IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received June 26, 2019, accepted July 7, 2019, date of publication July 29, 2019, date of current version August 12, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2930867

Predicting Students' Academic Procrastination in Blended Learning Course Using Homework Submission Data

AFTAB AKRAM[1,2], CHENGZHOU FU[1,3], YUYAO LI[1,5,6], MUHAMMAD YAQOOB JAVED[4], RONGHUA LIN[1], YUNCHENG JIANG[1], AND YONG TANG[1]

[1]School of Computer Science, South China Normal University, Guangzhou 510631, China
[2]Department of Computer Science and Technology, University of Education, Lahore 54770, Pakistan
[3]College of Medical Information Engineering, Guangdong Pharmaceutical University, Guangzhou 510006, China
[4]Department of Electrical and Computer Engineering, COMSATS University Islamabad (CUI), Lahore Campus, Lahore 54000, Pakistan
[5]School of Information Science, Guangdong University of Foreign Studies, Guangzhou 510420, China
[6]School of Cyber Security, Guangdong University of Foreign Studies, Guangzhou 510420, China

Corresponding author: Yong Tang (ytang@m.scnu.edu.cn)

This work was supported by the National Natural Science Foundation of China under Grant U1811263, Grant 61772210, and Grant 61772211.

ABSTRACT Academic procrastination has been reported affecting students' performance in computer-supported learning environments. Studies have shown that students who demonstrate higher procrastination tendencies achieve less than the students with lower procrastination tendencies. It is important for a teacher to be aware of the students' behaviors especially their procrastination trends. EDM techniques can be used to analyze data collected through computer-supported learning environments and to predict students' behaviors. In this paper, we present an algorithm called students' academic performance enhancement through homework late/non-submission detection (SAPE) for predicting students' academic performance. This algorithm is designed to predict students with learning difficulties through their homework submission behaviors. First, students are labeled as procrastinators or non-procrastinators using k-means clustering algorithm. Then, different classification methods are used to classify students using homework submission feature vectors. We use ten classification methods, i.e., ZeroR, OneR, ID3, J48, random forest, decision stump, JRip, PART, NBTree, and Prism. A detailed analysis is presented regarding performance of different classification methods for different number of classes. The analysis reveals that in general the prediction accuracy of all methods decreases with increase in the number of classes. However, different methods perform best or worst for different number of classes.

INDEX TERMS Blended learning, computer-assisted learning, educational data mining as an inquiry method, e-learning, higher education, learning management systems, online learning.

I. INTRODUCTION

On-line learning management system (LMS) provides a flexible and efficient way to promote beyond classroom interactions between students and teachers. Also, such systems can store abundant data regarding students' and teachers' interactions with the system. Reference [1] termed it as the gold mine of educational data. Besides its benefits, the on-line LMSs also rise pedagogical challenges for teachers, since many students fail to adapt to the requirements of on-line learning environments. Many studies have reported that students face challenges while they are learning through on-line LMS due to various factors, e.g., LMS require more effort by the students, learning through LMS is self-paced and self-regulated, keeping pace with the fast learning through LMS is difficult for many students, and many students do not successfully adapt to learning through LMS [2], [3]. References [4] and [5] reported that procrastination is frequently observed behavior in on-line learning environments. References [6] and [7] have reported the negative effect of procrastination on students' achievement. By applying state-of-the-art data mining and machine learning techniques on students' behavioral data acquired from LMS logs, algorithms can be built which can be used to predict students' future behaviors [8].

Educational data mining (EDM) has emerged as a discipline to analyze data arising from educational settings.

The associate editor coordinating the review of this manuscript and approving it for publication was Jerry Chun-Wei Lin.

Using data mining and machine learning techniques, the researchers can understand the students and their learning environments. The data collected through the on-line LMS can be used for this purpose. This data is about the interactions of the students with the system, e.g., how often students have used the system, what are the task submission trends, how often the students have participated in course forums, how often certain courses or help materials have been accessed or downloaded by the students, etc. By applying EDM techniques and methods, the teachers can get the useful insight into the learning process of the students and can take appropriate decisions which will eventually help the students facing learning difficulties.

