Mining Learners' Behaviors: An Approach Based on Educational Data Mining Techniques

Ouafae El Aissaoui, Yasser El Alami El Madani, Lahcen Oughdir, Ahmed Dakkak and Youssouf El Allioui

Abstract Educational data mining is a research field that aims to apply data mining techniques in educational environments. Many data mining techniques such as clustering, classification, and prediction can be performed on educational data in order to analyze the learner behaviors. In this work, we have used the clustering and classification techniques to predict the learners' learning styles. The students' behaviors while using the e-learning system have been captured from the log file and given as an input of a clustering algorithm to group them into 16 clusters. The resulted clusters were labeled with learning styles combinations based on the Felder and Silverman learning style model. Then the labeled behaviors were given as input to four classifiers: naive Bayes, Cart, Id3, and C4.5 to compare their performance in predicting students' learning styles. The four classifiers were performed using Weka data mining tool, and the obtained results showed that Id3 yielded better results than the other classifiers.

Keywords Educational data mining · Data mining techniques · Clustering · Classification · E-learning system · Felder and Silverman learning style model

1 Introduction

In recent years many technologies have been developed to improve the efficiency of E-learning environments. Using those technologies, the learners' behaviors and preferences can be detected and used to enhance the learning process. In this context, the Educational data mining field (EDM) has emerged.

O. El Aissaoui (✉) · L. Oughdir · A. Dakkak
LSI, FPT, University of Sidi Mohammed Ben Abdellah, Taza, Morocco
e-mail: ouafae.elaissaoui@usmba.ac.ma

Y. El Alami El Madani
ENSIAS, Mohammed V University, B.P.: 713, Agdal, Rabat, Morocco

Y. El Allioui
LS3M, FPK, USMS University, B.P.: 145, 25000 Khouribga, Morocco

V. Bhateja et al. (eds.), *Embedded Systems and Artificial Intelligence*,
Advances in Intelligent Systems and Computing 1076,
https://doi.org/10.1007/978-981-15-0947-6_62

EDM is a research field that aims to apply data mining techniques on educational data to extract useful information about the students' behaviors and then make enhanced learning strategies based on the extracted information. Various data mining techniques can be used in the EDM field to achieve different purposes. One of the important purposes of the EDM is the enhancement of the student model by predicting the learners' characteristics.

The learning style is one of the vital characteristics that construct a student model. Knowing learners' learning styles helps adaptive E-learning systems to provide customized contents that fit the learners' preferences. Many solutions have been proposed to identify students' learning styles, the traditional one consists in asking the students to fill in a questionnaire, this solution has many disadvantages. Firstly, filling in a questionnaire is a boring task that consumes a lot of time. Secondly, students aren't always aware of their learning styles and the importance of the further use of questionnaires, which can lead them to give arbitrary answers, therefore, the results obtained from the questionnaires can be inaccurate and might not reflect the real learning styles of the students. Thirdly, the results obtained from the questionnaires are static, while the learning styles can be changed during the learning process.

This paper presents an approach to detect the learning style automatically using the existing learners' behaviors and based on the This paper presents an approach to detect the learning style automatically using the existing learners' behaviors and based on the Felder and Silverman Learning Style Model (FSLSM). Four classifiers have been performed to detect the learners' learning styles which are: Naive Bayed, Id3, CART, and C4.5. We have compared the performance of the four classifiers to get the most efficient in detecting students' learning styles.

The rest of this paper is organized as follows. Section 2 gives a brief definition of EDM and describes the FSLSM and the used algorithms. Section 3 introduces a literature review of related work. Section 4 describes the methodology of our approach. The results are presented in Sect. 5; Finally, Sect. 6 presents our conclusions and future works.

2 Preliminary

2.1 Application of Data Mining in Educational Data

Data mining, also known as Knowledge Discovery in Data (KDD), is the process of finding hidden and useful information from large volumes of data. Many fields have exploited the efficiency of data mining techniques to make decisions such as e-commerce, bioinformatics, and e-learning. Applying data mining techniques to educational data is commonly known as educational data mining (EDM). One of the major utilities of EDM is the improvement of the student model. According to [1] a student model contains many characteristics which are: Knowledge and skills, misconception and errors, affective factors, Cognitive features, Meta-cognitive skills, and learning style. Many researches have been focusing on detecting the learner' learning style, various techniques have been used and many approaches have been proposed. Among the techniques that were widely used there are: Neural Network, Decision Tree, Bayesian Networks, and Association Rule.

