed course recommender system provides the more appropriate courses according to the historical data of students' activities, especially courses enrollments.

C. Data visualization

In order to view more clearly the results of the conducted experiments, we use R-extension called arulesViz[4] package which is an advanced technique for visualizing the strong relationships between courses.

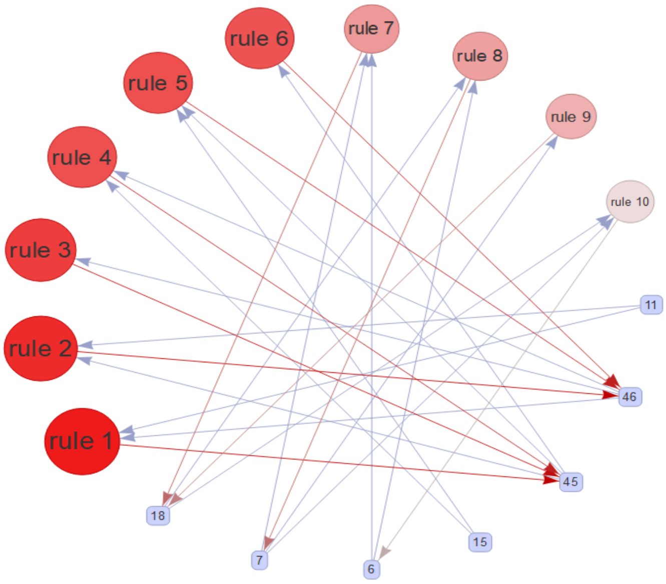

Fig. 7: Graph-based visualization with rules as vertices

Fig. 7 illustrates a very clearly the 10 interesting rules using graph-based techniques. Indeed, It visualizes association rules using vertices and edges where vertices typically represent item(course) or item set (set of courses) and edges indicate the relationship in rules. As we can see from rule 1 from the graph in Fig. 7, students who enroll in course {11, 46} enroll also into {45} course.

Another technique to represent the strong rules is grouped matrix (Fig. 8) that is capable to analyze large rule sets[4]. Indeed, this technique gives us the possibility to select and zoom the interesting relationship between courses.

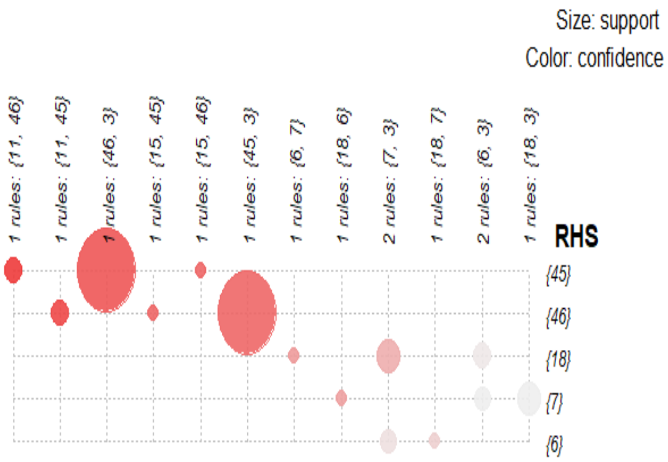

Fig. 8: Grouped matrix-based visualization

To visualize the grouped matrix we use a balloon plot with antecedent groups as columns and consequents as rows. The color of each balloon represents the confidence measure in the group and the size of the balloon shows the support. Furthermore, the columns and rows in the plot are reordered such that the quality of rules is decreasing from the top to down and from left to right, directing the user to the most interesting group in the top left corner.

VI. Conclusion & Future work

In this work, we have developed a smart recommender system for helping students' to take more relevant courses according to the preferences and needs of each learning situation. It aims to recommend a catalog of the most suitable and convenient pedagogical resources to the learners based on finding similarities between learners enrolments. In order to validate the effectiveness of our solution, we have used the past activities of learners collected form e-learning system of High School of Technology of Fez. To develop our solution we have employed the association rules model, FP-growth algorithm and RStudio for data integration, modeling, processing, and visualization through arulesViz[4] package. The experiences of the recommender system give encouraging results. Moreover, The presented recommendation engine can be applied and easily integrated into other learning environments such as MOOC (Massive Open Online courses). The future work aims to improve the performances and efficiency of this work by employing it in a large-scale environment utilizing the distributed storage and the parallel processing of Hadoop ecosystem and the advanced analysis of machine learning algorithms provided by Apache Spark Framework.