In this paper, we build an algorithm to predict students' procrastination behaviors. The data used in this study was collected from an on-line blended learning course conducted through SCHOLAT Course, SCHOLAT on-line LMS. SCHOLAT[1] is an academic social networking system, designed to promote collaborations between the researchers, the teachers and the students. Through its various features, the research community can communicate and collaborate with each other. SCHOLAT Course is on-line LMS, where teachers and students can interact beyond classroom settings. In this blended learning course, besides other on-line learning activities, the students were required to submit their tasks or homework through on-line homework submission page. Submitting homework with hard deadlines is a part of the course assessment scheme. Using students' homework submission data, a feature vector is built to represent students' homework submission behaviors. This feature vector is then used to analyze students' procrastination behaviors and to build an algorithm to predict students' procrastination tendencies. The students are first labeled as procrastinators or non-procrastinators by applying k-means clustering on feature vector. Then, ten classification methods, i.e., ZeroR, OneR, ID3, J48, random forest, decision stump, JRip, PART, NBTree and Prism are used to classify students. The performance of the classification methods is compared for different number of classes. Finally, using these result, an algorithm, i.e., Students' Academic Performance Enhancement through homework late/ non-submission detection (SAPE) is built.

The rest of the paper is organized as follows: section II describes brief related work, section III presents the research question proposed in this study, section IV describes the data and methods used for analysis, section V presents the results of the analysis, in section VI a brief discussion is presented on the implications of the results, and finally section VII presents conclusion and some directions of future work.

II. RELATED WORK

In this section, a brief overview of studies related to academic procrastination and its impact on students' performance is presented. Generally, procrastination is a characteristic behavior trait and mostly people habitually delay tasks. References [9] and [10] defined procrastination as the compulsion of delaying tasks until the point of discomfort. People in general unknowingly postpone starting or completion of tasks thinking that the task can be done later but eventually end up failing to complete the task. Similar behavior is demonstrated by the students while performing academic tasks. References [11] and [12] defined academic procrastination as failure to complete academic tasks in due time. Majority of students procrastinate, i.e., [13] reported that 80%-95% students at college or university level procrastinate. Reference [10] reported that 50% students are consistent procrastinators.

In many studies, the greater procrastination tendencies are linked to lower achievements, poor goal achievement and planning abilities. Reference [6] reported negative relationship between academic procrastination and students achievement. Some other studies, e.g., [7], [14]–[16] also reported the similar outcome of academic procrastination on course achievements. In their study, [17] verified the hypothesis that low academic procrastination was linked to high grade point averages. Reference [18] studied negative relationship between academic procrastination and goal achievement. Reference [19] found out that students with higher procrastination tendencies fail to plan their academic goals. Reference [20] investigated the relationship between academic procrastination and misconduct. They found out that academic procrastination affected the frequency of six different forms of academic misconduct, i.e., using fraudulent excuses, plagiarism, copying in exams, copying homework, and fabrication of data. The students with higher procrastination tendencies tend to use these tactics more often than the students with lower procrastination tendencies. Reference [21] discussed that academic procrastination affects self- efficacy, self-control and organizational behaviors of the students and they eventually achieve lower academic scores. In these and many other studies, it has been emphasized that the procrastination is one such critical phenomenon that effects students' learning and it is deeply rooted in their behaviors. Before effecting their academic achievement, it effects different behavioral aspects, for example, the student loses self-control. He cannot manage his time and plan his activities. Therefore, it is necessary to timely detect if some student is having difficulty with his learning especially if the student is exhibiting procrastination tendency. Many different types of academic and behavioral indicators have been used to detect procrastination. For example, delaying or not submitting academic tasks, self-reports, assessment scales, and questionnaires. In on-line learning environments, it is much easier to get data about students tasks submissions. This data can be used effectively to analyze the procrastination tendencies of students. In this study, we use the homework submission data of students to build a feature vector of their homework submission behaviors. This feature vector can be used to characterize each student as procrastinator or non-procrastinator. Generally, classification and clustering are used for labeled and unlabeled

[1] www.SCHOLAT.com

datasets, respectively. Clustering can be used to label an unlabeled dataset. Classification predicts category or class of an instance using a classification model trained on labeled dataset.

