



An Innovative Approach to Prevent Learners' Dropout from MOOCs using Optimal Personalized Learning Paths: An Online Learning Case Study

El Miloud Smaili, Chaimaa Khoudda, Soukaina Sraidi, Salma Azzouzi, My El Hassan Charaf*

Laboratory of Research in Informatics, Faculty of Sciences, Ibn Tofail University, Morocco

Abstract Due to the rapid evolution of information technology, distance education has undergone a sustained expansion. Within the scope of open distant learning, MOOCs (Massive Open Online Courses) tend to open up access to education to all by bridging geographical and economic barriers. However, the evolution of this learning mode is facing some challenges such as the low completion rate of the courses as well as the heterogeneity of the learners' profiles. In this paper, we aim to personalize the MOOC contents for each learner in order to improve their academic performance and to enhance the online platforms efficiency. The idea is to build a system that takes into account the heterogeneity of learners profiles and offers each learner a path adapted to their needs through the exploitation of their interactions with the learning environment. To this end, we suggest using the PSO method "Particle Swarm Optimization" in order to construct the optimal choices of learning paths in the system. Furthermore, we conduct an online learning case study to show the effectiveness of the proposed approach.

Keywords Adaptive Learning, MOOC, Particle Swarm Optimization, Drop out

AMS 2010 subject classifications 68T05, 97U50, 62H30

DOI: 10.19139/soic-2310-5070-1206

1. Introduction

Nowadays, Higher education is experiencing several challenges, including quality, students' profiles variety and equitable accessibility to courses. A wide range of innovative approaches are emerging to address these challenges by providing online learning solutions. Massive Open Online Courses (MOOCs) is one of such popular online solutions as it offers a flexible way to learn courses online in the form of videos, written courses, as well as exercises and interactive discussion spaces. Moreover, these MOOCs are accessible to people with an active Internet connection and who desire to learn without a need to be enrolled in university. Consequently, this has revolutionized higher education as everyone could access to education regardless of their standard of living or place of residence and in most cases the lessons are free with possibility to obtain a diploma at the end of the training. Nevertheless, the evolution of MOOCs presents some challenges, although there are recognized opportunities and growing interest in using this mode of learning. One of the main problems is the high dropout rate and, consequently, the low certification rate, which does not exceed 10% despite the extremely high number of registrants [1]. This dropout rate is commonly a result of inadequate tools and methods used to adapt learning activities and to motivate learners in these environments. Another problem is related to the diversity of learners' profiles with different learning styles and knowledge levels [2]. This leads to the necessity to personalize the way of contents delivering in such learning environments. Additionally, the selection of appropriate parameters to consider while defining the learner's profile is not easy since a course may be taken by many learners with different profiles from different regions of the world. To this end, it is important to handle the multiplicity of learner profiles by developing content that meet the needs

*Correspondence to: El Miloud Smaili (Email: miloud.smaili@uit.ac.ma). Laboratory of Research in Informatics, Faculty of Sciences, Ibn Tofail University, Kenitra, Morocco (14000).

and objectives of each learner in MOOCS. Besides, learners usually use digital tools (especially social media) to ease interactions between them and their teachers. Therefore, this will allow researchers to leverage the traces generated during their interaction with these digital systems in order to analyze their behaviors. In this context, the rise of big data makes it possible to record and interpret individual student characteristics and their real-time status in all aspects of learning. Moreover, data mining and learning analytics can support more effective use of recorded learning data to assess learning processes, predict future performance, and identify potential problems. For this purpose, one solution to overcome both the high dropout rate and the heterogeneity of learner profiles may be the use of adaptive learning [3]. As mentioned earlier, MOOCs are a way to serve a large number of learners, and consequently to serve a large heterogeneity of profiles on various online platforms. Personalization can play a crucial role in this process. Therefore, adaptive MOOCs apply adaptive features to present personalized learning experiences based on course data collection. The idea is to build a system that offers each learner a path adapted to their needs through the exploitation of their interactions with the learning environment. In this context, appropriate learning paths should be provided to learners based on their actual learning demands and their knowledge levels. Generally, the adaptive e-learning model should integrate and enhance existing learning frameworks to provide a set of learning paths that can meet the different knowledge levels of learners. Based on classification and swarm intelligence algorithms, an adaptive learning model can satisfy learners' demands by building a system that allows to find the best learning path. The method as designed is similar to the path-finding approach used by ants for foraging, for example. While the classification method is used to determine some learning and search parameters. By doing so, we ensure a better follow-up as we generate the learning path suggestions adapted to the skill profile and behavior of each participant. Our solution consists in personalizing the MOOC contents for each learner. Thus, we propose a new method to optimize courses selection by applying the appropriate personalization strategy based on classification and optimization algorithms. The proposed solution aims also at improving the retention rate in MOOCs by identifying learners at risk. Compared to other literature proposals, the solution as described is automatic and does not require pedagogical team intervention which eliminates considerably the observed complexity in other adaptive systems. The paper is structured as follows: The second section provides some preliminaries. In section 3, we present the problematic statement. Then, we introduce the related work of this study in section 4. The section 5 gives an overview of our proposed learning model. In the section 6 we explain the implementation of the proposed solution using a case study. Finally, the section 7 gives some conclusions and identifies future works.

2. PRELIMINARIES

2.1. MASSIVE OPEN ONLINE COURSES (MOOC)

The MOOC acronym (Massive Open Online Courses) refers to a means of training or distance learning, over the Internet, in free and open access. The MOOC is a phenomenon relatively new that offers a combination of learning opportunities and open education for people outside of higher education to participate actively in their own learning [4]. This emerging field of learning facilitates the dissemination of knowledge across multiple networks so that educational experiences are made available to the public across different learning structures and educational applications. Therefore, these learning approaches reinforce the social inclusion of people in education, disseminate formal and informal knowledge to a large number of people and enable teachers to innovate their pedagogical skills.