2.2 The Felder and Silverman Learning Style Model

Felder and Silverman learning style model was developed by Felder and Silverman in 1998 [2]. This model uses the notion of dimensions where each dimension contains two opposite categories, and each student has a dominant preference in each dimension's category. The four dimensions of the FSLSM are: processing (active/reflective), perception (sensing/intuitive), input (visual/verbal), understanding (sequential/global). A learning style is defined by combining one category from each dimension.

Active learners tend to retain and understand information best by doing something with the learning material. Reflective learners prefer to think about the learning material quietly first. Active learners also prefer to study in group, while the reflective learners prefer to work individually.

Sensing learners like courses that deal with the real-world facts. They tend to be more practical by doing hands-on (laboratory) works and they are more competent than intuitive at memorizing facts. Intuitive learners don't like materials that contain a lot of memorization and routine calculations, and they prefer the ones involving abstractions and mathematical formulations. They may be better than sensors at innovating and grasping new concepts.

The visual learners prefer to see what they learn by using visual representations such as pictures, diagrams, and charts. While the verbal learners like information that are explained with words; both written and spoken.

Sequential learners tend to go through the course step by step in a linear way, with each step followed logically by the next one. Global learners tend to learn in large jumps, by accessing courses randomly without seeing connections.

In this work, we have based on the FSLSM for many reasons. Firstly; this model uses the concept of dimensions and describes thoroughly the learning styles preferences. This description mentions the types of learning objects that can be included in each learning style preference, and this is an interesting characteristic to our work because knowing the learning styles preferences of learning objects helps us to determine the learning style of learners 'sequences. Secondly; according to our approach a LS can be changed over time, and then updated using a classifier technique. The FSLSM provides this possibility by considering LS as tendencies and students can act in a non-deterministic way as pointed by [3]. Thirdly, the FSLSM is the most used in adaptive e-learning systems and the most appropriate to implement them as mentioned by [4, 5].

According to FSLSM, each learner prefers a specific category in each dimension. Thus a learning style is defined by combining one preference from each dimension. As a result, we will obtain sixteen combinations:

```
Learning Styles Combinations (LSCs) = {(A,Sen,Vi,G),
(A,Sen,Vi,Seq), (R,Sen,Vi,G), (A,Sen,Ve,Seq),
(A,Sen,Ve,G), (R,Sen,Ve,Seq), (R,Sen,Ve,G), (A,I,Ve,G),
(A,I,Vi,Seq), (A,I,Vi,G), (R,I,Vi,Seq), (R,I,Vi,G),
(R,S,Vi,Seq), (A,I,Ve,Seq), (R,I,Ve,Seq), (R,I,Ve,G)}.
```

2.3 Used Algorithms

K-modes clustering algorithm

The *k*-modes is a clustering algorithm that aims to group similar categorical objects into *k* clusters. This algorithm is an extension of *k*-means clustering algorithm with three modifications: replacing the Euclidean distance function with the simple matching dissimilarity measure. Instead of means, it uses modes, and instead of updating the centroids it updates modes using a frequency based method.

The following steps illustrate how to cluster a categorical data set *X* into *k* clusters:

Step 1: Randomly Select *k* initial modes, one for each cluster.

Step 2: compute the distance from each object to each mode using the dissimilarity Measures, and then associate each object to the cluster whose mode is the nearest to it. This association defines the first *k* clusters.

Step 3: Update the modes of the newly defined clusters from Step 2 using Theorem (1) described below and then retest the dissimilarity between the objects and the updated modes. If an object is found such that its nearest mode belongs to another cluster rather than its current one, reallocate the object to that cluster and update the modes of both clusters.

Step 4: repeat step 3 until no object has moved to another cluster after a full cycle test of the whole data set.

The dissimilarity Measures and the theorem (1) that are used in the previous steps; are described below:

- Dissimilarity Measures

To compute the dissimilarity between two objects *X* and *Y* described respectively by *m* categorical attributes values $(x_1, x_2, \ldots, x_m)$ and $(y_1, y_2, \ldots, y_m)$, the two following functions can be used:

$$d(X, Y) = \sum_{j=1}^{m} \delta(x_j, y_j) \text{ where } \delta(x_j, y_j) = \begin{cases} 0, x_j = y_j \\ 1, x_j \neq y_j \end{cases} \quad (1)$$

$$d_{x^2}(X, Y) = \sum_{j=1}^{m} \frac{\left(n_{x_j} + n_{y_j}\right)}{n_{x_j} n_{y_j}} \delta(x_j, y_j) \tag{2}$$

where n_{x_i} and n_{y_j} are respectively the numbers of objects that have the categories x_j and y_j for attribute j.