Classification is a commonly used data mining technique to assign an object to a class or category. Classification is a supervised learning technique, i.e., the training dataset used to train the model has labeled classes or categories. Using this labeled dataset a classifier is built. A classifier is a model that predicts the class of an object from other explanatory variables. In education, the classification is used to classify students based on their knowledge, motivation, and behavior [22]. Among different classification techniques, decision trees are the best-known classification paradigm. The decision trees are simple and very easy to understand. They can handle numerical and nominal attributes. Decision trees have high representative powers [22]. In present study, ten methods, i.e., ZeroR, OneR, ID3, J48, random forest, decision stump, JRip, PART, NBTree and Prism are used for analysis. A decision tree represents a set of classification rules in a tree form [23]. Each root-leaf path corresponds to a rule of form $T_{i1}, T_{i2}, \wedge ..... \wedge T_{il} \rightarrow (C = c)$, where c is the class value in the leaf and each T_{il} is a Boolean-valued test on attribute A_{ij}. ID3 [24] and C4.5 [25] are two best-known decision tree algorithms. J48 is an implementation of C4.5 in WEKA [26]. Random forest is a tree -structured classifier which is used as a combination of large number of trees. In this technique, a large of trees are constructed using data sampled from main dataset. Once, these trees have been built, each tree produces a classification and votes for instance being in one class or other. The true class of the instance is the one voted by most of the trees [27]. JRip was proposed by [28]. This algorithm implements a proportional rule learner called Repeated Incremental Pruning to Produce Error Reduction (RIPPER). The algorithm is implemented in four stages namely, rule, pruning, optimization and selection stages. PART algorithm uses a separate-and-conquer strategy. Each iteration of the algorithm builds a partial C4.5 decision tree and selects best leaf as final node [29]. Prism classifier is based on ID3 classifier and can only deal with nominal attributes. This classifier does not do any pruning and cannot handle missing values. However, it does has some advantages over classic ID3 classifiers by introducing modular rules [30]. Decision stump is more sophisticated classifier which can handle missing values by treating them as separate values. This classifier is used for regression analysis or classification using entropy as validation criterion. Decision stump is implemented in conjunction with boosting algorithms like Adaboost [31]. NBTree is a hybrid algorithm which combines decision tree with Naive Bayes classifiers. NBTree generates decision trees based on Naive Bayes classifiers at the leaves. The NBTree algorithm was designed to scale up to the accuracy Naive Bayes classifiers [32].

In different studies, decision trees and other data mining methods have been used to predict students' performance and learning behaviors. Also, different variables from variant sources have been used as explanatory variables. Reference [33] used C5.0, CART, SVM and neural networks to classify teachers according to their performances. Reference [34] in his study used C4.5, ID3, CART, and CHAID decision trees to predict students' performance. The data in his study was collected through questionnaires. Different demographic and academics related variables were used in the analysis. The author concluded that the academic performance of students is not totally dependent on their academic efforts, there are other factors, for example, their background, income, mother's occupation also have an impact on their learning. Reference [35] used J48 and four other classification algorithms to predict slow learners. They used 13 variables related to high school to examine whether or not the student qualifies as a slow learner. The maximum accuracy of 75% was achieved in their analysis. Reference [36] used CART, J48 and M5P classification methods to predict students' performance in future tests. They used a dataset of 1000 students from two different courses. Their goal was to predict at-risk students at the early stage of the semesters. Reference [37] also used J48 and other classification algorithms to predict students' performance. They use students' GPA as a dependent variable and their grades of team work, attendance and practical exam scores are used as explanatory variables. The students are classified as good, satisfactory or poor students. Reference [38] used seven different classification algorithms to predict students' performance using data of 1000 students which also include missing values and outliers. Reference [39] used decision trees and clustering to predict students' performance in four-year study program. They classified students into low-high achieving groups. Here, we point out two important considerations, first the use of classification techniques particularly decision trees is fairly common in predicting students' future performance and identifying at-risk students. Second, the explanatory variables used in these studies are demographic variables or related to different academic activities, for example, group participation, practical tasks scores, past grades, etc. However, use of homework submission behaviors is not seen in such studies. In this study, we use students' homework submission behaviors to classify them as procrastinators or non-procrastinators. The novel method of building homework submission feature vector is used for this purpose. The students classified as procrastinators are potentially at-risk of lower achievement in the future. The timely detection of such behaviors at the early stage of the course can help students to improve their achievement.