2.2. ADAPTIVE LEARNING

Adaptive learning is a pedagogical concept that aims at adapting pedagogical solutions to the particular skills and needs of each learner by providing personalized content and learning path according to learners' individual needs and abilities. Due to the heterogeneity of learners' profiles, these adaptive learning systems promise to take into consideration the learner's profile (knowledge, preferences, skills and objectives) to build a unique and adapted learning path. In other words, Adaptive Learning is defined as "the adaptation of pedagogy, programs and learning environments to meet the needs and learning style of each learner". Basically, the Adaptive Learning aims to change the classic teaching-centered perspective of the teacher into a learner-centered perspective [5]. In practical terms,

we all learn differently (memory capacity, preferences, rhythms...) and for an ideal learning experience, we need a tutor behind each learner, who takes into account all the characteristics of the learner. The objective of adaptive learning is to optimize learning for each learner by taking into account the diversity of learners' background.

2.3. CLASSIFICATION AND EVALUATION METRICS

Classification is a supervised learning approach used for categorizing some unknown data into a distinct or discrete set of groups.

Classification Algorithms: Basically, the classification algorithms learn from the input variable(s) to predict the output variable which is categorical. Therefore, the most popular algorithms are:

- Support Vector Machines: SVM determines classes by locating a line or hyperplane. The SVM works such that it maps the data into a higher dimensional space. Therefore, the data features can be classified despite the fact that the data are not separable.
- K Nearest Neighbors: is an algorithm that uses a group of labeled points to label other points. The algorithm can be implemented by choosing a K value, computing the distance between the new point and each of the other points in the data set, and finding then the observations in the data set that are closest to the unknown point.
- Logistic Regression: is a statistical and machine learning model that estimates the target variable by classifying the dataset based on the values of the input fields

Metrics: Generally, we compare the values predicted by our model with the actual values in the validation set and see how accurate it is. There are many evaluation metrics such as:

- The Jaccard index is one of the easiest accuracy measurements and may be defined as the size of the intersection divided by the size of the union of the label sets that are finite. If the sample strictly corresponds to the original set, the accuracy is 1.0.
- Log loss, also known as the logarithmic loss is the measure of the performance of the algorithm where the predicted output ranges between 0 and 1. It computes the log loss of each row by measuring how far the prediction is from each label and then averages all the rows. When the model is more perfect, the value of the log loss is lower.
- F1-score: is the harmonic average between the precision and the recall, precision is seen as the measurement quality and recall is seen as the measurement quantity. Therefore, the best value is 1 and the worst is 0.

2.4. SWARM INTELLIGENCE

Swarm intelligence is a form of artificial intelligence (AI) inspired by the behavior of social insects swarming and flocking. The main objective is to check the search space in order to find the more convenient solution. Basically, swarm intelligence describes how ants form perfect trails or how birds flock for example in the nature. In the world of AI, swarm systems draw input from individual people or machine sensors and then use algorithms to optimize the overall performance of the group or system in real time. We mentioned here the most swarm intelligence techniques: Particle Swarm Optimization (PSO) and Ant Colony Optimization (ACO). PSO is a stochastic global optimization method based on the social behavior (bird flocking or fish schooling) and intelligence of swarm searching for the global optimal whereas ACO is based on ants' behavior looking for a way between their state and a source of nourishment [6].

3. PROBLEMATIC STATEMENT

The spread of MOOCs has gained recently a great popularity owing to the rapid evolution of learners enrolled in these courses. The massive and open character of MOOCs makes it possible to be followed by learners with heterogeneous profiles, expectations and learning habits. However, a major problem arises since the number of learners who complete the courses remains between 5% and 15%. In fact, the learners drop out of the MOOC

prematurely either due to lack of motivation, lack of technical skills to complete the course, or sometimes due to lack of support from the pedagogical staff. The majority of MOOCs offer the same curriculum to learners despite their variability. This is considered a major drawback for these learning systems since the same program may not be suitable for all learners, resulting then in a low retention rate. To this end, we suggest in this article to apply personalization techniques to adapt the MOOCS contents and to suggest for learners the best learning paths according to their specificities.

4. LITERATURE REVIEW

Nowadays, a wide range of works has been devoted to study adaptive learning for e-learning platforms. In this context, the authors in [7] present their approach and the architecture they adopted for the development of an adaptive e-learning platform. The proposed system generates learning paths adapted to the learners' profiles according to the pedagogical objectives fixed previously. Furthermore, the aim of the study [8] is to recommend a learning path which corresponds to the students' cognitive capacities from a learning object ontology. The method used is Hybrid Particle Swarm Optimization (HPSO) which integrates both Binary Particle Swarm Optimization (BPSO) and Discrete Particle Swarm Optimization (DPSO). The authors in [9] present a MOOC-based open educational resources (OER) that can be plugged into an Online Learning Environments (OLE) to dynamically deliver OER recommendations to learners based on their profiles on the MOOC. The study [10] aims to achieve personalized adaptive learning according to the needs and abilities of each learner by creating a personalized adaptive learning behavior model and a personalized MOOC platform based on such model. Another work [11], implements the particle swarm optimization and adaptive optimization of social network recommendation using gradient algorithm to realize swarm intelligence recommendation of social network user information. The solution presented in [2] solves the problem related to content personalization of MOOC for each learner. To this end, the authors choose to optimize the selection of the personalization parameters then they apply the appropriate personalization strategy, based on a classification algorithm. Furthermore, the authors in [12] suggest a way to support user privacy, trust and engagement in the context of the Learning and Performance Support (LPSS) program at the National Research Council of Canada. On the other hand, the authors in [13] provides an overview of adaptive learning functionality developed by Harvard and Microsoft in collaboration with edX and other partners. The solution suggested in [1] called Adaptive Recommendation for MOOC (ARM) aimed at solving the problem of dropouts caused by low satisfaction and feelings of loneliness. In addition, the work [14] involves integrating frameworks for intelligent tutoring systems (ITS) as part of a hands-on approach to tailor MOOCs to learners using the Learning Tools Interoperability standard (LTI). The authors in [15] proposes a learner model for adaptable e-learning model. The proposed model is based on the findings of a survey conducted to investigate the profiles and preferences of the local learners. The main idea in [16] is to propose a PSO algorithm with an adaptive learning strategy (PSO-ALS) to increase the population diversity and strengthen the global search ability in PSO. We can also learn from the study in [17] the concept of an adaptive open course from the perspective of discipline adapting knowledge, including sociality, semantic aggregation, personalization, openness of learning data and adaptive mechanism. Besides, the paper [18] investigates the MOOCs system by reviewing the available literatures and suggesting a framework that consider a list of recommendation of instructional material using the learner's profile and experience. On the other side, the work [19] proposes an approach of recommending micro-learning path based on improved ant colony optimization algorithm. Whereas, the subject of the paper [20] is to organize and analyze the parameter of adaptation in learning path; and to discuss the challenges in implementing learning path adaptation. A new learner model is proposed in [21] including four features: basic information, learning style, knowledge state, and cognitive ability. The paper gives a formal representation of these feature elements and a method for initializing and updating the values. Finally, this paper is a continuity of previous works [3, 22, 23, 24, 25] where we provide some solutions to enhance adaptive learning systems using some optimization algorithms. Unlike other approaches proposed in the literature, the solution tracks the evolution of learners in real time and updates their profiles accordingly, which presents a serious challenge to existing research on adaptive learning. The solution as described is automatic and does not require pedagogical team intervention

