Proposing a Feature Selection Approach to Predict Learners' Performance in Virtual Learning Environments (VLEs)

https://doi.org/10.3991/ijet.v18i11.35405

Miami Abdul Aziz Al-Masoudy[1], Ahmed Al-Azawei[2](✉)
[1]Medical Instrumentation Techniques Engineering Department, Al-Mustaqbal University College, Hillah, Babil, Iraq
[2]College of Information Technology, University of Babylon, Hillah, Babil, Iraq
ahmedhabeeb@itnet.uobabylon.edu.iq

Abstract—Predicting students' success in virtual learning environments (VLEs) can help educational institutions improve their online services and provide efficient online learning content. However, this cannot be achieved without identifying the possible effective features that have a high influence on students' performance. This research aims to 1) provide an early prediction approach to learners' achievement on VLEs, 2) develop an adapted feature selection method which is called a Developed Sequential Feature Selection (D-SFS), and 3) help enhance online and virtual education quality by highlighting the most effective features that could highly enhance prediction accuracy. The findings suggest that the D-SFS method outperforms the original Sequential Forward Selection (SFS) approach. The prediction accuracy using the SFS method was 92.466% with seventeen features, whereas the proposed approach successfully predicted 92.518% of students' performance using seven features only. Such outcomes highlight the importance of implementing a feature selection method to enhance prediction accuracy, decrease the number of features, and reduce the model's time and execution complexity.

Keywords—educational data mining (EDM), education quality, feature selection, SFS, prediction techniques, student performance, virtual learning environment (VLE), open university learning analytics dataset (OULAD)

1 Introduction

Generally, features are classified into strongly relevant, weakly relevant, or irrelevant. Feature selection techniques can identify the most relevant features of a target class [1]. Such features are sufficient to describe the target class, while the deleted features should not affect the performance of a prediction model. Thus, a model's accuracy may be increased, whereas its complexity should be decreased accordingly [2]. Feature selection methods are divided into wrapper, filter, and embedded techniques [3]. Selecting the most effective features has been widely applied to predict students' performance on different online learning platforms [4].

Online learning becomes very popular in contemporary education [5, 6]. Many forms of online learning are currently available such as learning management systems (LMSs), massive open online courses (MOOCs), and virtual learning environments (VLEs) [6, 7]. However, a high number of learners either fail or drop out in online learning settings [8]. It was also found that students who drop out from online learning courses are significantly more than those in traditional learning courses [4]. This could be accounted for by the absence of direct interaction among the key pillars of the learning process and the absence of the learning atmosphere. Accordingly, guessing learners' academic level or predicting their future performance in such environments is a difficult task [9, 10]. Educational institutions need to pay further attention to understanding factors that may help predict learners' performance in online courses.

Previous literature shows that identifying features that can assist in predicting learners' achievement on VLEs courses still needs further work [11, 12]. Hence, providing early identification of students who will either withdraw or fail can help improve online learning outcomes. This can also lead to overcoming obstacles that may face students in completing their studies and providing clear guidance for educational institutions in designing their online courses based on learners' individual preferences and needs.

This research aims at achieving two objectives. First, it provides an early prediction model that can help educational institutions improve online learning courses and respond to learners' needs. Second, the study adapts a feature selection approach to select the most important features that may have more influence on learners' performance. This research has many contributions which are 1) extending previous work, 2) helping overcome obstacles that students may face in VLEs, and 3) providing a developed feature selection approach that showed better performance than traditional methods.

2 Theoretical background and previous work

2.1 Theoretical background

Prediction models are designed to detect a pattern of a specific problem. If the values of the target class are discrete, classification techniques can be used. While if the values of the target class are continuous, regression techniques can be applied [13]. This research aims at classifying students' performance into three groups namely, success, failure, and withdrawal. Therefore, classification techniques are appropriate for such a problem. This study adopts the Bagging ensemble method. This method was used due to its high prediction accuracy in comparison with other algorithms.

Ensemble methods. An ensemble method is a learning approach that combines multiple models. Each of them is constructed by applying a learning algorithm to a specific problem (a certain part of the used dataset). Such methods are used to enhance accuracy and reduce classification errors [14]. The predictions made by ensembles are usually more accurate than predictions made by a single classifier [15, 14].

The prediction process is carried out by collecting the predictions (voting) made by the basic classifiers (classification) or taking the average predictions of these models (regression) [13]. The general procedure of the ensemble method is explained in Figure 1 [13][15].

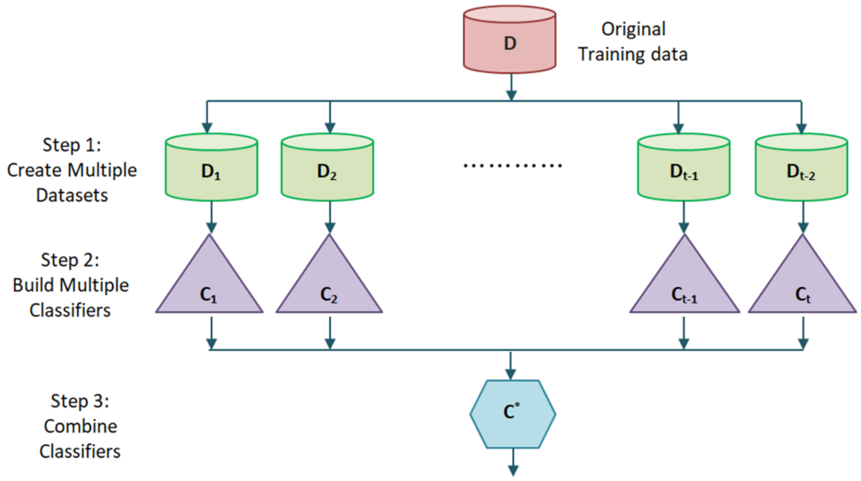


Fig. 1. The general procedure of ensemble methods [13][15]

The models are constructed by training each base classifier on a training set created by resampling the original data. Two common methods are used for constructing an ensemble classifier [13]:

1. Manipulating the input features: In this approach, the training sets are constructed from the original dataset by taking a subset of features. These subsets are selected randomly or depending on the domain experts [16]. This way needs a high-dimensional dataset. An example of this approach is the random forest technique.
2. Manipulating the training instances: In this approach, many models are built by applying a particular learning algorithm from many training sets. These sets are constructed from the original dataset by resampling the instances of this dataset. Finally, these models are integrated into a single model. Examples of this approach are bagging and boosting. This present research uses the bagging method because of its high prediction accuracy.

The name of bagging comes from two words "bootstrap aggregating". It is an ensemble method for improving unstable estimation or classification schemes [15]. In this method, the number of bootstraps is created according to the number of classifiers that are needed to be created. Each bootstrap has the same size as the original dataset and consists of instances that are selected according to a certain probability of replacement. Therefore, in each bootstrap, it is possible to find the same instance more than once in the absence of many other instances [15]. Any object can be classified by taking the predictions of the base classifiers and then making a vote to determine the class of that object according to the majority of votes. The learning algorithm that was used to build the classifiers in this study was the decision tree, particularly, is the REPTree algorithm. The rationale behind this selection is that the REPTree algorithm has the highest classification accuracy when included in the Bagging algorithm among the DecisionStump, J48, LMT, RandomForest, RandomTree, and HoeffdingTree algorithms.