The discussion presented in this section guided us to postulate the research question and to choose the classification techniques for analysis. The use of classification methods is common in categorizing students according to their performance and predicting their future performance. However, two factors are not considered in research studies, using students' tasks or homework submission behaviors and predicting academic procrastinations. In this study, we use students' homework submissions, i.e., homework submitted on time,

homework submitted late and homework not submitted to build a feature vector of students' homework submission behaviors. Building a feature vector using homework submission data is a novel way of representing students' behaviors. This feature vector is used to train a predictor to classify students into two or more categories. We try to reveal the most at-risk students by using different values of k, i.e., different number of clusters. It is expected that at much higher value of k, more refined groups of students with varying behaviors can be revealed. Later, we use different classification methods to find best method for a particular number of classes. The different classification methods are compared for different number of classes.

III. RESEARCH QUESTION

In present study, we aim to find best classification method for a given number of classes. As students are grouped into more clusters or classes, more finer details are revealed. Once, we identify different groups of students with different behaviors, we apply classification methods to see which method performs well. The research question posted in this study is:

- With different number of classes in feature vector, how accurate a classification method is to correctly classify students?

IV. DATA AND METHODS

A. DATA

The data used in this study was collected from SCHOLAT Course logs. The course was conducted in the spring of 2018. The course title is ACM Programming and it was taught in fourth-semester of the four-year study program in computer science. The course was conducted in the blended learning mode in which in addition to normal classroom sessions, the students were asked to perform various activities through on-line learning management system. For example, the students submit their homework on-line, participate in course question and answer forum, access course material on-line, etc. There are a total of 115 students in this course. Six students did not take the final exam. These students were excluded from dataset, remaining 109 students. Three variables were extracted from SCHOLAT Course logs, i.e., homework start date ($Date_{start}$), homework end date ($Date_{end}$) and homework upload date ($Date_{upload}$). Each homework is represented by a triple of three Boolean variables as shown in equation 1.

$$w_i = v_1, v_2, v_3 \tag{1}$$

The algorithm 1 shows detailed process of computing values of v_1, v_2 and v_3 and feature vector X. The algorithm inputs three values from course database logs, i.e., $Date_{start}, Date_{end}, Date_{upload}$. For each student, values of v_1, v_2 and v_3 are computed for each homework. Here, j is total number of students and n is the total number of homework. The algorithm returns a feature vector X as shown in equation 2.

$$X_j = w_{1j}, w_{2j}, .., w_{nj} \tag{2}$$

Algorithm 1 Algorithm to Build Feature Vector X

Require: $Date_{start}, Date_{end}, Date_{upload}$
  while $i \leq j$ **do**
    while $k \leq n$ **do**
      if $Date_{upload} \leq Date_{end}$ **then**
        $w_{ki}[v_1] \leftarrow 1$
      else
        $w_{ki}[v_1] \leftarrow 0$
      end if
      if $Date_{upload} > Date_{end}$ **then**
        $w_{ki}[v_2] \leftarrow 1$
      else
        $w_{ki}[v_2] \leftarrow 0$
      end if
      if $Date_{upload} = \phi$ **then**
        $w_{ki}[v_3] \leftarrow 1$
      else
        $w_{ki}[v_3] \leftarrow 0$
      end if
    end while
    return Feature Vector X_i
  end while

$v_1 = 1$ when a student submits homework on-time, v_2 and v_3 are 0 in that case. $v_2 = 1$ when a student submits homework late, i.e., after due date, v_1 and v_3 are 0. Similarly, $v_3 = 1$ if a student does not submit homework at all. In such case, v_1 and v_2 are 0. Only one of these values can be 1, the other two values will be 0.