thereby eliminating the complexity observed in other adaptive systems. Moreover, the proposed system has the advantage of offering two complementary and coherent solutions, a preventive solution allowing to anticipate the dropout by identifying in a systematic and permanent way the learners at risk, the other solution allows to optimize and to personalize the courses according to the learners' real needs. We aim also to analyze the constraints and difficulties encountered in the courses from the metadata provided in the MOOC platforms in order to prevent these problems and improving the quality of MOOCs.

5. ADAPTIVE LEARNING'S: MODEL

5.1. MODEL DESCRIPTION

This article proposes a new adaptive MOOCs system in order to face the growing problem of dropping out in such a mode of teaching. The goal is to apply some personalization techniques to adapt the MOOCs' contents to the specificities of each learner, taking into account the expectations, motivations, learning styles, habits and also the learners' needs. The underlying idea is to gather traces left by learners' interactions within their learning environment. Based on these data, we extract all relevant information related to the learners' profiles since the learners' first contact with the system until the end of the course. Furthermore, the generated profiles allow us to determine automatically the recommended activities for each learner and to suggest for each learner the best path to follow by exploiting the power of the particle swarm optimization (PSO) algorithm. More precisely, the particles in such method are randomly dispersed in a research space. These particles represent the learners in our case, and as soon as a learner locates a learning material, the learner will be then followed by other persons. The method is based on the collaboration between individuals to converge towards the optimal global solution. In this context, the learner most closely to the optimum communicates the information to the others at each step so that they can modify their choice by taking into account their profiles' characteristics.

5.2. PROCEDURE

We describe in the figure below the procedure of our adaptive learning. The objective is to dynamically adapt the course to the preferences of each learner, using the particle swarm optimization algorithm (PSO). Indeed, the architecture of the adaptive learning system will be implemented in a set of steps as detailed below:

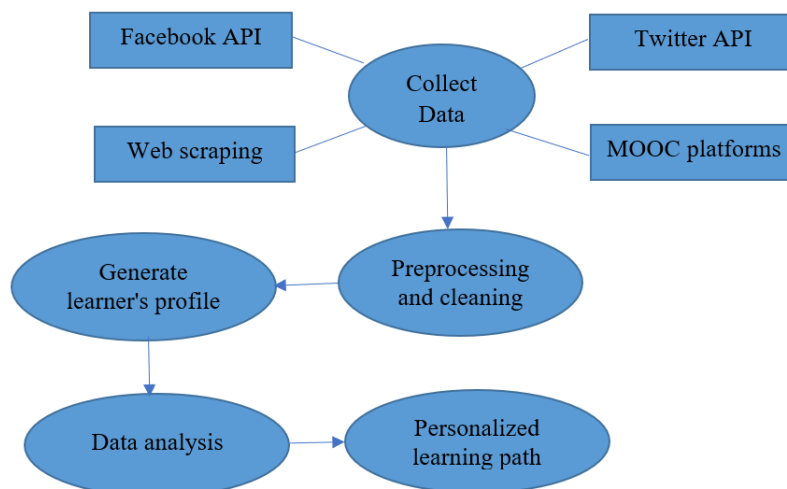


Figure 1. Our Adaptive learning Model

Data Collection: In order to implement our approach to adapt the courses content for learners, we need information identifying the learners' profiles. Several techniques will be used to collect information about each learner to ensure that a wide range of data is available to produce desired results:

- **The MOOC platform:** The first data source to be used will be the MOOC platforms and their associated forums, as they are a rich place for exchange and participation where discussion is focused on the course or part of it, allowing us to better understand the learner's need.
- **Facebook:** Generally, before the start of the course, the learners create groups of sharing, communication and collaboration between them, these spaces contain a huge amount of data useful for our research. Two significant ways exist to explore these media: Facebook API and Web scraping. For our case, we use web browser extensions that automatically detect comments sections to scrape them.
- **Twitter:** is the second most used social media platform in the world behind Facebook. Twint is an advanced Twitter scraping tool written in Python that allows Tweets scraping from Twitter profiles. In addition to the API, Twint allows us to collect every single detail on Twitter. Afterwards, the collected data can have several types of errors such as typing errors, missing information, inaccuracies etc. The improper part of the processed data can be then replaced, modified or deleted.

Data Preprocessing: The preprocessing of textual information is an essential step insofar as it improves the quality of the collected data. Therefore, the objective is to remove the noises and to transform the data into an understandable and readable format. In fact, mostly real-world data is composed of:

- **Missing data:** There are many reasons for missing data such as: Data is not continuously collected, a mistake in data entry or other technical problems and in order to deal with such issue, we can fill in missing value manually or using computed values or even ignoring the missing record.
- **Noisy data:** The reasons could be technical problems as well as a human mistake during data entry. In this case, we can use data binning also known as local smoothing, or linear/multiple linear regressions.
- **Inconsistent data:** The presence of inconsistencies is due for example to duplication within data or violation of data constraints. Then, the data is managed using external references and knowledge engineering tools.

Learner's profile Generation: After the text data preprocessing phase, it becomes easier to extract information from the text which will facilitate the generation of the learners' profiles using these characteristics, namely objectives, preferences, level of knowledge, learning styles and academic motivations. In this case, we can use some classification algorithms for categorizing learners' data into a distinct set of profiles. Generally, these classification algorithms learn from the input variable(s) to predict the output variable which is categorical. The most popular algorithms are: SVM, K-NN and Regression Logistics.