Decision tree. The decision tree is a hierarchical structure that can handle both categorical and numerical data. It consists of directed edges and nodes [17]. These nodes are divided into three types: root nodes, leaf or terminal nodes, and internal nodes.

Each leaf node in a decision tree represents a certain class. The root and other internal nodes represent the attributes and test conditions that are used to classify the records. Hunt's algorithm was the basis of many decision tree algorithms such as ID3, C4.5, CART, and REPTree [13]. REPTree is a decision tree based on a splitting criterion known as the information gain ratio [18]. Equation 1 represents the formula of gain ratio [19, 20].

$$gain\,ratio = \frac{gain}{split\,info} \tag{1}$$

$$gain = entropy(P) - \left[\sum_{i=1}^{k} \frac{n_i}{n} * entropy(i)\right]$$

$$split\ info = -\sum_{i=1}^{k} \frac{n_i}{n} log \frac{n_i}{n}$$

where *k* represents the number of partitions in parent node *P*, n_i is the number of records in the partition *i*

A REPTree classifier may generate large trees and this can lead to overfitting (small training error, but test error is large), limiting the performance of the classifier, and requiring more resources from memory allocation. Therefore, Reduced Error Pruning was applied to solve these issues by decreasing the size of the tree. The measure that is used in pruning is the Mean Square Error (MSE). Equation 2 represents the formula of MSE [21]:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(O_i - P_i)^2 \tag{2}$$

where *n* represents the data points, *O* is the vector of observed values, and *P* is the vector of predicted values

The procedure followed in the pruning process was checking the internal nodes from bottom to top and replacing each internal node whose error is less than that of its child with the most frequent class initiated among its instances. This procedure will keep out to trim the nodes until any further trimming causes a reduction in the accuracy of the tree [22]. As such, the tree of the REPTree classifier is characterized by its accuracy and simplicity.

The wrapper feature selection. The search space contains many possible subsets of a given length. Based on a certain search strategy, various subsets are selected from the space of possible feature subsets. The wrapper method selects a subset from these various subsets of features based on the quality of the performance in the prediction model [23]. This way is designed for a specific learning algorithm [24]. The wrapper method framework is presented in Figure 2 [16].

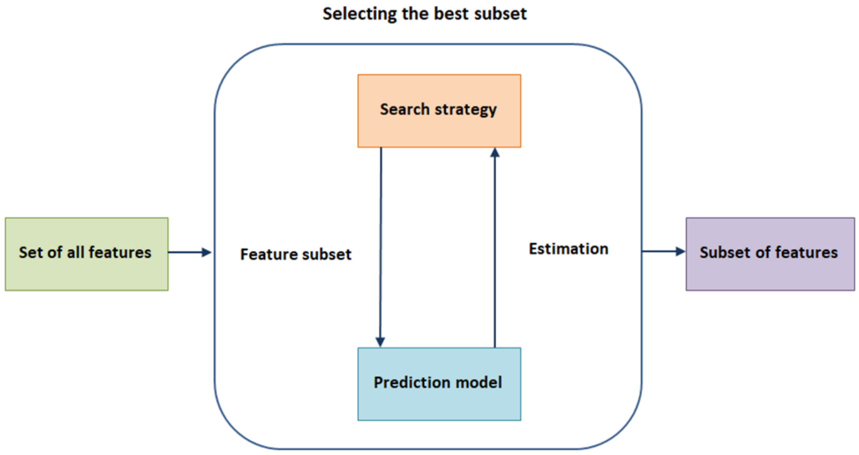


Fig. 2. The wrapper method framework [16]

Wrapper methods are classified into Heuristic Search Algorithms and Sequential Selection Algorithms. Sequential selection algorithms are called sequential because of the nature of adding and deleting features. This algorithm can do a forward or backward selection. The sequential Forward Selection (SFS) algorithm starts with an empty set. The features are added individually in each step to the previous subset (if any). The added feature which can provide the maximum classification accuracy is selected. Therefore, one feature is added in each step and a new subset is formed. The process is repeated until the required number of features is added and the best accuracy is obtained [25].

Figure 3 represents the SFS flowchart. K, d, R, and n represent the length of the current best subset, the required number of features in the best subset, the total number of features that are not mentioned in the current best subset, and the number of added attributes to the current best subset that is survived from the previous stage respectively.

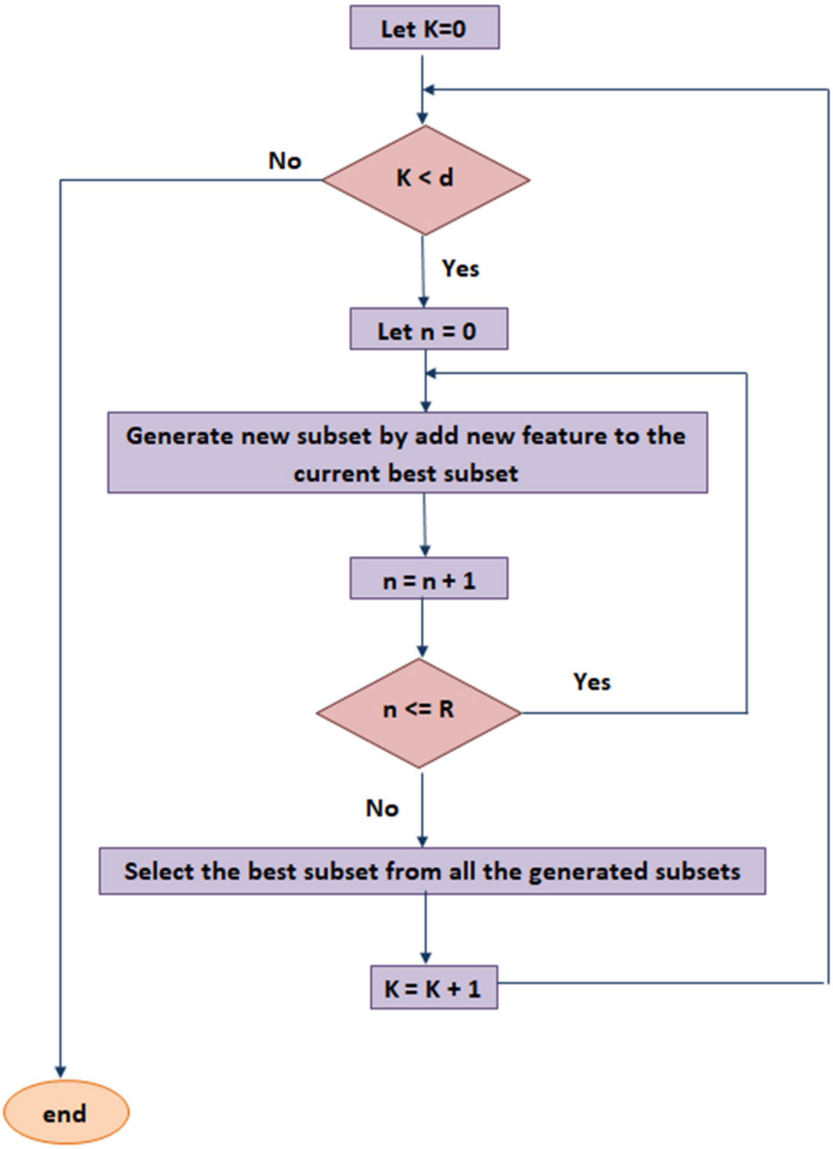

Let K=0
K < d
No
Yes
Let n = 0
Generate new subset by add new feature to the current best subset
n = n + 1
n <= R
Yes
No
Select the best subset from all the generated subsets
K = K + 1
end