- How to select a mode for a set

Let $X = \{X_1, X_2, \ldots, X_n\}$ a mode of X is a vector $Q = [q_1, q_2, \ldots, q_m]$ that minimizes the following equation:

$$D(Q, X) = \sum_{i=1}^{n} d(X_i, Q) \tag{3}$$

where D can be either Eqs. (1) or (2).

Theorem 1 *The function* $D(Q, X)$ *is minimized if* $f_r\left(A_j = q_j \backslash \boldsymbol{X}\right) \geq f_r\left(A_j = C_{k,j} \backslash \boldsymbol{X}\right)$ *for* $q_j \neq C_{k,j}$ *for all* $j = 1, \ldots, m$ *where* $f_r\left(A_j = C_{k,j} \backslash \boldsymbol{X}\right) = \frac{n_{C_{k,j}}}{n}$ *is the relative frequency of category* $C_{k,j}$ *in X.*

Naive Bayes

The Naive Bayes is a supervised classifier and it is an extension of the Bayes theorem with two simplifications [6, 7].

The first simplification is to use the conditional independence assumption. That is, each attribute is conditionally independent of every other attribute given a class label C_i. See the following equation:

$$\begin{aligned} P(C_i \backslash A) &= P(C_i) * \frac{P(a_1, a_2, \ldots, a_m \backslash C_i)}{P(a_1, a_2, a_3, \ldots, a_m)} \\ &= P(C_i) * \frac{P(a_1 \backslash C_i) P(a_2 \backslash C_i), \ldots, P(a_m \backslash C_i)}{P(a_1, a_2, a_3, \ldots, a_m)} \\ &= P(C_i) * \frac{\prod_{j=1}^{m} P\left(a_j \backslash C_i\right)}{P(a_1, a_2, a_3, \ldots, a_m)} \end{aligned} \tag{4}$$

The second simplification is to ignore the denominator $P(a_1, a_2, a_3, \ldots, a_m)$. Because $P(a_1, a_2, a_3, \ldots, a_m)$ appears in the denominator of $\boldsymbol{P(C_i \backslash A)}$ for all values of i, removing the denominator will have no impact on the relative probability scores and will simplify calculations. After applying the two simplifications mentioned earlier, we obtain the Naive Bayes classifier which consists in labeling the object A with the class label C_i that maximizes the following equation:

$$P(C_i \backslash A) \sim P(C_i) \cdot \prod_{j=1}^{m} P\left(a_j \backslash C_i\right), \; i = 1, 2, \ldots, n \tag{5}$$

Decision Tree Algorithms

Many algorithms exist to implement a decision tree, where each algorithm use a specific method to construct a tree. Some popular algorithms include ID3 [8], C4.5 [9], and CART [10].

- ID3 Algorithm:

ID3 (or Iterative Dichotomiser 3) is one of the first decision tree algorithms that Handles only Categorical value, and it was developed by John Ross Quinlan. Let S be a set of categorical input variables where each input variable y belongs to an attribute Y, x be the output variable (or the predicted class), The ID3 requires the following steps:

Step 1: compute the entropy $H(S)$ for data-set
Step 2: for every attribute:

Step 2.1: Compute the conditional entropy $H_{x/y}$. Given an attribute Y, its value y, its outcome X, and its value x.
Step 2.2: compute the information gain for the current attribute.
Step 2.3: choose the attribute with the largest information gain for the first split of the decision tree.
Step 2.4: Repeat until we get the desired tree.

The entropy $H(S)$, the conditional entropy $H_{x/y}$, and the information gain are computed respectively as follows:

$$H(S) = -\sum_{\forall x \in X} p(x) \cdot \log_2 p(x) \tag{6}$$

$$H_{x/y} = -\sum_{\forall y \in Y} p(y) \cdot \sum_{\forall x \in X} p(x/y) \cdot \log_2 p(x/y) \tag{7}$$

$$\text{InfoGain}_A = H_S - H_{S/A} \tag{8}$$

- C4.5 Algorithm

The C4.5 algorithm is an improved version of the original ID3 algorithm. The C4.5 algorithm can handle missing data. If the training set contains missing attribute values, the C4.5 evaluates the gain for an attribute by considering only the records where the attribute is defined.