In next step, feature vector X is verified for its correctness. Two rules as shown in equation 3 and 4 are used for verification. The algorithm 2 shows detailed process of verifying feature vector X. Two flags f_1 and f_2 are created for two rules mentioned in equations 3 and 4 respectively. First three variable are computed, i.e., $total_{ontime}, total_{late}$ and $total_{NH}$. The total of these three variables should be n, i.e., the total number of homework. If the condition is satisfied, f_1 is set to *true*, otherwise *false*. For each of homework, the sum of v_1, v_2 and v_3 should be 1. So, if this condition is met, f_2 is set to *true*, otherwise *false*. Finally, if both flags f_1 and f_2 are *true*, then the feature vector X_i is verified, otherwise not.

$$total_{ontime} + total_{late} + total_{NH} = n \tag{3}$$

$$v_1 + v_2 + v_3 = 1 \tag{4}$$

where $w_i[v_1]$ is the v_1 variable of i_{th} homework, $w_i[v_2]$ is the v_2 variable of i_{th} homework, and $w_i[v_3]$ is the v_3 variable of i_{th} homework. Where n is total number of homework. In this study, $n = 10$ and $j = 109$. So, there were total 10 homework and 109 instances in the dataset. This makes total 30 attributes for homework. A row in the dataset is shown in Equation 5:

$$w_{1j}[v_1], w_{1j}[v_2], w_{1j}[v_3], w_{2j}[v_1], w_{2j}[v_2], w_{2j}[v_3], w_{3j}[v_1], w_{3j}[v_2], w_{3j}[v_3], ., w_{10j}[v_1], w_{10j}[v_2], w_{10j}[v_3] \tag{5}$$

So there are 30 attributes which are used to represent each homework for each student in the course during the

Algorithm 2 Algorithm to Verify Feature Vector X

Require: Feature Vector X_i, j, n
  set flags f_1 and f_2 to *false*
  $f_1 \leftarrow false, f_2 \leftarrow false$
  while $p \leq j$ **do**
    $total_{ontime} \leftarrow \sum_{i=1}^{n} w_i[v_1]$
    $total_{late} \leftarrow \sum_{i=1}^{n} w_i[v_2]$
    $total_{NH} \leftarrow \sum_{i=1}^{n} w_i[v_3]$
    if $total_{ontime} + total_{late} + total_{NH} = n$ **then**
      $f_1 \leftarrow true$
    else
      $f_1 \leftarrow false$
    end if
    while $q \leq n$ **do**
      if $v_1 + v_2 + v_3 = 1$ **then**
        $f_2 \leftarrow true$
      else
        $f_2 \leftarrow false$
      end if
    end while
    if f_1 **and** f_2 are *true* **then**
      Feature Vector X_i is verified
    else
      Feature Vector X_i is not verified
    end if
  end while

complete semester. The variables $total_{ontime}$, $total_{late}$, and $total_{NH}$ are used to form the clusters of students using k-means algorithm. After the clusters have been formed, another class attribute is added in the feature vector. The class attribute is used to specify the behavioral category of the student as formulated by clustering. So, the final dataset looks as shown below:

$$w_{1j}[v_1], w_{1j}[v_2], w_{1j}[v_3], w_{2j}[v_1], w_{2j}[v_2], w_{2j}[v_3], w_{3j}[v_1], w_{3j}[v_2], w_{3j}[v_3], \ldots\ldots, w_{10j}[v_1], w_{10j}[v_2], w_{10j}[v_3], \; class \tag{6}$$

This makes a total of 31 attributes in the dataset and all of these attributes are used to train the classifiers.

B. METHODS

After building the feature vector, the three variables $total_{ontime}$, $total_{late}$, and $total_{NH}$ are used to form clusters of students using the k-means algorithm. Here, clusters are formed at four different values of k, i.e., $k = 2, 3, 4$, and 5. The main purpose of clustering is to identify procrastinating, i.e., those students who submit their homework late or do not submit at all, and non-procrastinators, i.e., those students who submit their homework on-time. As, the value of k is increased, more refined clusters are revealed marking a clear boundary between procrastinators and non-procrastinators.