Fig. 3. The SFS method framework

In Sequential Backward Selection (SBS), the algorithm starts from a set of all features and then SBS removes the features one by one. The feature that is chosen for deletion is the one that gives the model a performance boost. Deletion of features is stopped as long as there are no additional features that can be removed and that does not decrease the prediction accuracy. This means that the SBS work system is opposite to the SFS work system.

Since the feature subsets in the wrapper method are evaluated using a real learning algorithm, a feature subset with better performance than other methods is obtained, but this feature subset will be biased towards the used learning algorithm. A common drawback of the wrapper method is that it may be computationally intensive, particularly if the evaluated model needs a high computational cost. In this paper, a selection method was developed based on the SFS method by adding the mutation step.

2.2 Previous work

Waheed et al. [26] proposed a system to predict students' performance. Firstly, it predicted students' performance by using students' activities in VLEs and demographic information. This was to find the optimal features impacting students' performance. Secondly, the prediction process was used in four periods using four quarterly clickstream data for each student. This was to obtain an early prediction. The study also proved the effectiveness of the deep learning model in predicting students' academic achievement. Each time, the dataset was arranged based on four categories namely, 'withdrawn-pass', 'pass-fail', 'distinction-pass', and 'distinction-fail'. Results indicated that the students' clickstream activity after the module started and demographic characteristics had an important effect on their performance. Moreover, students' participation in the learning environment before starting the modules had no association with their performance. The first analysis model showed an accuracy of 94.7%, 84.48%, 80.54%, and 86.40 for each category respectively. On the other hand, the second analysis model for the first quarter showed an accuracy of 78.68%, 77.22%, 80.25%, and 80.63% for each category respectively.

Abu Saa et al. [11] analyzed 36 articles that were published between 2009 and 2018. The results showed that the most suitable predicting algorithm which may be used to classify and predict students' factors are decision trees, Naïve Bayes classifiers, and artificial neural networks. Moreover, the results of the analysis indicated that the influencing factors are grouped under four main categories, namely class performance and students' previous grades, students' demographics, students' e-Learning activity, and students' social information.

Sobnath et al. [27] investigated the characteristics of the disabled during and after school years to identify their engagement patterns. Using machine learning principles with the big data approach, this study identified subsets of features useful to construct a predictor which will enhance the chances of engaging disabled school leavers in employment about 6 months after graduation. Features such as institution, age, and disability type were found to be the employment model's base factors. This study also shows that the Logistic Regression models and Decision Tree Classifier obtained the highest accuracy (96%) for predicting the Standard Occupation Classification (SOC) of a disabled.

Aggarwal et al. [28] compared two models: one was built using academic parameters only and another was built on both academic and non-academic (demographic) parameters to predict students' performance. The research concludes that a student's performance depends mainly on academic parameters and demographic (non-academic) parameters. Moreover, the constructed models will be more effective if non-academic parameters are also considered along with academic parameters for predicting students' performance.

3 Research methodology

3.1 The research dataset

In this study, the Open University Learning Analytics Dataset (OULAD) is used. The dataset was published on 28 November 2017 by the Open University (OU). It contains information about three Social Science modules and four Technology, Science, Mathematics, and Engineering (STEM) modules (22 courses and 32,593 students) [6][29]. The students' information was stored in seven tables (students' information, courses, students' registration, assessments, students' assessment, VLE, and students' VLE) [6].

In this research, data from the Science module (DDD) were used. It belongs to students enrolled in the October 2013 presentation which includes 1938 students. To clean this data set, different preprocessing steps were conducted. This enables the application of prediction algorithms. Figure 4 shows the proposed system for this study.

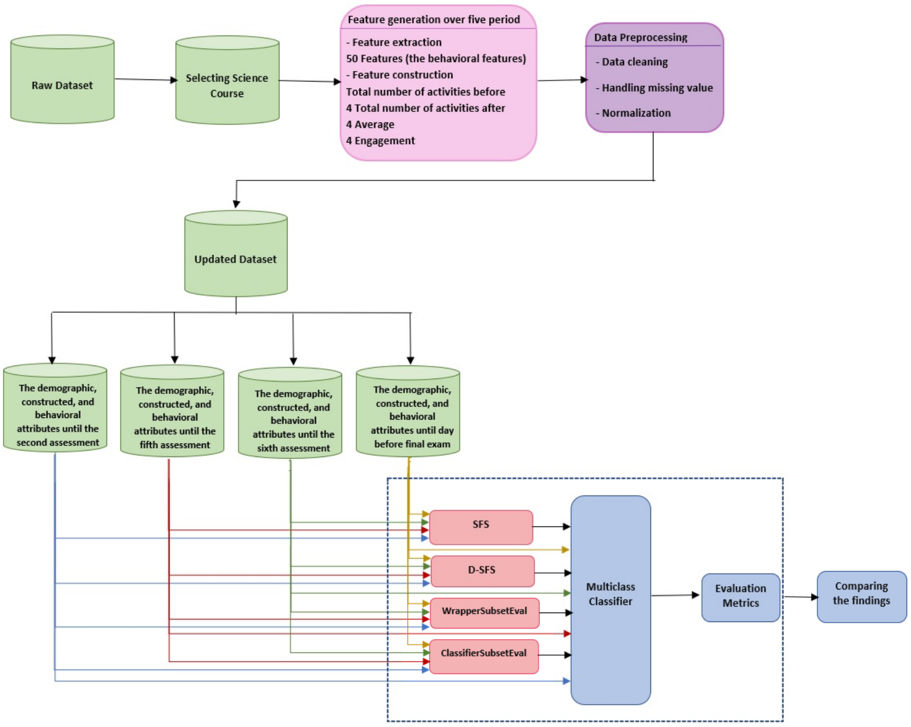

Raw Dataset
Selecting Science Course
Feature generation over five period
- Feature extraction
50 Features (the behavioral features)
- Feature construction
Total number of activities before
4 Total number of activities after
4 Average
4 Engagement
Data Preprocessing
- Data cleaning
- Handling missing value
- Normalization
Updated Dataset
The demographic, constructed, and behavioral attributes until the second assessment
The demographic, constructed, and behavioral attributes until the fifth assessment
The demographic, constructed, and behavioral attributes until the sixth assessment
The demographic, constructed, and behavioral attributes until day before final exam
SFS
D-SFS
WrapperSubsetEval
ClassifierSubsetEval
Multiclass Classifier
Evaluation Metrics
Comparing the findings