The C4.5 algorithm Handles both Categorical and continuous attributes. Values of a continuous variable are sorted and partitioned. For the corresponding records of each partition, the gain is calculated, and the partition that maximizes the gain is chosen for the next split.

The C4.5 algorithm addresses the overfitting problem in ID3 by using a bottom-up technique called pruning to simplify the tree by removing the least visited nodes and branches.

- CART Algorithm

CART (or Classification And Regression Trees) can handle both Categorical and continuous attributes like the C4.5 algorithm. Whereas C4.5 uses entropy based criteria to rank tests, CART uses the Gini diversity index defined in the following Equation

$$\mathrm{Gini}_X = 1 - \sum_{\forall x \in X} P(x)^2 \tag{9}$$

Similar to C4.5, CART can handle missing values and overcome the overfitting problem by using the Cost-Complexity pruning strategy.

3 Related Works

Many researchers have focused on detecting students' learning styles using educational data mining techniques. The EDM uses algorithms from the field of data mining and machine learning to build a model from the existing students' behaviors and then the constructed model is used to determine the learning style of a new student. Among the popular algorithms being used, there are the neural network, Decision tree, and Bayesian network. Reference [11] introduces an approach which uses artificial neural networks to identify students' learning styles. The approach has been evaluated with data from 75 students and found to outperform the current state of the art approaches. Reference [12] presents an approach to recognize automatically the learning styles of individual students according to the actions that they have performed in an e-learning environment. This recognition technique is based upon feed-forward neural networks.

Beside the neural network, the decision tree is also a widely used technique to determine LSs. In [13] the authors compare the performance of J48, NBTree, RandomTree and CART in inferring LSs. Authors in [14] used NBTree classification algorithm in conjunction with Binary Relevance classifier, the learners are classified based on their interests. Then, learners' learning styles are detected using these classification results.

According to [15] the Bayesian networks are ones of the most widely adopted classifiers to infer the learning style [16]. Used a Bayesian network (BN) in order to identify students' learning styles. They identified various behaviors that may be relevant to detect learning styles in a given E-learning system. Then, a BN was trained with data from 50 students, using initial probabilities based on expert knowledge. The trained BN was then evaluated using 27 students. For each student, the BN provides a probability that the student has a particular learning style preference. As a result, the approach obtained an overall precision of 58% for A/R, 77% for S/I a 63% for S/G (the V/V dimension was not considered) [17]. Presents a work that aims to enhance the capability of such systems by introducing adaptivity in the way the information is presented to the online learner. The adaptivity takes into account the learners' learning styles by modeling them using a Bayesian Network.

Most of the proposed approaches aim to classify learners according to the eight learning styles preferences of FSLSM. This study differs from others by considering 16 learning styles combinations instead of eight.

4 Methods

The EDM techniques consist in applying data mining techniques on educational data. One of the sources of educational data is the log file since most of the E-learning environments have their own log files that record the interaction history of students. The data recorded in log files can be analyzed by using web usage mining techniques to investigate the learners' behaviors. The basic steps of the web usage mining process are; data collection, preprocessing and pattern discovery.

4.1 Data Collection

In our work, the learners' behaviors were collected from the E-learning platform of Sup 'Management Group. The online educational system was developed based on Moodle platform. Moodle is a free Open Source software used to help educators and students to enhance the learning process. We collected 1235 sequences from the E-learning platform, which represent the learners' behavior.

4.2 Data Preprocessing

In This step, we cleaned the collected data from all unnecessary information and we kept just the ones that seemed relevant for our study. Thus, for each learner's sequence, we kept the following information: the sequence id, session id, learner id, and the set of learning objects accessed by the learner in a session.

The accessed learning object can be used to determine the learning style of a learner's sequence, because each student prefers to use some specific types of learning objects while learning, so there is a relationship between the learner's learning styles and the accessed learning objects, this relationship has to be analyzed by matching each LO with its relevant learning style preferences.

Based on the matching table presented in our previous work [18]; we have obtained the following table, where each LO is matched with its appropriate LSC.

4.3 Pattern Discovery

This step aims to apply data mining techniques to the data preprocessed in the previous step in order to predict the students' learning styles. In this study we have used two data mining techniques:

- The clustering to group the students into 16 clusters where each cluster corresponds to LSC.
- The classification to predict the learning style combination for a new sequence.

The following schema resumes our proposed approach (Fig. 1).