Data Analysis and Learning Path Personalization: Once the profile is created, the profile is then analyzed using an adaptive learning approach based on particle swarm optimization algorithm that supports the organization of the learning material. In fact, each particle in the population of PSO represents a possible solution of the optimization problem with two vectors (velocity and position) [5]. In our case, each learner i - in a research space of dimension D of the swarm- is modeled by its position vector \vec{x}_i and by its speed vector \vec{v}_i . Therefore, the velocity vector and the position vector of each learner can be updated during the optimization process using the following mathematical formulas [26]:

$$\begin{cases} v_{i,j}^{k+1} = v_{i,j}^k + c_1 r_1 (x_{best_{i,j}}^k - x_{i,j}^k) + c_2 r_2 (x_{gbest_i}^k - x_{i,j}^k) \\ x_{i,j}^{k+1} = x_{i,j}^k + v_{i,j}^{k+1} \end{cases} \quad (1)$$

Where: $x_{i,j}^k$ and $v_{i,j}^{k+1}$ are the j^{th} component of the i^{th} particle's position and velocity vector, respectively, in the k^{th} iteration; r_1 and r_2 are two random numbers uniformly distributed in the range (1,0); x_{best_i} and x_{gbest} indicate the best positions experienced so far by the i^{th} particle and the whole swarm, respectively; and c_1 and c_2 are two parameters representing the particle's confidence in itself (cognition) and in the swarm (social behavior), respectively. These two parameters are among the most important parameters of the algorithm in that they control the balance between exploration and exploration tendencies. A relatively high value of c_1 will encourage the

particles to move toward their local best experiences, while higher values of c_2 will result in faster convergence to the global best position. Therefore, the courses will be customized automatically after each profile update to fulfill the learners' expectations and needs.

5.3. APPLICATION PROTOTYPE

We describe in this section our adaptive learning prototype. The system contains three modules: (1) learner module, (2) learning environment module, and (3) Decision Making module. We give in the figure (Fig. 2) an overview of our architecture as follows:

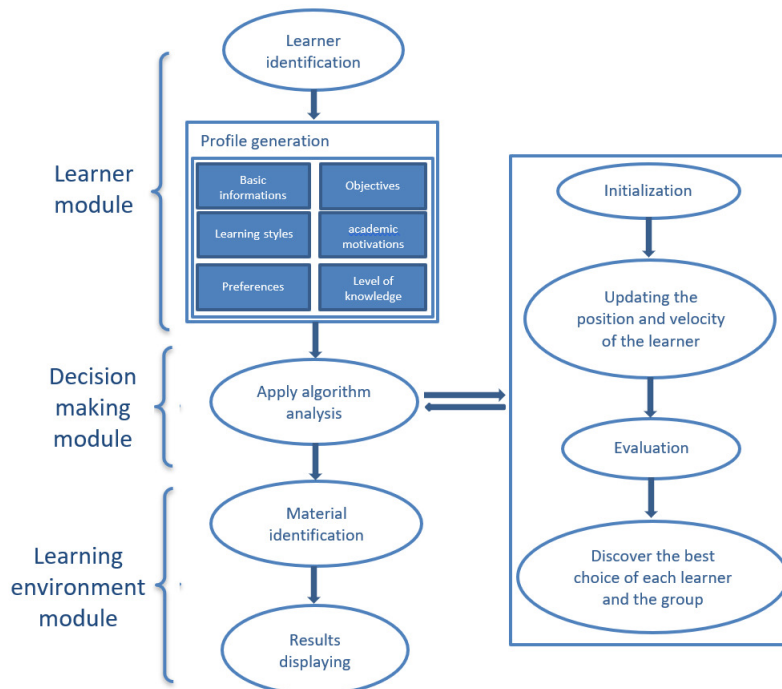


Figure 2. Architecture of our Adaptive learning model

Learning environment module: provides learning objects corresponding to different learners' needs. A learning environment -that promotes the development of students' skills and enables them to carry out their work - has a positive effect on their motivation. Therefore, materials and activities should be conducted in a way that stimulates and encourages learning and learner development.

Learner module: Learners' profiles are identified on the basis of information gathered upon their interactions with the system, even before the learning process begins. These profiles typically reflect learners' characteristics, such as goals, preferences, level of knowledge, learning styles and academic motivations. In this context, the "**Learner module**" explores the traces left by learners throughout their interactions with the learning environment. The information describing the learners' profiles is then gathered, preprocessed and stored in our database.

Decision-Making module: is responsible to make the correct choice of learning materials that meet the profile of each learner. In other words, the module chooses in what order to select the activities that the learner will engage in, and decides whether the learner is able to take the next learning steps. Based on the learners' profiles, the system will automatically detect the recommended courses as well as the path to follow for each learner. The profile in this case plays a crucial role to apply the PSO algorithm.

6. ADAPTIVE LEARNING'S OPTIMIZATION: CASE STUDY

In this article, we address the drop-out issue of learners from the MOOC programs. Therefore, we provide a solution to increase the retention rate by customizing courses for each learner. To this end, we firstly identify learners at risk to drop out from the program using the logistic regression algorithm. Then, we group those students into profiles according to their skills as described in our dataset. Finally, we use the PSO algorithm to suggest for each learner the suitable courses according to the average score attributed by the neighboring learners in the corresponding profile.

6.1. IDENTIFICATION OF LEARNERS AT RISK

Dataset Description: To illustrate our approach, we extract a dataset of MITx and Harvardx courses from the EdX platform. The dataset contains the following attributes:

[Institute, course_id, title, skills, year, semester, userid_DI, registered, viewed, explored, certified, final_cc_cname_DI, LoE, age, gender, grade, start_time_DI, last_event_DI, nevents, ndays_act, nplay_video, nchapters, nforum_posts]

In this case study, we focus on the following attributes:

- **viewed:** 0/1; anyone who has accessed the "Courses" tab of the edX platform.
- **explored:** 0/1; anyone who has completed at least half of the chapters in the tutorial.
- **LoE:** level of education. Possible values: "Less than Secondary," "Secondary," "Bachelor" "Master" and "Doctorate"
- **gender:** Possible values: m (male), f (female) et o (other).
- **nevents:** number of interactions with the course, empty if no interaction after registration.
- **ndays_act:** number of days the learner interacted with the course.
- **nchapters:** number of chapters (in the tutorial) accessed by the student.
- **nforum_posts:** number of posts on the forum of discussion.
- **certified:** 0/1; anyone who has obtained a certificate. Certificates are based on course grades and, depending on the course, the threshold for a certificate varies from 50% to 80%.