Fig. 4. The proposed research model

3.2 Feature generation

Because of the nature of the data in the Open University Learning Analytics Dataset (OULAD), students' interactions with various id sites were randomly saved and not divided into types. More specifically, the ID of the interacted site is saved without mentioning its type and the ID sites belonging to different types of activities were all stored in the same column. It became very difficult to know the extent to which students interacted with each activity type separately. Therefore, feature extraction was used to extract students' interactions with the various VLE activity types. Types of id sites in the department of Science are a resource, content, forum, homepage, subpage, URL, collaborate, glossary, Wiki, and external quiz. Furthermore, based on the original features, four new features were constructed which are the total number of activities before the formal start of the course, the total number of activities after the formal start of the course, the average, and the engagement.

Feature extraction. In this dataset, features are classified into three types: behavioural, demographics, and performance. The behavioural attributes were extracted from the student VLE table. This was carried out by adding students' interaction with each site to the total of their interaction with the type to which this site belongs. The type of site is defined through the use of the VLE table which contains all id sites and the type of each one. Ten types of id sites were found for the course. Therefore, ten new attributes were obtained.

Students' interaction with these activities was calculated at five different intervals. The first was before the formal start of the course. The second was after the second assessment which was after 53 days of the course. The third was after the fifth assessment which was after 165 days. The fourth was after the sixth assessment which was after 207 days. The last was a day before the final exam which was after 260 days of the course. It is worth noting that students' interactions with the different types of sites before the start of the course were used as predicted features in the four predictive periods. This was in addition to the behavioural features of these periods. The rationale is that such features may reflect the extent of students' interest and their incentives. A full description of all features can be found in [6].

Feature construction. New features are generated in this study to enhance the accuracy of the prediction process. These are:

a) The total number of activities: This attribute is calculated from the students' behavioural attributes at five different intervals (before starting the courses, after 53, 165, 207, and 260 days from starting the course).
b) Average: This attribute is generated based on the grades of the assessments that students conducted until the prediction day. Initially, students' grades were extracted for each exam separately. This is because the grades were stored in one column randomly in the raw dataset. Then, the student averages were calculated for each period. Each assessment takes weight at this average according to its weight relative to the rest of the course assessments. In the Science course, the weights of the first, second, third, fourth, fifth, and sixth assessments were 10, 12.5, 17.5, 20, 20, and 20 respectively.
c) Engagement: This attribute reflects the level of students' participation until the prediction day. This feature gave a good indication of how the students' participation affected their final results. Students were divided into three levels based on the number of the predicted classes which were low, moderate, and high engagement levels. Therefore, three values (0, 0.25, and 1) were used to refer to these levels of attribute respectively. It was calculated based on the following developed formula [6].

$$\text{Engagement} = \begin{cases} \rightarrow 0, \text{(average = -1) OR (average = 0 AND total number of activities = 0)} \\ \rightarrow 0.25, \text{(current\_last\_assessment < 50) AND (average < = 65) AND (total number of activities < threshold)} \\ \rightarrow 1, \text{Otherwise} \end{cases}$$

3.3 Data preprocessing

Data cleaning: In this step, features that were not useful in the prediction process such as ID number, code module, and code_presentation were removed.

Handling missing values: In the OULAD dataset, there are missing values for both the assessment scores feature and the deprivation band (imd band) feature. Based on the recommendation of the Open University, which emphasized that it neglected all assessment values of students who did not perform their assessment, a value of –1 was placed in the place of the missing assessments. The reason for choosing the value –1, not zero, is to distinguish the student with a zero score from the one who did not take the exam at all. Moreover, the most frequent value in the deprivation band (imd band) attribute was considered as a substitute for its missing values because it is a discrete attribute and contains a few missing values widely scattered.

Normalization: Normalization was performed for all values of numeric features. This is to ensure that the values of all attributes remained within one range. In this research, a min-max normalization was adopted. Equation 3 is used to calculate the values of the normalized feature [13][30].

$$\hat{V} = \frac{V - min_A}{max_A - min_A} \tag{3}$$

where *V* represents the feature value, *min_A* is the minimum original value for any feature, and *max_A* is the maximum original value for any feature.

3.4 Feature selection

The feature selection method was applied to reduce the dimensionality of feature space, select the important features for the prediction process, and enhance the prediction accuracy. The traditional feature selection methods were applied first. However, a Developed Sequential Feature Selection (D-SFS) method was proposed to reduce the number of features and improve the prediction accuracy.

The developed sequential feature selection (D-SFS) method. Wrapper methods choose features' subsets based on the quality performance of the prediction model. In this thesis, the Bagging model was adopted for evaluating the features of the D-SFS method. This method adds to the SFS algorithm. In this method, the―Strain‖ name was given to each step in which possible subsets of a certain length are generated. Therefore, each Strain consisted of many subsets of a certain length and differed from the length of the subsets in the other Strains. Each subset represented a different feature subset that is used to predict students' performance.

The D-SFS method proposes predicting students' performance based on each feature individually and, thus, the first Strain is formed. In each Strain, a subset or a group of subsets (k subsets to increase the probability of obtaining a higher accuracy) is chosen to form the next Strain. The number of surviving subsets (k) depends on the accuracy of each subset in this Strain. If subsets carry the amount of accuracy equal to the accuracy of the best subset, the number of survivors will be large and vice versa.

After selecting the surviving subsets, the mutation step is performed with the hope of obtaining better subsets. If the accuracy of subsets who survived this Strain exceeds all the accuracy of subsets who survived the Strains that precede them, a mutation is carried out on a single feature. Otherwise, a mutation is made with two features at the same time for the surviving subset.

The mutation on one feature is done by mentioning all the features that are not highlighted in this subset and adding them instead of a feature in a certain location one by one with the hope of obtaining the best subset from the existing subset. If the mutation process produces a new subset better than the mutated subset, the latter will be replaced by the best subset resulting from this mutation process. After that, a mutation on the second surviving subset (if any) is made until all the surviving subsets are completed.

Concerning the mutation on two features, all possible combinations of length 2 are generated from the set of features that are not mentioned in this surviving subset. Then, these groups are replaced instead of two features, group by group, with the hope of obtaining a better subset than the mutated subset. If it is obtained, the original survivor subset is replaced by the best subset resulting from the mutation process. Otherwise, it remains the same.

For determining the position of mutation, a weighting process is applied to the attributes of the mutated subset. This is by calculating the weight of each feature separately based on deleting this particular feature and calculating the difference between the total accuracy before and after its deletion. The difference represents the weight of that attribute. Upon completion of the attribute weighting, a mutation is conducted on one or two attributes with a lower weight among all features. It is worth mentioning that the mutation process starts from the second Strain because the first Strain cannot be improved further. Furthermore, if the mutation is on one feature, the last Strain to be mutated is the Strain that precedes the last Strain. On the other hand, if the mutation is on two characteristics, the last Strain to be mutated is the Strain that precedes the pre-last Strain. This is because there are not enough features for the mutation action.