Clustering
This step aims to group the learners' sequences into 16 LSCs. We have used the *k*-modes clustering algorithm because of its ability to deal with categorical objects, since the learners' sequences which were considered as the input to that algorithm have categorical attribute values.

After extracting and preprocessing the learners' behaviors from the log file using web usage mining techniques, we can use them as an input to the *K*-modes by turning

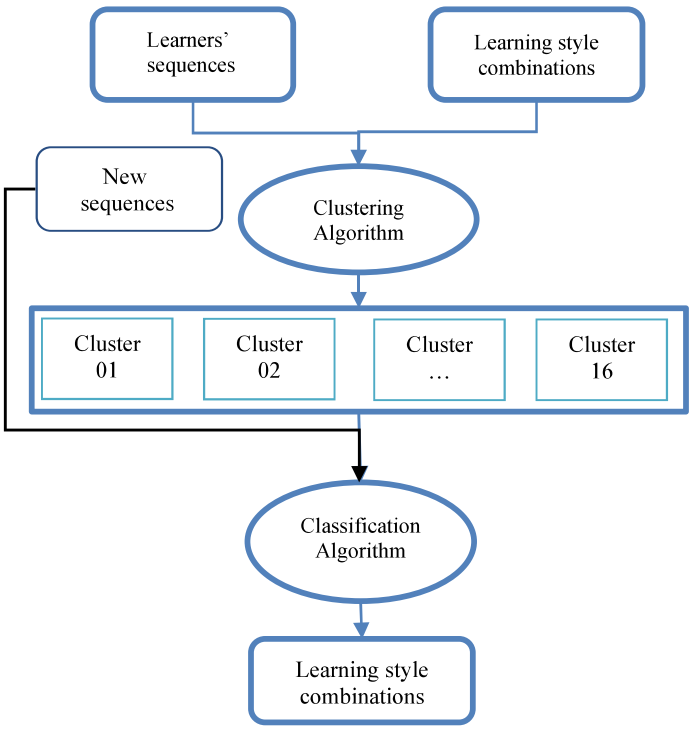

Fig. 1 Our approach

them into a matrix with M rows corresponding to M sequences and sixteen columns to store the attribute values where the attributes of each sequence correspond to the sixteen Los presented in the previous mapping table (Table 1).

Let $S = \{S_1, \ldots, S_i, \ldots, S_n\}$ be a set of n sequences (categorical objects), each sequence S_i is defined by 16 attribute values $\left(s_{i1}, \ldots, s_{ij}, \ldots, s_{i16}\right)$.

And let $A_1, \ldots, A_j, \ldots, A_{16}$ be the 16 attributes describing the n sequences where: $\boldsymbol{A}_1$ = video, $\boldsymbol{A}_2$ = PPTs, $\boldsymbol{A}_3$ = demo, $\boldsymbol{A}_4$ = Exercise, $\boldsymbol{A}_5$ = Assignments, $\boldsymbol{A}_6$ = PDFs , $\boldsymbol{A}_7$ = Announcements, $\boldsymbol{A}_8$ = References, $\boldsymbol{A}_9$ = Examples, $\boldsymbol{A}_{10}$ = Practical Material, $\boldsymbol{A}_{11}$ = Forum, $\boldsymbol{A}_{12}$ = Topic list, $\boldsymbol{A}_{13}$ = Images, $\boldsymbol{A}_{14}$ = Charts, $\boldsymbol{A}_{15}$ = Email, $\boldsymbol{A}_{16}$ = Sequential.

For each sequence, Each attribute has two possible categorical attribute values: {yes or no, yes if the *j*th learning object exists in the sequence, No if it doesn't exist.

Therefore, each sequence is presented as $\left(s_{i1}, \ldots, s_{ij}, \ldots, s_{i16}\right)$ where s_{ij} = {yes or no}.

To perform the *k*-modes algorithm we used the *R* software framework for statistical analysis and graphics. The dataset employed in our approach was extracted from the E-learning platform's log file of Sup' Management Group. This dataset records 1235 learners' sequences. The following figure displays the results obtained after running the *k*-modes with the *R* (Fig. 2).

The above figure shows the size of each cluster. The obtained clusters were labeled with the LSCs based on the minimum distances between the clusters' Modes and the LSCs' vectors presented in Table 1. The distances are computed using the dissimilarity measure (described in subsection II. C. 1).