The average of certified learners in our dataset is 3.5%:

- **0:** 0.964288
- **1:** 0.035712

Data Preprocessing: Since the quality of data affects directly the efficiency to train our model, it is important to proceed on data preprocessing. Data preprocessing comprises several steps including mainly: data cleaning, encoding feature selection and Data normalization.

- **Data cleaning:** We start the cleaning process to overcome issues such as: data noise, missing values and incorrect data format.
- **Encoding:** associates non-numerical categories of a variable with a unique decimal value. We have 2 categorical variables in our dataset: 'LoE_DI' and 'gender'. These variables have been encoded as follows: code = {'Secondary':1,'Bachelor':2,'Master':3,'Doctorate':4,'Less than Secondary':5, 'm':1, 'f':2, 'o':3}.
- **Feature selection:** Before proceeding the next step (Data normalization), we start by analyzing the relationship between our target (certification) and different variables (features) in our Dataset. For this purpose, we used the distribution diagram separating certified learners from non-certified ones. Therefore, the obtained results are very interesting and show the relation between our target and some variables such as:
explored, nevents, ndays_act, nchapters, nplay_video, gender,LoE (level of education) and grade. We give below some examples:

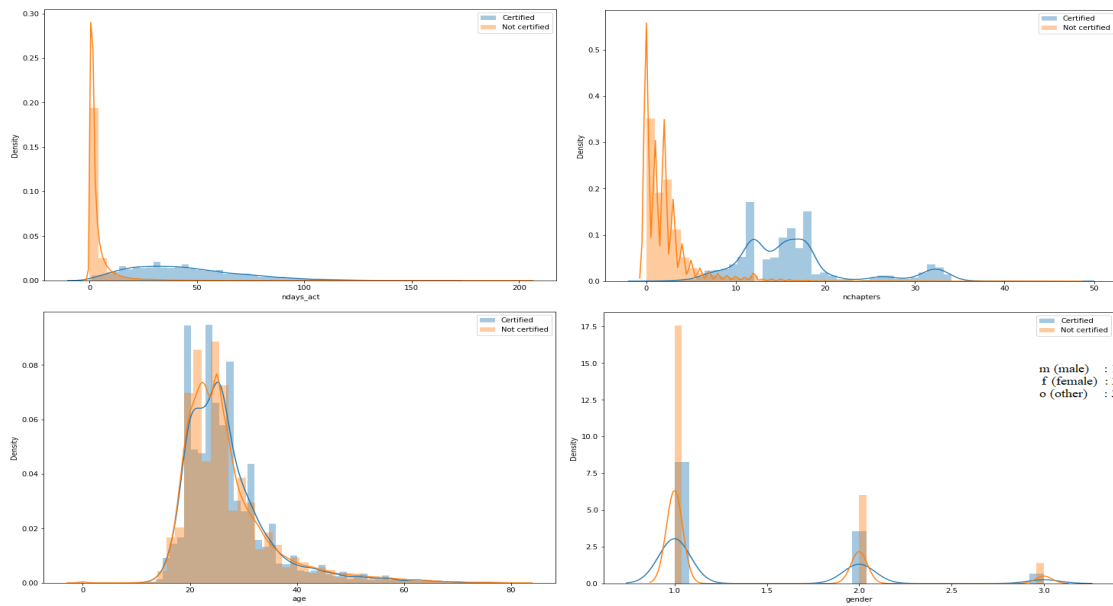


Figure 3. Relationship between our target (certification) and different variables (features) in our Dataset

Furthermore, it is crucial to investigate the dependency between variables in our dataset. In fact, since the correlation value is close to 1.0, this means that there is a perfect positive relationship between the two variables.

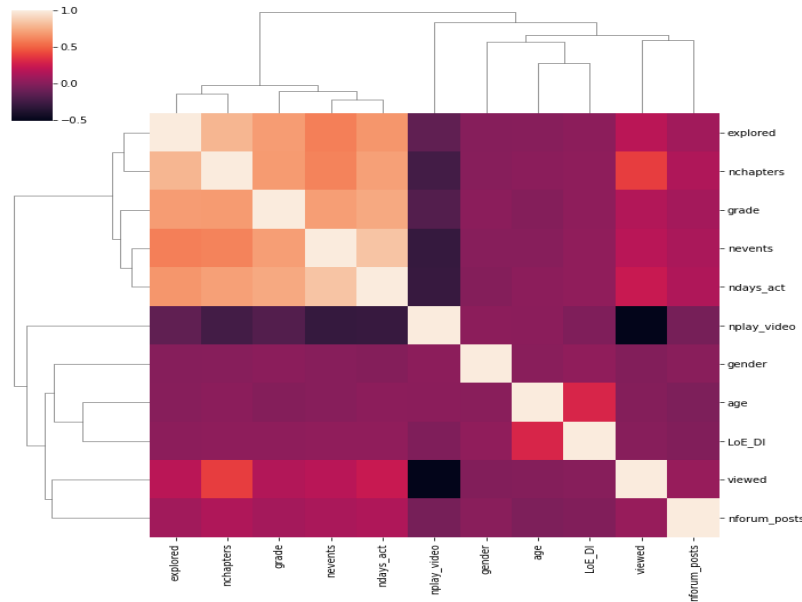


Figure 4. Features dependency in our Dataset

Therefore, the selected features of our model are: **explored, nevents, ndays_act, nchapters and grade.**

- **Data normalization:** At this stage, it is essential to normalize our quantitative data in order to put them all on the same scale. For this purpose, one of the most used normalization techniques is standardization. The

idea behind standardization is to transform data so that its distribution will have a mean value of 0 (zero) and a standard deviation of 1.

To do this, we need to subtract each x-value from the initial mean (μ) of our variable and deviate it by the initial standard deviation (σ) of our variable as follow:

$$X_{scaled} = \frac{X - \mu}{\sigma} \quad (2)$$

Prediction of learners At-risk of MOOC Drop-out: The objective is to predict whether a learner is at risk to drop out from the MOOC program or not. Usually, we subdivide our dataset into three partitions: Train_set (data used to train the model: 250 151), Test_set (reserved for the evaluation of our model: 83 385) and validation_set (83 385) to look for the model settings that give the best performance.