After completing the mutation process, the next Strain is created based on the survivors of the previous Strain after performing a mutation operation. The new Strain is built by browsing each one of these subsets and obtaining the features (n) that are not included in this subset and adding them to this subset one by one. Each adding process produces a new subset to the new Strain. In the new Strain, surviving subsets are chosen to contribute to the creation of the next Strain as happened earlier. The number of the total generated Strains is (s), where s is the number of the total features in the original dataset. The length of the subset in each Strain is the number of Strain +1 in which one is the class to be predicted. The number of subsets in each Strain is a product of n*k, where n is the number of features that are currently available for adding to the surviving subset. It should be mentioned that n in the first Strain is equal to s, while in the second Strain, it equals s –1 … etc. In other words, n is the difference between the total number of features and the number of features in the survivor subset. Moreover, k is the number of survivor subsets.

After the completion of the generation Strain process, the weighting process is re-applied to the subset (subsets) of features that have the highest accuracy across all Strains. Attributes with zero or less weight are deleted and the remaining features are re-weighted again. This is kept until obtaining a group of features with weights greater than zero. At the end:

- Attempting to take into account the extent of interconnection between features and their effect on raising accuracy
- Trying to introduce new attributes for the subset to obtain rid of the stability of the features resulting from the sequential addition of the attributes
- Obtaining rid of the features that may have impaired prediction accuracy, which has been (in the previous Strains) proven as good features due to the sequential addition process.

Figure 5 shows the main steps of the D-SFS method. Algorithm 1 illustrates the D-SFS process.

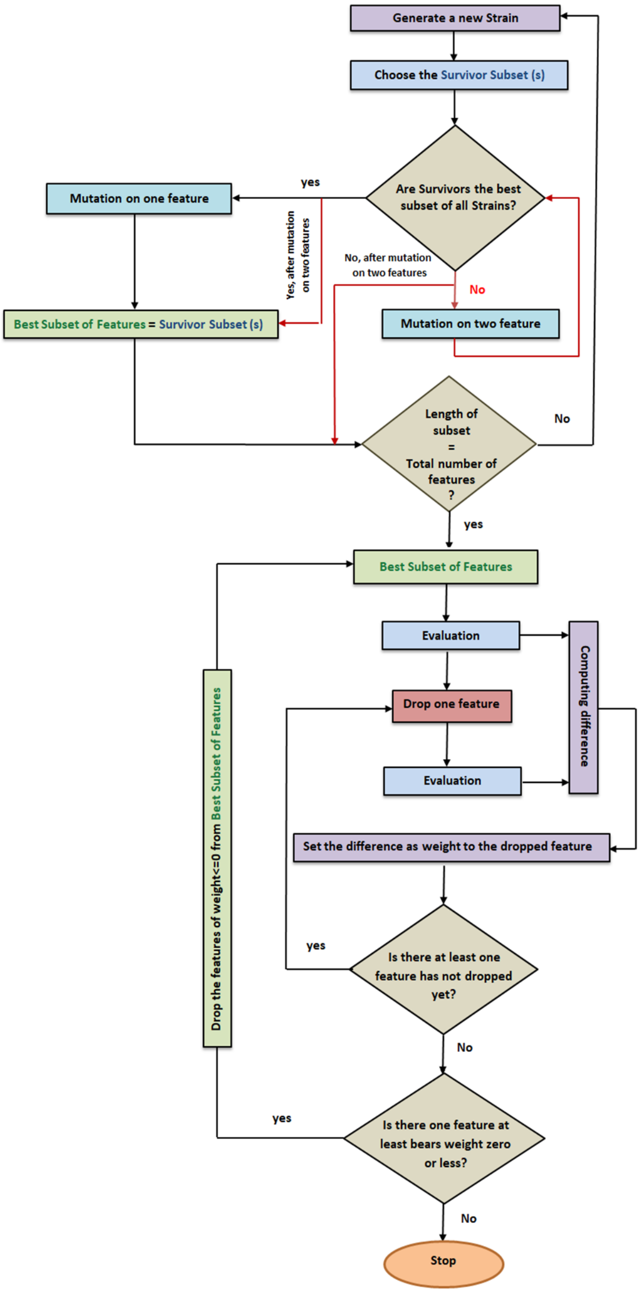

Generate a new Strain
Choose the Survivor Subset (s)
Are Survivors the best subset of all Strains?
yes
Mutation on one feature
Yes, after mutation on two features
No, after mutation on two features
No
Mutation on two feature
Best Subset of Features = Survivor Subset (s)
Length of subset = Total number of features ?
No
yes
Best Subset of Features
Evaluation
Computing difference
Drop one feature
Evaluation
Set the difference as weight to the dropped feature
Drop the features of weight<=0 from Best Subset of Features
yes
Is there at least one feature has not dropped yet?
No
yes
Is there one feature at least bears weight zero or less?
No
Stop

Fig. 5. The developed sequential feature selection method (D-SFS)

Algorithm 1: The Developed Sequential Feature Selection (D-SFS) Method

Input: attributes [m], instances [n*m] where m is the number of features and n is the number of instances

Output: best_subset [d*z] where d is the number of best-surviving subsets among all Strains and z is the number of features in each subset.

```
Let new_features [], best_subset [][], survivor [][], new_Strain [][] empty arrays at the beginning
max_acc_best_subset = 0, acc_ survivor = 0
  for i = 1 to m-1
      j = 0
      do
          new_features = find_new_features (survivor [j], attributes)
          new_Strain += build_ Strain (survivor [j], new_features)
          j = j+1
     while j <= number of rows in the survivor array
     computing _ accuracy_ Strain (new_Strain, instances)
     acc_survivor = find_survivor (new_ Strain, survivor)
      If max_acc_best_subset < acc_survivor
         mutation_one (survivor)
         best_subset = survivor
         max_acc_best_subset = The greatest accuracy of the survivors after the mutation process
      else
         mutation_two (survivor)
         If max_acc_best_subset < The greatest accuracy of the survivors after the mutation process
                best_subset = survivor
               max_acc_best_subset = The greatest accuracy of the survivors after the mutation
               process
         end if
    end if
   new_ Strain = null
  end for i
    for i = 1 to d
      flag=0
     for j= 1 to z-1
        delete feature j from best_subset [i]
        weight [i][j] = the difference in accuracy between before and after
        delete the feature j
        if weight [i][j] > 0 then
            flag = flag + 1
        end if
      end for j
        if flag < z-1
            Drop all the features with a weight less than or equal to zero
            i = i - 1
        end if
    end for i
```

3.5 Building and testing the predictive model

The highest accuracy for predicting students' final results was obtained by the Bagging method. Furthermore, this research used a 3-fold cross-validation technique to train and test the data. This technique divides the dataset into K subsets (in this paper three subsets) of equal size. It consists of K (three) stages. In each stage, all subsets are used for training except for one subset which is used for testing. This method is implemented, where each partition is used exactly once for testing.