Classification

After applying the *K*-Modes algorithm and labeling the sequences with the LSCs, the labeled sequences can be used as a training dataset to train a classification algorithm, and then use it to predict the LSC for a new sequence.

In this work, we have applied four classifiers, the naïve Bayes, CART, C4.5, and ID3 in order to compare their performance in predicting learning styles. The four algorithms were run using the Weka data mining tool. Weka implements machine learning algorithms for data mining tasks. It contains tools for data preprocessing, classification, regression, clustering, association rules mining, and visualization. The performance of the four classifiers was evaluated using the *K*-fold cross validation technique.

5 Results and Discussion

To compare the performance of the four applied algorithms in predicting LSC, we used the 10-fold cross validation for every classifier as suggested by Weka. The results of the experiment are shown in Fig. 3, Tables 2 and 3. Table 2 presents for each classifier the results of the Correctly Classified Instances, the Incorrectly Classified Instances, the Kappa statistic and the Time taken to build the model.

Table 1 Matching learning objects to LSCs

Cluster ID	Cluster meaning	Videos	PPTs	Demo	Exercise	Assignments	PDFs	Announcements	References
C01	Reflective-intuitive-verbal-global	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes
C02	Active-intuitive-verbal-global	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
C03	Reflective-sensing-verbal-global	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes
C04	Active-sensing-verbal-global	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
C05	Reflective-intuitive-visual-global	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes
C06	Active-intuitive-visual-global	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes
C07	Reflective-sensing-visual-global	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes
C08	Active-sensing-visual-global	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes
C09	Reflective-intuitive-verbal-sequential	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes
C10	Active-intuitive-verbal-sequential	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
C11	Reflective-sensing-verbal-sequential	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes
C12	Active-sensing-verbal-sequential	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
C13	Reflective-intuitive-visual-sequential	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes
C14	Active-intuitive-visual-sequential	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes
C15	Reflective-sensing-visual-sequential	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes
C16	Active-sensing-visual-sequential	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes

(continued)

Table 1 (continued)

Cluster ID	Cluster meaning	Examples	Practical Material	Forum	Topic list	Images	Charts	Email	Sequential
C01	Reflective-intuitive-verbal-global	No	No	Yes	Yes	No	No	Yes	No
C02	Active-intuitive-verbal-global	No	No	Yes	Yes	No	No	Yes	No
C03	Reflective-sensing-verbal-global	Yes	Yes	No	Yes	No	No	Yes	No
C04	Active-sensing-verbal-global	Yes	Yes	No	Yes	No	No	Yes	No
C05	Reflective-intuitive-visual-global	No	No	Yes	Yes	Yes	Yes	No	No
C06	Active-intuitive-visual-global	No	No	Yes	Yes	Yes	Yes	No	No
C07	Reflective-sensing-visual-global	Yes	Yes	No	Yes	Yes	Yes	No	No
C08	Active-sensing-visual-global	Yes	Yes	No	Yes	Yes	Yes	No	No
C09	Reflective-intuitive-verbal-sequential	No	No	Yes	Yes	No	No	Yes	Yes
C10	Active-intuitive-verbal-sequential	No	No	Yes	Yes	No	No	Yes	Yes
C11	Reflective-sensing-verbal-sequential	Yes	Yes	No	No	No	No	Yes	Yes
C12	Active-sensing-verbal-sequential	Yes	Yes	No	No	No	No	Yes	Yes
C13	Reflective-intuitive-visual-sequential	No	No	Yes	Yes	Yes	Yes	No	Yes
C14	Active-intuitive-visual-sequential	No	No	Yes	Yes	Yes	Yes	No	Yes
C15	Reflective-sensing-visual-sequential	Yes	Yes	No	No	Yes	Yes	No	Yes
C16	Active-sensing-visual-sequential	Yes	Yes	No	No	Yes	Yes	No	Yes