• **Model Performance:** For prediction purpose, there are many algorithms used to model such issue. In this section, we use the “validation_set” data to evaluate firstly these algorithms according to their performance. To measure the performance of each model according to the confusion matrix, we use the F1 Score that relates Precision to Recall where:

- Precision: Minimizes the False Positive rate (Learners who are certified but the system identifies them as at risk).
- Recall: Minimizes the rate of False Negatives (Learners who are at risk but who are identified by the system as certified).
- The confusion matrix represents the four possible outcomes (4 different combinations of expected and actual values)

		Actual Values	
		Certified	Not Certified
Predicted values	Certified	True certified	False certified
	Not Certified	False not certified	True not certified

The F1 Score was measured using different algorithms such as Support Vector Machine (SVM), K-Nearest Neighbor (KNN) and Logistic Regression.

SVM						
		[[79742	643]			
		[851	2149]]			
				precision	recall	f1-score
				support		
	0	0.99	0.99	0.99	80385	
	1	0.77	0.72	0.74	3000	
KNN						
		[[79603	782]			
		[826	2174]]			
				precision	recall	f1-score
				support		
	0	0.99	0.99	0.99	80385	
	1	0.74	0.72	0.73	3000	

Figure 5. Confusion Matrices for SVM and KNN algorithms

Therefore, we retain the Logistic Regression algorithm as it returns the best predictions results and achieves very good scores in a short time even if the size of our “validation_set” is relatively large. Therefore, the classification report obtained for the Logistic Regression is as follow:

Logistic Regression

	precision	recall	f1-score	support
0	1.00	1.00	1.00	80385
1	0.97	0.97	0.97	3000

Figure 6. Confusion Matrix for Logistic regression algorithm

- Among 80385 non-certified learners, we were able to identify 80287 correctly, and we have 98 cases in error with an accuracy and recall of almost 100%.
 - We correctly identified 2910 certified learners and incorrectly rejected 90 learners with an accuracy and recall of almost 97%.
- **Model Execution:** In this section, we address the execution of the retained algorithm (logistic regression) using default hyperparameters. These parameters are useful to improve the model and to resolve some model issues such as underfitting or overfitting problems. As the logistic regression algorithm does not really have any critical hyperparameters to set, we only adjust the solver as well as the strength of the regularization parameter “C” accordingly. Usually, we can diagnose these model issues by examining the learning curves during the training phase. In fact, learning curves are widely used as a diagnostic tool for algorithms that learn incrementally from a dataset. Therefore, during the model training phase we generate learning curves after each update by the “Training_Set” data to evaluate the performance of the model based on a “Test_Set” dataset.

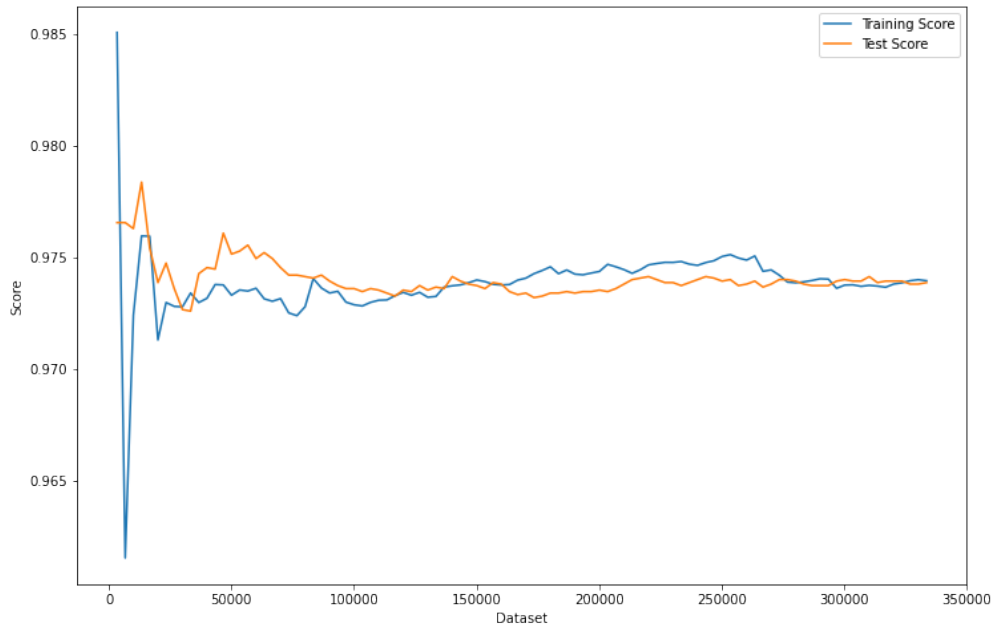


Figure 7. Learning curves related to the training and the test scores

The learning curve above shows the relationship between the training score and the test score for a recall estimator with a variable number of data samples. Generally, if the Training score and Test score converge as more data is added, the model is certainly not overfitting and could be generalized more efficiently. In our case, we can observe that the Training score curve and the Test score curve converge towards a recall score of 0.97. Hence, we can identify learners at risk to drop out from the program using the logistic regression algorithm with accuracy of 97%.

6.2. LEARNERS' PATH RECOMMANDATION

Once we have identified at-risk learners, we will group them into profiles based on their skills in our dataset. Then, we use the PSO algorithm to analyze their profiles so that the learning materials can be easily recommended at each interaction with the system.

Profiles Identification: Generally, we can determine the optimal number of clusters by following the steps below:

- Compute clustering algorithm (e.g., k-means clustering) for different values of k. For instance, by varying k from 1 to 10 clusters.
- For each k, calculate the total within-cluster sum of square (WSS).
- Plot the curve of WSS according to the number of clusters k.
- The location of a bend (knee) in the plot is generally considered as an indicator of the appropriate number of clusters.