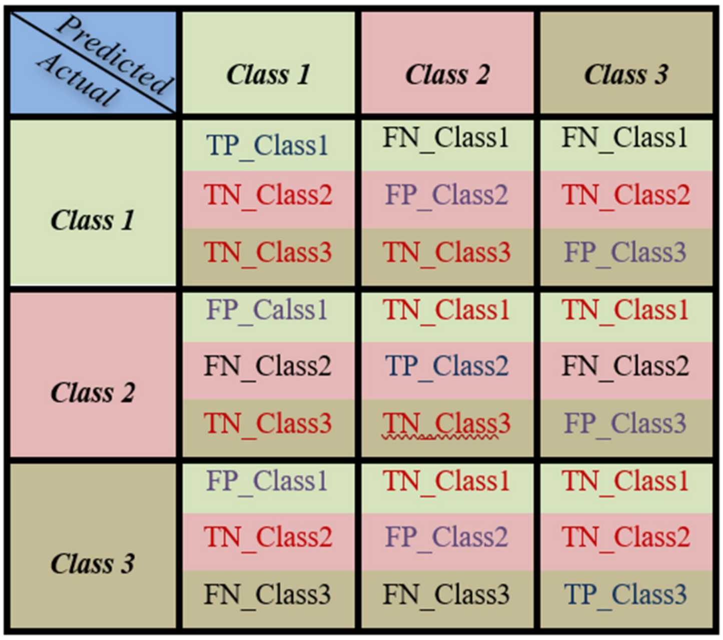
Evaluation measures. The performance metrics are used to evaluate the generalization power of the trained model. This encompasses accuracy, F1-measure, precision, and recall. However, one of the most common metrics to evaluate the generalization power of models is accuracy [31]. Through accuracy, the trained model is evaluated based on the total instances that are correctly predicted by the trained model when it is tested with the unseen data. The accuracy is the number of correct predictions divided by the total number of predictions. The accuracy can be computed based on Equation 4 [13].

$$\text{Accuracy} = (TP + TN) \,/\, TP + TN + \sum FP + \sum FN \tag{4}$$

The calculation of such measures is based on computing the confusion matrix. This matrix summarizes the number of instances wrongly or properly predicted by a classification model. In this paper, the predicted classes are three, so the form of the confusion matrix is as shown in Table 1, where:

1. True Positive (TP): The instances that are correctly classified.
2. False Negative(FN): The instances that are wrongly classified.
3. False Positive (FP): The instances that are wrongly classified.
4. True Negative (TN): The instances that are correctly classified.

Table 1. The three-dimensional confusion matrix



Predicted Actual	*Class 1*	*Class 2*	*Class 3*
Class 1	TP_Class1 TN_Class2 TN_Class3	FN_Class1 FP_Class2 TN_Class3	FN_Class1 TN_Class2 FP_Class3
Class 2	FP_Calss1 FN_Class2 TN_Class3	TN_Class1 TP_Class2 TN_Class3	TN_Class1 FN_Class2 FP_Class3
Class 3	FP_Class1 TN_Class2 FN_Class3	TN_Class1 FP_Class2 FN_Class3	TN_Class1 TN_Class2 TP_Class3

4 Results and discussion

This study aims at predicting students' performance in an online learning environment, particularly VLEs. Four prediction models were built over four time periods for the course which are the second, fifth, and sixth assessments as well as immediately before the final exam. It was performed in such different periods to provide a continuous indicator of students' final results if they would stay at the same present academic level. The total number of the categorized instances was the 1938 records (student). Figure 6 depicts the actual number of successful, failed, and withdrawn students in the course.

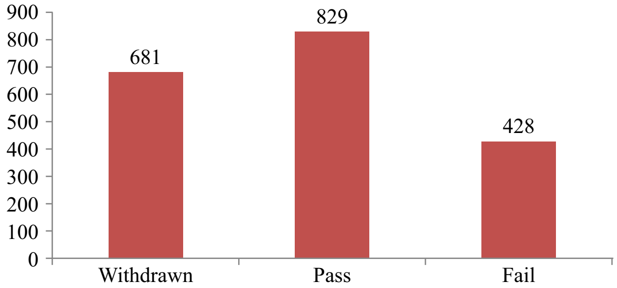


Fig. 6. The actual number of successful, failed and withdrawn students in the course

These instances were used to construct the proposed model and evaluate it by using the 3-fold cross-validation technique. The model was built based on the Bagging method in the reality of three REPTree classifiers in each model. This way was used to build the models to obtain the highest possible accuracy. Demographic and behavioural alongside the constructed features were used in the classification process to evaluate the prediction accuracy of the models. A comparison was made for the four-time periods between the model's accuracy once without using a feature selection method and once with the use of SFS and D-SFS methods. The results of this comparison are shown in Table 2. Table 3 illustrates a comparison between the results of the proposed method and the results of the subset evaluation methods for the four prediction stages of the course. Table 4 highlights the features that were chosen after applying the D-SFS method to the course attributes. It is clear from the comparisons that the proposed method outperformed in terms of the prediction model accuracy of the rest of the methods implemented.

Table 2. A comparison of the prediction accuracy of the course final results without feature selection along with SFS and D-SFS methods

Date of Prediction	Accuracy without Feature Selection	Used Features	Accuracy with SFS	Used Features	Accuracy with D-SFS	Used features
First Prediction	68.7822%	31	72.0330%	13	72.6006%	12
Second Prediction	82.5593%	31	84.4685%	11	84.8297%	11
Third Prediction	87.9257%	31	89.2672%	12	89.5252%	11
Last Prediction	90.9185%	31	92.4664%	16	92.5180%	7

Table 3. A comparison between the subset evaluation methods and the D-SFS method in the accuracy of the fourth prediction periods

The Method	Search Method	Accuracy	Number of Attributes	Date of Prediction
ClassifierSubsetEval	GreedyStepwise	68.4727%	5	First Prediction
ClassifierSubsetEval	BestFirst	67.0795%	6	First Prediction
WrapperSubsetEval	GreedyStepwise	69.969%	4	First Prediction
WrapperSubsetEval	BestFirst	70.485%	8	First Prediction
D-SFS		72.6006%	12	First Prediction
ClassifierSubsetEval	GreedyStepwise	82.3529%	7	Second Prediction
ClassifierSubsetEval	BestFirst	82.3529%	7	Second Prediction
WrapperSubsetEval	GreedyStepwise	83.1269%	3	Second Prediction
WrapperSubsetEval	BestFirst	83.1269%	3	Second Prediction
D-SFS		84.8297%	11	Second Prediction
ClassifierSubsetEval	GreedyStepwise	87.8225%	7	Third Prediction
ClassifierSubsetEval	BestFirst	87.8225%	7	Third Prediction
WrapperSubsetEval	GreedyStepwise	87.9773%	3	Third Prediction
WrapperSubsetEval	BestFirst	88.0289%	5	Third Prediction
D-SFS		89.5252%	11	Third Prediction
ClassifierSubsetEval	GreedyStepwise	90.5573%	6	Last Prediction
ClassifierSubsetEval	BestFirst	90.5573%	6	Last Prediction
WrapperSubsetEval	GreedyStepwise	90.8669%	7	Last Prediction
WrapperSubsetEval	BestFirst	90.8669%	7	Last Prediction
D-SFS		92.5180%	7	Last Prediction