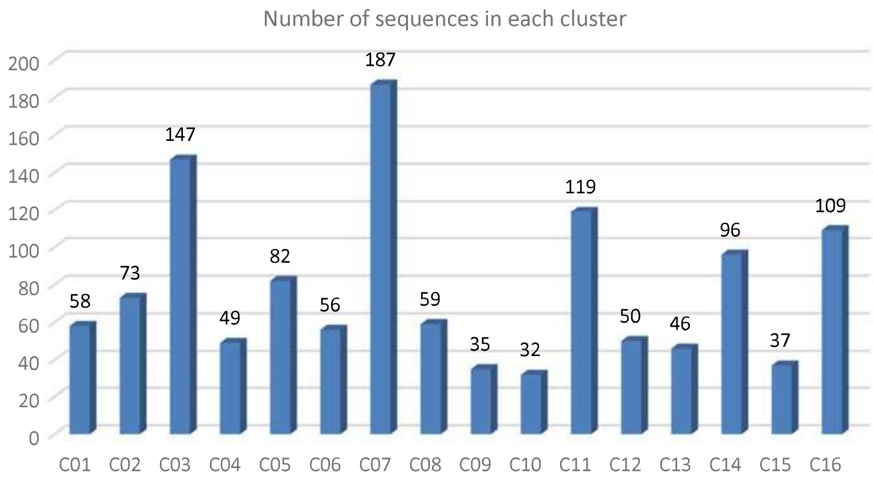

Fig. 2 Result of the *K*-modes algorithm

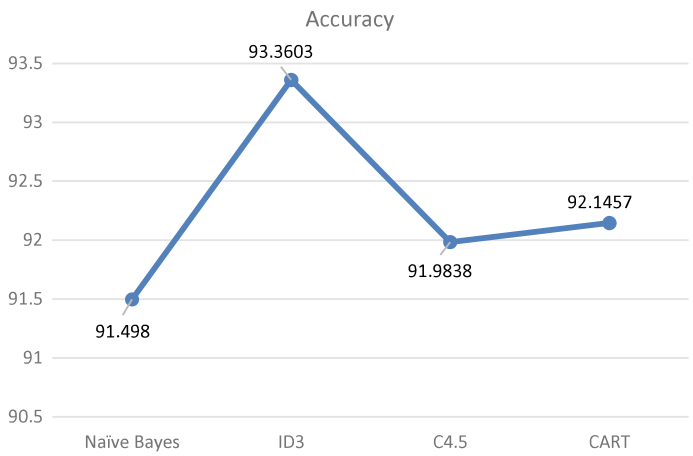

Fig. 3 Accuracy result by classifier

Table 2 Validation metrics by classifier

Algorithms	Correctly classified instances (%)	Incorrectly classified instances (%)	Kappa statistic	Time taken (s)
Naïve Bayes	91.498	8.502	0.9093	0.01
ID3	93.3603	6.6397	0.9292	0.03
C4.5	91.9838	8.0162	0.9145	0.05
CART	92.1457	7.8543	0.9162	0.52

Table 3 Error metrics by classifier

Algorithms	Mean absolute error	Root mean squared error	Relative absolute error (%)	Root relative squared error (%)
Naïve Bayes	0.0186	0.1005	15.8854	41.5141
ID3	0.0087	0.0679	7.4314	28.0504
C4.5	0.0163	0.0921	13.895	38.0679
CART	0.0129	0.0843	11.0151	34.8072

The Accuracy defines the rate at which a model classifies the records correctly. From the above table and Fig. 3, we can notice that the ID3 algorithm has the heist accuracy with a rate of 93.3603%, so this algorithm is more efficient in predicting correct instances than the others classifiers. The lowest Accuracy is 91.498% and it belongs to the Naïve Bayes algorithm.

The Kappa statistic is a measure of agreement between the predictions and the actual labels. It can also be defined as a comparison of the overall accuracy to the expected random chance accuracy. The higher the Kappa metric is, the more efficient a classifier to be as a random chance classifier. The kappa statistic results of the used algorithms are between 0.9093 and 0.9292, so all of them have a good accuracy.

The time taken to build the model is also an important factor to be considered. From the above table, we can observe that the naive Bayes is the fastest one with 0.01 s, while the slowest on is the CART with 0.52 s

Table 3 presents the results of some statistical metrics that are computed based on the differences between forecast and the corresponding observation. Those metrics can be computed using the following formula.

Let's denote the true value of interest as Y and the value estimated using some algorithm as X.

$$\mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N}|X_i - Y_i| \tag{10}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(X_i - Y_i)^2} \tag{11}$$

$$\mathrm{RAE} = \frac{\sum_{i=1}^{N}|X_i - Y_i|}{\sum_{i=1}^{N}\left|\overline{Y} - Y_i\right|} \tag{12}$$

$$\mathrm{RRSE} = \sqrt{\frac{\sum_{i=1}^{N}(X_i - Y_i)^2}{\sum_{i=1}^{N}\left(\overline{Y} - Y_i\right)^2}} \tag{13}$$

As we can notice, all the above metrics compare true values to their estimates, but each one do it in a slightly different way. They all tell us "how far away" are our estimated values from the true value of *Y*. The model with the smaller metrics values is the more performant, in our case, the ID3 has the lowest values so it performs better than the other algorithms in predicting correctly.