In this paper, we suggest using the so-called "Elbow" method [27] to identify the optimal number of learners' profiles (clusters). This method performs several tests with different values of k (the number of clusters). Then, it records for each execution the accuracy score of the clusters by measuring the variance values. Indeed, by plotting the evolution of our model with respect to the number of clusters, we obtain a "node" zone for cost minimization (inertia.). In our case, the optimal number of clusters is "4" as shown in the figure below:

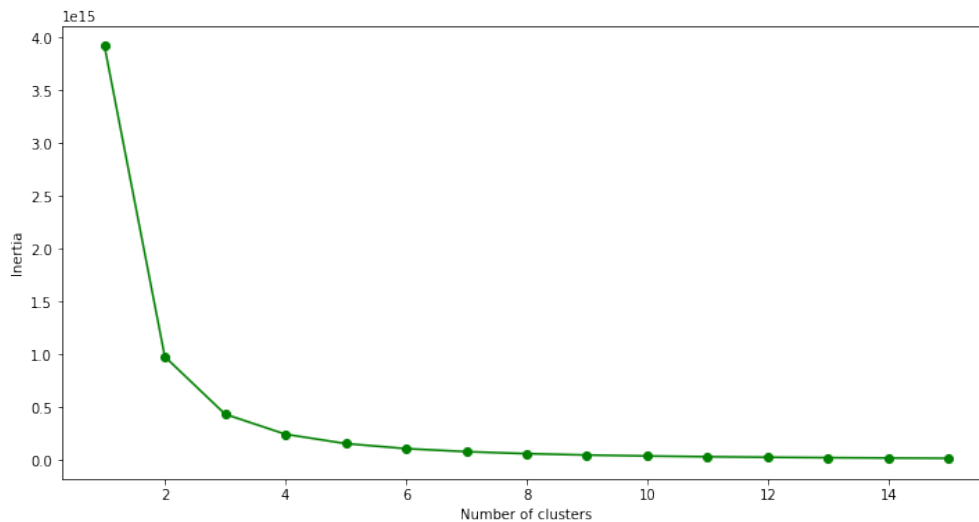


Figure 8. Optimal number of profiles for our case study

Now that we have the learners grouped in four profiles (clusters), these profiles contain among others the courses taken by the neighboring learners. Hence, we gather the importance of each course in different profiles. Then, we suggest to the user the best courses according to the average score attributed by the neighboring learners in his profile. This will encourage at-risk learners to stay connected to the MOOC and to continue the program learning.

Learners' Path Optimization: Based on learners' experience and knowledge shared with their neighborhoods, the PSO algorithm will search in an iterative way to determine the best material for learners during the advancement of the learning process. In fact, each individual has position and velocity parameters to move in a search space. Therefore, the system adjusts the learner position according to the two memories: The best position obtained up to the moment considered, and the best position of the population.

Indeed, PSO was used to find the optimal positions of the profiles' centroids. The solution we choose is therefore the one that minimizes the sum of the distances between the elements of a cluster (profile) and its center (centroid). In this case, a swarm learner is only aware of the position and speed (velocity) of the closest neighbors with a very close profile. The following graph shows the projection of our variables in a 2D space, with the centroids marked in red:

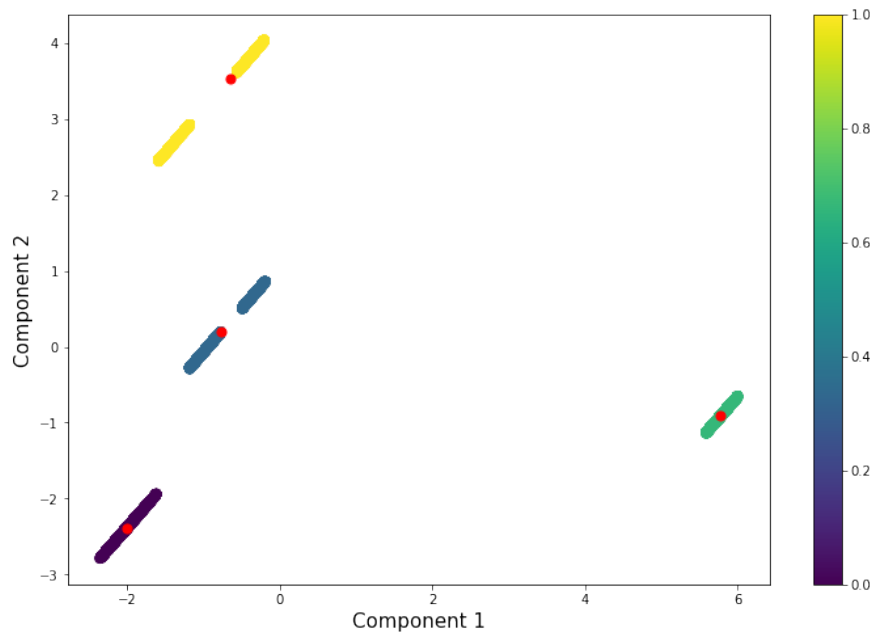


Figure 9. Optimal positions of the profiles' centroids using PSO algorithm

Moreover, learners will use not only their own memories, but also local information on their closest neighbors to decide their own choice. In this case, simple rules, such as "go at the same speed as others", "follow the same path as others" or "stay close to your neighbors" are examples of behaviors that are sufficient to maintain the cohesion of learners, and which allow the implementation of complex and adaptive collective behaviors.

7. CONCLUSION AND PERSPECTIVE

In summary, we suggest in this paper an adaptive learning approach based on particle swarm optimization (PSO) algorithm. The main objective is to increase the retention rate of learners on the MOOCs programs by providing for each learner the necessary support based on their actual learning demands and different knowledge levels. Although adaptive learning interest in education, the implementation of such approach remains quite limited. A review of the literature shows that higher education institutions encounter a wide range of challenges while adopting adaptive learning concepts. These challenges are mainly real-time data processing or difficulties in integrating adaptive learning solutions into existing learning management systems. As prospects, we plan in the near future to apply the proposed approach to other larger sample sizes so as to analyze the size effect. In the meantime, we also plan

to apply other optimization approaches by taking into consideration possible factors that might affect the learning performance of students.