Table 4. The selected features of the Science course in the D-SFS method

Date of Identifying the Features	The Selected Attributes by the D-SFS Method	Weights
After 53 days	Average	3.2507
	Gender	2.2703
	Collaborate	2.0123
	External quize	1.5479
	URL before	0.8255
	Num_prev_attempts	0.6191
	Engagement	0.5159
	Total of activities	1.5479
	Glossary	0.4127
	Disability	0.3095
	Age	0.2063
	Collaborate before	0.1547
After 165 days	Average	17.6986
	Homepage before	1.0835
	Collaborate	1.1867
	Age	0.7223
	Num_prev_attempts	0.4127
	Disability	0.6191
	Forum before	0.4643
	Gender	0.5675
	Collaborate before	0.4643
	Total of activities	0.6707
	Glossary before	0.1547
	Average	4.2827
	Total before	0.8771
	URL before	0.8771
	Num_prev_attempts	0.1031
	Glossary	0.4643
	Resource	0.9287
	Content before	1.0319
	Gender	0.6191
	Age	0.6191
	Disability	0.3611
	Engagement	0.2579
	Average	8.5139
	Forum	0.9803
	Collaborate	1.3415
	Resource	0.2063
	Total before	0.7223
	URL	1.2899
	Num_prev_attempts	0.2579

Figure 7 presents a visual comparison between the results obtained in the first and last prediction stages with the actual results of the course.

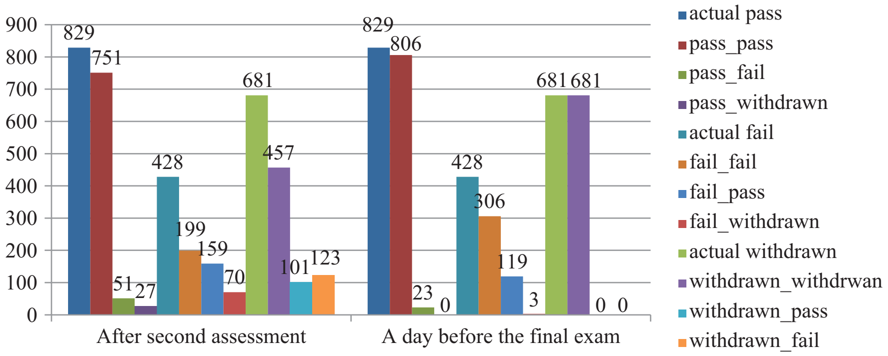


Fig. 7. A comparison between actual and predicted results of the last and first prediction in the Science course

Because of the differences in the selected features across the four prediction periods and their instability, all features that appeared across the periods as influencing features were taken. Then, they were entered into the model to ensure their importance as well as the extent of accuracy that would be reached by using these features. The total influencing features across the four time periods for the Science course are Average, Gender, Collaborate, External quiz, URL before, Num_prev_attempts, Engagement, Total of activities, Glossary, Disability, Age, Collaborate before, Homepage before, Forum before, Glossary before, Total before, Resource, Content before, Forum, and URL.

Using the characteristics above, an accuracy of 91.8989% was obtained. The obtained overall accuracy is close to the final accuracy obtained when applying the proposed feature selection algorithm based on the latest prediction period. This means that the characteristics obtained are the most effective features that are recommended in the Science online course. Thus, the problem of feature fluctuation across different prediction periods was solved.

5 Conclusion

This research aimed at proposing a new feature selection method to predict students' performance on VLEs. Based on the research outcomes, many conclusions can be drawn. First, the quality of online and virtual education can be improved by highlighting features that can have a high influence on learners' performance. Second, the use of feature selection methods can improve prediction accuracy and significantly reduce the number of features. Thus, through the features that will be selected as important features in the predicting process, the activities that the student must focus on during the course can be determined. Third, the features that affect the student's level may vary across different course periods. However, important features that may affect students' performance in the Science course were Average, Gender, Collaborate, External quiz, URL before, Num_prev_attempts, Engagement, Total of activities, Glossary, Disability, Age, Collaborate before, Homepage before, Forum before, Glossary before, Total before, Resource, Content before, Forum, and URL. Finally, educational institutions need to provide permission for learners and encourage them to interact with various activities that exist within the learning environment before starting the course. Such features affect learners' academic achievement where they appeared as influential features over the four prediction periods. Thus, educational institutions and policy-makers need to consider different activities and learning materials to ensure providing successful online educational settings.

6 Acknowledgment

The authors would like to express their great gratitude to Al-Mustaqbal University College for sponsoring this research.

7 References

[1] A. Jović, K. Brkić, and N. Bogunović, "A review of feature selection methods with applications," in 2015 38th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), 2015, pp. 1200–1205. https://doi.org/10.1109/MIPRO.2015.7160458

[2] Y. Liu, Q. Pan, and Z. Zhou, "Improved Feature Selection Algorithm for Prognosis Prediction of Primary Liver Cancer," in 2nd International Conference on Intelligence Science (ICIS), 2018 Beijing, China, November 2–5, 2018, pp. 422–430. https://doi.org/10.1007/978-3-030-01313-4_45

[3] J. Miao and L. Niu, "A survey on feature selection" Procedia Comput. Sci., vol. 91, pp. 919–926, 2016. https://doi.org/10.1016/j.procs.2016.07.111

[4] P. S. Muljana and T. Luo, "Factors contributing to student retention in online learning and recommended strategies for improvement: A systematic literature review," J. Inf. Technol. Educ. Res., vol. 18, pp. 19–57, 2019. https://doi.org/10.28945/4182

[5] A. Al-Azawei and H. M. Habeeb, "Which factors affect learner satisfaction in educational hypermedia systems? A case study of the moodle system," J. Eng. Appl. Sci., vol. 12, no. Specialissue10, pp. 8864–8874, 2017. https://doi.org/10.3923/jeasci.2017.8864.8874

[6] A. Al-Azawei and M. A. A. Al-Masoudy, "Predicting learners' performance in virtual learning environment (VLE) based on demographic, behavioral and engagement antecedents," Int. J. Emerg. Technol. Learn., vol. 15, no. 9, pp. 60–75, 2020. https://doi.org/10.3991/ijet.v15i09.12691

[7] M. Hussain, W. Zhu, W. Zhang, and S. M. R. Abidi, "Student engagement predictions in an e-learning system and their impact on student course assessment scores," Hindawi, vol. 6, pp. 1–22, 2018. https://doi.org/10.1155/2018/6347186

[8] A. M. Mogus, I. Djurdjevic, and N. Suvak, "The impact of student activity in a virtual learning environment on their final mark," Active Learning in Higher Education, vol. 13, no. 3, pp. 177–189, 2012. https://doi.org/10.1177/1469787412452985

[9] A. Al-Azawei, "Modelling e-learning adoption: The influence of learning style and universal learning theories," December 2016, 2019.