6 Conclusion

In this paper, we have used two educational data mining techniques to predict the learners' learning styles. The first technique is the clustering and it has been applied in order to group the learners into 16 LSCs using the *K*-modes algorithm and based on the FSLSM. The second one is the classification that aimed to use the result of the clustering algorithm as a training set to fit a classifier on it and then predict LSCs of new learners' sequences.

In this work, we have compared the performance of four classifiers in predicting LSCs using Weka data mining tool. The comparison was done based on the correctly and incorrectly classified sequences, the time taken by each classifier, the Kappa statistic and the mean error.

The obtained results showed that the ID3 algorithm yielded the highest accuracy and Kappa statistic and the lowest mean error, so it is the most efficient in classifying correct sequences. The naive Bayes is less performant in predicting correct instances, but it is the fastest one.

In the future work, we'd like to have determined automatically others learner's characteristics such as the affective factors in order to create a more efficient learning style model.

References

1. Chrysafiadi, K., Virvou, M.: Student modeling approaches: a literature review for the last decade. Expert Syst. Appl. **40**(11), 4715–4729 (2013)
2. Felder, R.M., Silverman, L.K.: Learning and teaching styles. J. Eng. Educ. **78**, 674–681 (1988)
3. Kinshuk, Liu, T.C., Graf, S.: Coping with mismatched courses: students' behaviour and performance in courses mismatched to their learning styles. Educ. Technol. Res. Dev. **57**(6), 739–752 (2009)
4. Graf S., Kinshuk, Advanced adaptivity in learning management systems by considering learning styles. In: Proceedings—2009 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology—Workshops, WI-IAT Workshops 2009, vol. 3, pp. 235–238 (2009)
5. Kuljis J, Liu, F.: A comparison of learning style theories on the suitability for elearning. In: Conference on Web Technologies, Applications, and Services, pp. 191–197 (2005)
6. Dietrich, D., Heller, R., Yang, B., EMC Education Services: Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data
7. Richert W., Coelho, L.P.: Building Machine Learning Systems with Python (2013)

8. Quinlan, J.R.: Induction of decision trees. Mach. Learn. **1**(1), 81–106 (1986)
9. Salzberg, S.L.: Book review: C4.5: programs for machine learning by J. Ross Quinlan. Morgan Kaufmann Publishers Inc., 1993. Mach. Learn. **16**(3), 235–240 (1994)
10. Breiman, L., Friedman, J., Stone, C.J., Olshen, R.A.: Classification and Regression Trees, 1st edn, Chapman & Hall (1993)
11. Bernard, J., Chang, T.-W., Popescu, E., Graf, S.: Using Artificial Neural Networks to Identify Learning Styles, pp. 541–544. Springer, Cham (2015)
12. Villaverde, J.E., Godoy, D., Amandi, A.: Learning styles' recognition in e-learning environments with feed-forward neural networks. J. Comput. Assist. Learn. **22**(3), 197–206 (2006)
13. Ahmad N.B.H., Shamsuddin, S.M.: A comparative analysis of mining techniques for automatic detection of student's learning style. In: 2010 10th International Conference on Intelligent Systems Design and Applications, 2010, pp. 877–882
14. Özpolat E., Akar, G.B.: Automatic Detection of Learning Styles for an E-Learning System, vol. 53(2). Pergamon (2009)
15. Feldman, J., Monteserin, A., Amandi, A.: Automatic detection of learning styles: state of the art. Artif. Intell. Rev. **44**(2), 157–186 (2015)
16. García, P., Amandi, A., Schiaffino, S., Campo, M.: Evaluating Bayesian networks' precision for detecting students' learning styles. Comput. Educ. **49**(3), 794–808 (2007)
17. Alkhuraiji, S., Cheetham, B., Bamasak, O.: Dynamic adaptive mechanism in learning management system based on learning styles. In: 2011 IEEE 11th International Conference on Advanced Learning Technologies, 2011, pp. 215–217
18. El Aissaoui, O., El Madani El Alami, Y., Oughdir, L., El Allioui, Y.: Integrating web usage mining for an automatic learner profile detection: a learning styles-based approach. In: 2018 International Conference on Intelligent Systems and Computer Vision (ISCV), 2018, pp. 1–6