TABLE 1. Descriptive statistics.

	$total_{ontime}$	$total_{late}$	$total_{NH}$	Score
$total_{ontime}$	1.0			
$total_{late}$	-0.44	1.0		
$total_{NH}$	-0.92	0.049	1.0	
Score	0.80	-0.16	-0.82	1.0
Mean	7.64	0.63	1.72	83.83
SD	2.30	0.91	2.06	16.30
Maximum	10	3	10	98
Minimum	0	0	0	0

After clustering has been performed, a class variable is added to the dataset and classification algorithms are applied on the dataset. In present analysis, ten classification methods, i.e., ZeroR, OneR, ID3, J48, random forest, decision stump, JRip, PART, NBTree and Prism are used. Four different performance measures, i.e., percentage of correctly and incorrectly classified instances, kappa statistics and RMSE (Root Mean Square Error) are used for method evaluations. In all methods, 10-fold cross validation was used.

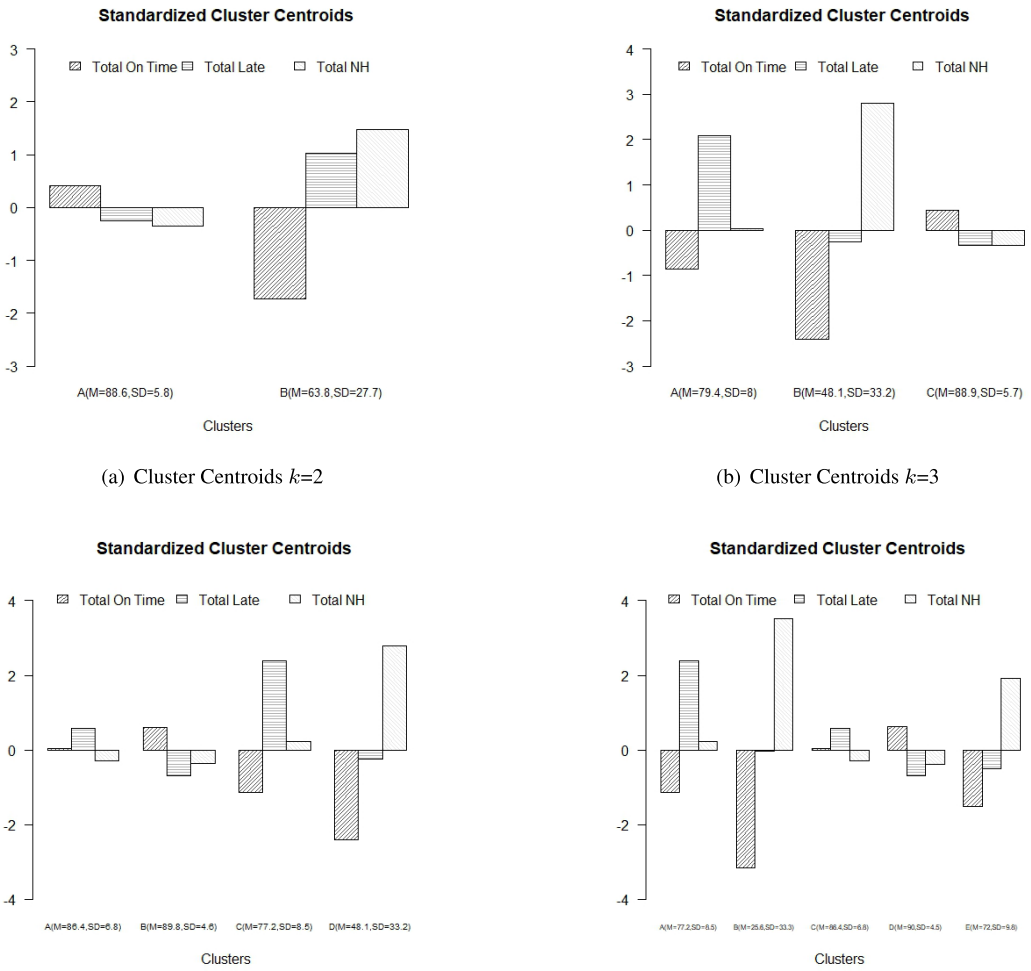
V. RESULTS

A. DESCRIPTIVE STATISTICS

First, we present descriptive statistics of data used for analysis. We only present descriptive statistics for total number of homework submitted on-time, total homework submitted late, total homework not submitted and final scores. The students who did not take the final examination were excluded. Table 1 shows the results of descriptive statistics. We present 5 descriptive statistics, i.e., correlation, mean, standard deviation, maximum and minimum values. Table 1 shows that there is a positive correlation of total number of homework submitted on-time with final score. However, there is a negative correlation of number of homework submitted late and number of homework not submitted at all with the final score.

B. CLUSTER FORMATION

Clustering is essentially is the process of splitting data into partitions where highly similar objects are grouped together. k-means clustering algorithm, however, does not enforce any number of clusters. Figure 1 shows the standardized cluster centroids with different values of k, i.e., $k = 2, 3, 4$ and 5. Figure 1(a) shows standardized cluster centroids at $k = 2$. There are two distinct clusters, cluster A with positive score on on-time homework submission and cluster B with negative homework submission. The cluster A can be called as group of non-procrastinator with high average final scores, i.e., mean = 88.6 and SD = 5.8. And the cluster B can be called as group of non-procrastinators with a lower final score average than cluster A, i.e., mean = 63.8 and SD = 27.7. Figure 1(b) shows standardized cluster centroids at $k = 3$. There are three clusters, cluster A has high average final scores, i.e., mean = 79.4 and SD = 8. But, this group has positive late submission scores. The group of procrastinators,