[10] A. Al-Azawei, P. Parslow, and K. Lundqvist, "The effect of universal design for learning (UDL) application on e-learning acceptance: A structural equation model," Int. Rev. Res. Open Distance Learn., vol. 18, no. 6, pp. 54–87, 2017. https://doi.org/10.19173/irrodl.v18i6.2880

[11] A. Abu Saa, M. Al-Emran, and K. Shaalan, "Factors affecting students' performance in higher education: A systematic review of predictive data mining techniques," Technology, Knowledge and Learning, vol. 24, no. 4, pp. 567–598, 2019. https://doi.org/10.1007/s10758-019-09408-7

[12] P.-N. Tan, M. Steinbach, A. Karpatne, and V. Kumar, "A review on predicting student's performance using data mining techniques," Procedia – Procedia Comput. Sci., vol. 72, pp. 414–422, 2015. https://doi.org/10.1016/j.procs.2015.12.157

[13] P. Tang, M. Steinbach, and V. Kumar, "Introduction to data mining", Pearson Education, Inc., 2006.

[14] M. Imran, S. Latif, D. Mehmood, and M. S. Shah, "Student academic performance prediction using supervised learning techniques," Int. J. Emerg. Technol. Learn., vol. 14, no. 14, pp. 92–104, 2019. https://doi.org/10.3991/ijet.v14i14.10310

[15] P. Bühlmann, "Bagging, boosting and ensemble methods," in Handbook of Computational Statistics, Springer, 2012, pp. 985–1022. https://doi.org/10.1007/978-3-642-21551-3_33

[16] T. A. H. Hameed, "Fuzzy feature selection and adaptive random forest to improve the prediction of higher education students completion and earning," Master Thesis, University of Babylon, 2019.

[17] E. K. M. Al-Yasiri, "Arabic sentiment analysis for identifying terrorism supporters on twitter using data mining techniques," 2019.

[18] D. T. Bui et al., "A hybrid intelligence approach to enhance the prediction accuracy of local scour depth at complex bridge piers," Sustain., vol. 12, no. 3, 2020. https://doi.org/10.3390/su12031063

[19] S. Mukherjee and N. Sharma, "Intrusion detection using naive bayes classifier with feature reduction," Procedia Technol., vol. 4, pp. 119–128, 2012. https://doi.org/10.1016/j.protcy.2012.05.017

[20] V. Podgorelec, P. Kokol, B. Stiglic, and I. Rozman, "Decision trees: An overview and their use in medicine," J. Med. Syst., vol. 26, no. 5, pp. 445–463, 2002. https://doi.org/10.1023/A:1016409317640

[21] J. Devore, "A modern introduction to probability and statistics: Understanding why and how," vol. 101, no. 473, 2006. https://doi.org/10.1198/jasa.2006.s72

[22] M. Belouch, S. El, and M. Idhammad, "A two-stage classifier approach using RepTree algorithm for network intrusion detection," Int. J. Adv. Comput. Sci. Appl., vol. 8, no. 6, pp. 389–394, 2017. https://doi.org/10.14569/IJACSA.2017.080651

[23] E. Corchado, H. Yin, V. Botti, and C. Fyfe (Eds.), "Intelligent data engineering and automated learning—IDEAL 2006," 2006. https://doi.org/10.1007/978-3-642-32639-4

[24] M. Walowe Mwadulo, "A review on feature selection methods for classification tasks," Int. J. Comput. Appl. Technol. Res., vol. 5, no. 6, pp. 395–402, 2016. https://doi.org/10.7753/IJCATR0506.1013

[25] G. Chandrashekar and F. Sahin, "A survey on feature selection methods," Comput. Electr. Eng., vol. 40, no. 1, pp. 16–28, 2014. https://doi.org/10.1016/j.compeleceng.2013.11.024

[26] H. Waheed, S. Ul Hassan, N. R. Aljohani, J. Hardman, and R. Nawaz, "Predicting academic performance of students from VLE big data using deep learning models Hajra," Elsevier, vol. 53, no. 9, pp. 1689–1699, 2019. https://doi.org/10.1017/CBO9781107415324.004

[27] D. Sobnath, T. Kaduk, I. U. Rehman, and O. Isiaq, "Feature selection for UK disabled students' engagement post higher education: A machine learning approach for a predictive employment model," IEEE Access, vol. 8, pp. 159530–159541, 2020. https://doi.org/10.1109/ACCESS.2020.3018663

[28] D. Aggarwal, S. Mittal, and V. Bali, "Significance of non-academic parameters for predicting student performance using ensemble learning techniques," International Journal of System Dynamics Applications, vol. 10, no. 3, pp. 38–49, 2021. https://doi.org/10.4018/IJSDA.2021070103

[29] J. Kuzilek, M. Hlosta, and Z. Zdrahal, "Data descriptor: Open university learning analytics dataset," Sci. Data, vol. 4, pp. 1–8, 2017. https://doi.org/10.1038/sdata.2017.171

[30] L. Al Shalabi and Z. Shaaban, "Normalization as a Preprocessing Engine for Data Mining and the Approach of Preference Matrix," Proc. Int. Conf. Dependability Comput. Syst. DepCoS-RELCOMEX 2006, pp. 207–214, 2006. https://doi.org/10.1109/DEPCOS-RELCOMEX.2006.38

[31] M. Hossin, and M. N. Sulaiman, "A review on evaluation metrics for data classification evaluations," Int. J. Data Min. Knowl. Manag. Process, vol. 5, no. 2, pp. 01–11, 2015. https://doi.org/10.5121/ijdkp.2015.5201

8 Authors

Miami Abdul Aziz Al-Masoudy is an assistant lecturer in the Medical Instrumentation Techniques Engineering Department, Al-Mustaqbal University College, Iraq. She received her MSc degree in Computer Science from the University of Babylon, Iraq. Miss Al-Masoudy has published many papers in international journals. Her research interest focuses on data mining and data analysis. Miss Al-Masoudy can be contacted via this email: miami.abdulaziz@uomus.edu.iq.

Ahmed Al-Azawei is an assistant professor in the College of Information Technology at the University of Babylon, Iraq. He received his PhD degree in Computer Science from the University of Reading, UK. Dr Al-Azawei has published many papers in international journals and conferences. His research interest focuses on information systems, educa-tional technologies, Web technologies, social media sites, and data mining. Dr Al-Azawei can be contacted via this email: ahmedhabeeb@itnet.uobabylon.edu.iq.

Article submitted 2022-09-18. Resubmitted 2023-04-05. Final acceptance 2023-04-06. Final version published as submitted by the authors.