REFERENCES

1. Y. Pang, W. Liu, Y. Jin, H. Peng, T. Xia, and Y. Wu, *Adaptive recommendation for MOOC with collaborative filtering and time series*, Computer Applications in Engineering Education, vol. 26, no. 6, pp. 2071–2083, 2018.
2. A. Qaffas, K. Kaabi, R. Shadiey, and F. Essalmi, *Towards an optimal personalization strategy in MOOCs*, Smart Learning Environments, vol. 7, no. 1, 2020.
3. S. E. Miloud, S. Soukaina, A. Salma, and C. M. E. Hassan, *An Adaptive Learning Approach for Better Retention of Learners in MOOCs*, in Proceedings of the 3rd International Conference on Networking, Information Systems & Security, (New York, NY, USA), Association for Computing Machinery, 2020.
4. A. Gulati, *AN OVERVIEW OF MASSIVE OPEN ONLINE COURSES (MOOCs): SOME REFLECTIONS*, International Journal of Digital Library Services, vol. 3, no. 4, pp. 37–46, 2013.
5. D. S. Chaplot, E. Rhim, and J. Kim, *Personalized Adaptive Learning Using Neural Networks*, in Proceedings of the Third (2016) ACM Conference on Learning @ Scale, (New York, NY, USA), pp. 165–168, Association for Computing Machinery, 2016.
6. D. W. Corne, A. Reynolds, and E. Bonabeau, *Swarm Intelligence*, in Handbook of Natural Computing (G. Rozenberg, T. Back, and N. Kok, eds.), (Berlin, Heidelberg), pp. 1599–1622, Springer Berlin Heidelberg, 2012.
7. B. Hssina and M. Erritali, *A personalized pedagogical objectives based on a genetic algorithm in an adaptive learning system*, Procedia Computer Science, vol. 151, no. 2018, pp. 1152–1157, 2019.
8. E. Subiyantoro, A. Ashari, and S. Suprpto, *Learning Path Recommendation using Hybrid Particle Swarm Optimization*, Advances in Science, Technology and Engineering Systems Journal, vol. 6, no. 1, pp. 570–576, 2021.
9. G. Frankl, S. Napetschnig, and P. Schartner, *Pathways to Successful Online Testing: eExams with the “Secure Exam Environment” (SEE)*, volume 1022. 2019.
10. J. Xi, Y. Chen, and G. Wang, *Design of a Personalized Massive Open Online Course Platform Overview of MOOCS*, pp. 58–70.
11. L. Xiao-Li, *Intelligent recommendation algorithm for social networks based on gradient particle swarm optimization*, Journal of Physics: Conference Series, vol. 1168, no. 5, 2019.
12. I. Kondratova, H. Molyneaux, and H. Fournier, *Supporting Trust and Engagement in Personalized Learning*, volume 10925 LNCS. Springer International Publishing, 2018.
13. R. P. B, I. I. Bittencourt, and W. Lemos, *Visualizing Learning Analytics*, volume 1. Springer International Publishing, 2018.
14. V. Aleven, J. Sewall, J. M. Andres, R. Sottolare, R. Long, and R. Baker, *Towards Adapting to Learners at Scale: Integrating MOOC and intelligent tutoring frameworks*, Proceedings of the 5th Annual ACM Conference on Learning at Scale, L at S 2018, 2018.
15. M. Uddin, N. Ahmed, and A. Mahmood, *A Learner Model for Adaptable e-Learning*, International Journal of Advanced Computer Science and Applications, vol. 8, no. 6, 2017.
16. Y. Zhang, X. Liu, F. Bao, J. Chi, C. Zhang, and P. Liu, *Particle swarm optimization with adaptive learning strategy*, Knowledge-Based Systems, vol. 196, p.105789, 2020.
17. M. Elad, *3D Virtual Reality in K-12 Education: A Thematic Systematic Review*. 2020.
18. A. Alzaghoul and E. Tovar, *A proposed framework for an adaptive learning of Massive Open Online Courses (MOOCs)*, Proceedings of 2016 13th International Conference on Remote Engineering and Virtual Instrumentation, REV 2016, no. February, pp. 127–132, 2016.
19. Q. Zhao, Y. Zhang and J. Chen, *An Improved Ant Colony Optimization Algorithm for Recommendation of Micro-learning Path*, IEEE International Conference on Computer and Information Technology (CIT), pp. 190-196, 2016.
20. A. Muhammad, Q. Zhou, G. Beydoun, D. Xu, and J. Shen, *learning path adaptation in online learning systems*, Proceedings of the 2016 IEEE 20th International Conference on Computer Supported Cooperative Work in Design, CSCWD 2016, pp. 421–426, 2016.
21. W. Ding, Z. Zhu, and Q. Guo, *A New Learner Model in Adaptive Learning System*, 2018 3rd International Conference on Computer and Communication Systems, ICCCS 2018, no. 1, pp. 472–476, 2018.
22. E. M. Smaili, C. Khouda, S. Sraidi, and M. E. H. Charaf, *An Optimized Method for Adaptive Learning based on PSO Algorithm*, 2020 IEEE 2nd International Conference on Electronics, Control, Optimization and Computer Science (ICECOCS), pp. 1–5, Dec, 2020.
23. S. Soukaina, S. E. Miloud, and C. M. E. Hassan *MOOCs Performance Analysis based on Quality and Machine Learning Approaches*, 2020 IEEE 2nd International Conference on Electronics, Control, Optimization and Computer Science (ICECOCS), pp. 1–5, Dec, 2020.
24. E. M. Smaili, S. Sraidi, S. Azzouzi, and M. E. H. Charaf, *Towards Sustainable e-Learning Systems Using an Adaptive Learning Approach*, in Emerging Trends in ICT for Sustainable Development (M. B. Ahmed, S. Mellouli, L. Braganca, B. A. Abdelhakim, and A. Bernadetta, eds.), (Cham), pp. 365–372, Springer International Publishing, 2021.
25. S. Soukaina, S. E. Miloud, S. Azzouzi, and C. M. E. Hassan, *Quality Approach to Analyze the Causes of Failures in MOOC*, in 2020 5th International Conference on Cloud Computing and Artificial Intelligence: Technologies and Applications (CloudTech), pp. 1–5, Nov, 2020.
26. A. Kaveh, *Particle swarm optimization*, In: Advances in Metaheuristic Algorithms for Optimal Design of Structures. pp. 11–43, Springer, Cham, 2017.
27. D. Haroon, *Clustering*, In: Python Machine Learning Case Studies. Apress, Berkeley, CA, 2017.