(a) Cluster Centroids k=2

(b) Cluster Centroids k=3

(c) Cluster Centroids k=4

(d) Cluster Centroids k=5

FIGURE 1. Standardized Cluster Centroids at different values of *k*.

i.e., cluster B is revealed with little more detail. The mean average final scores of this group is 48.1 and standard deviation is 33.2. Finally, the cluster C has positive on-time submission scores and highest average final scores, i.e., mean = 88.9 and SD = 5.7. Figure 1(c) shows four clusters when the value of k is 4. Here, it can be seen that three groups, i.e., clusters A, B and C are with high average score and cluster D is with very low average final score. The average final scores of cluster D is 48.1 and standard deviation is 33.2. The other groups, i.e., clusters A, B and C although have similar achievements but their homework submission behaviors are different. For example, cluster B has highest average final score, i.e., mean = 89.8 and SD = 8.5, and positive on-time submission scores. The clusters A and C have high average final scores, i.e., mean scores 86.4 and 77.2 respectively, and standard deviation is 6.8 and 8.5 respectively. However, cluster A has very little positive on-time submission scores and cluster C has negative on-time submission scores. Figure 1(d) shows five clusters. Here, also group of procrastinators, i.e., cluster B is distinct with high negative on-time submission scores and very low average final scores, i.e., mean = 25.6 and SD = 33.3. Similarly, cluster D is group of highest achievers, i.e., mean = 90 and SD = 4.5. This group has highest positive on-time submission scores. The other three clusters A, C and E have similar achievements but they have different submission behaviors. For example, cluster C has very low positive on-time submission scores but also have positive late submission scores. This group has closet average final scores to most successful group, i.e., mean = 86.4 and SD = 6.8. The other two clusters A and E have similar average final scores, i.e., mean scores are 77.2 and 72 respectively, and standard deviation is 8.5 and 9.8 respectively. Both groups have high negative on-time submission scores.