References

[1] K. Dahdouh, A. Dakak, et L. Oughdir, « Integration of the Cloud Environment in E-Learning Systems », Trans. Mach. Learn. Artif. Intell., vol. 5, no 4, sept. 2017.

[2] F. O. Isinkaye, Y. O. Folajimi, et B. A. Ojokoh, « Recommendation systems: Principles, methods and evaluation », Egypt. Inform. J., vol. 16, no 3, p. 261-273, nov. 2015.

[3] « Amazon.com: Online Shopping for Electronics, Apparel, Computers, Books, DVDs & more ». [En ligne]. Disponible sur: https://www.amazon.com/. [Consulté le: 30-sept-2018].

[4] M. Hahsler et S. Chelluboina, « Visualizing Association Rules: Introduction to the R-extension Package arulesViz ». 2018.

[5] G. Mihai, « Recommendation System Based On Association Rules For Distributed E-Learning Management Systems », ACTA Univ. Cibiniensis, vol. 67, no 1, janv. 2015.

[6] Y. Zhou, D. Wilkinson, R. Schreiber, et R. Pan, « Large-Scale Parallel Collaborative Filtering for the Netflix Prize », in Algorithmic Aspects in Information and Management, vol. 5034, R. Fleischer et J. Xu, Éd. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, p. 337-348.

[7] J. Liu, P. Dolan, et E. R. Pedersen, « Personalized news recommendation based on click behavior », in Proceedings of the 15th international conference on Intelligent user interfaces - IUI '10, Hong Kong, China, 2010, p. 31.

[8] N. Mangal, R. Niyogi, et A. Milani, « Analysis of Users' Interest Based on Tweets », in Computational Science and Its Applications – ICCSA 2016, vol. 9790, O. Gervasi, B. Murgante, S. Misra, A. M. A. C. Rocha, C. M. Torre, D. Taniar, B. O. Apduhan, E. Stankova, et S. Wang, Éd. Cham: Springer International Publishing, 2016, p. 12-23.

[9] D. T. Larose et C. D. Larose, Data mining and predictive analytics, Second edition. Hoboken, New Jersey: John Wiley & Sons Inc, 2015.

[10] R. Agrawal, T. Imielinski, et A. Swami, « Mining Association Rules between Sets of Items in Large Databases », Proceedings of the 1993 ACM SIGMOD international conference on Management of data, p. 207-216, 1993.

[11] J. Han, J. Pei, et Y. Yin, « Mining Frequent Patterns without Candidate Generation », Int. Conf. Manag. Data Proc. ACM SIGMOD, p. 1-12, 2000.

[12] A. IBRIZ, « Une Démarche Innovante de Conduite de Projet Elearning: C.D.I.O ». 2ème Congrès International du Génie Industriel et du Management des Systèmes (CIGIMS), 2015.

[13] A. IBRIZ, et A. SAFOUANE, « L'Innovation Pédagogique dans les EST du Maroc : Le Model et la Conduite d'un cas réussi à travers le Projet ESTenLigne ». 2014.

[14] L. Oughdir, A. Ibriz, et M. Harti, « Modélisation de l'apprenant dans le cadre d'un environnement d'apprentissage en ligne ». TELECO2011 & 7ème JFMMA Mars 16-18, 2011 – Tanger MAROC, 2011.

[15] M. BENSLIMANE, K. OUAZZANI, M. TMIMI, et M. BERRADA, « Proposal of an Approach of Online Course Design and Implementation: A Case Study of an Algorithmic Course », vol. 7, p. 7, 2016.

[16] A. Ibriz, M. Benslimane, et K. Ouazzani, « Didactics in Online Learning Technical Courses: A Case Study Based on Activity Theory », Int. J. Comput. Sci. Inf. Technol., vol. 7, p. 849-854, 2016.

[17] « Moodle - Open-source learning platform | Moodle.org ». [En ligne]. Disponible sur: https://moodle.org/. [Consulté le: 03-avr-2018].

[18] A. GARG, « SCORM Based Learning Management System For Online Training ». KANSAS STATE UNIVERSITY Manhattan, Kansas, 2012.

[19] J. Kuchar et T. Kliegr, « rCBA: CBA Classifier for R ». https://cran.r-project.org/web/packages/rCBA/index.html, 16-mai-2018.