Figure 2 shows optimal number of clusters as proposed by Elbow method. The Elbow method compares total within cluster sum of squared error with different number of clusters. As number of clusters are increased, the total within cluster sum of squared errors decreases sharply and then become almost constant. An elbow is created just before the measure

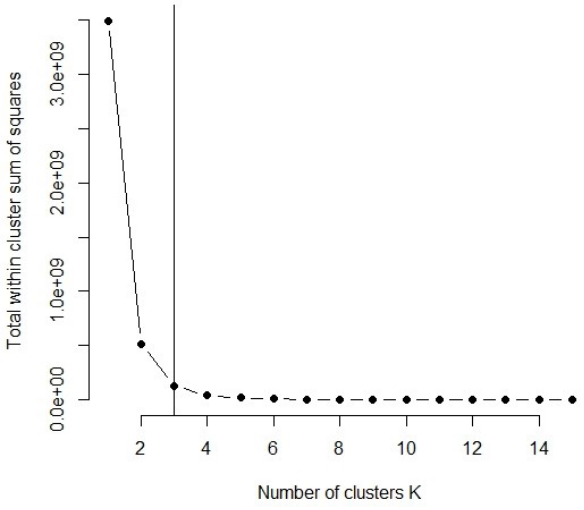


FIGURE 2. Optimal number of clusters as proposed by Elbow method.

become steady and there is no significant change in within cluster errors even number of clusters are increased further. The point where elbow is created is taken as the optimal number of clusters. Here, it can seen in Figure 2 that $k = 3$ is the optimal number of clusters.

C. CLASSIFICATION RESULTS

Ten classification methods, namely ZeroR, OneR, ID3, J48, random forest, decision stump, JRip, PART, NBTree and Prism were used to classify data. Classification results were compared with different number of classes. Four evaluation measures, i.e., percentage of correctly classified instances, percentage of incorrectly classified instances, kappa statistic and root mean squared error were used to evaluate the performance of classification methods with different number of classes. Figure 3 shows percentage of correctly classified instances. It is evident that percentage of correctly classified instances decreases as the number of classes increases for the same method. However, ZeroR and OneR showed worst performance, i.e., decreasing from 80.7 and 90.8 to 51.4 and 65.1 respectively when number of classes are increased from 2 to 5. A similar picture is seen in Figure 4, where ZeroR and OneR are again the worst methods in performance comparisons, i.e., 19.3 and 9.2 to 48.6 and 34.9 respectively when number of classes are increased from 2 to 5. Figure 5 and 6 shows kappa statistics and RMSE values for different methods when number of classes are increased from 2 to 5 respectively. Table 2 shows standard deviation of different methods for four different measures. The table shows how these four measures, i.e., percentage of correctly and incorrectly classified instances, kappa statistics and root mean squared error change as number of classes are changed from 2 to 5. For correctly classified instances, ZeroR and OneR showed higher standard deviation values w.r.t other methods, i.e., the number of correctly classified instances vary greatly for these two methods. A similar trend

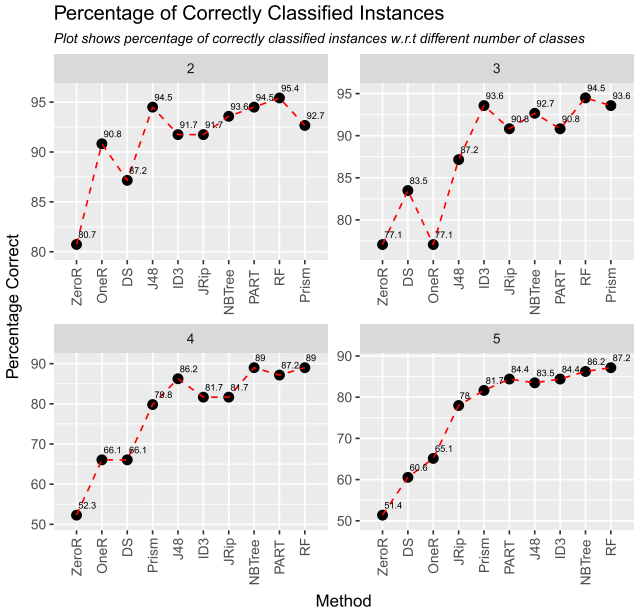


FIGURE 3. Number of correctly classified instances w.r.t different number of classes.

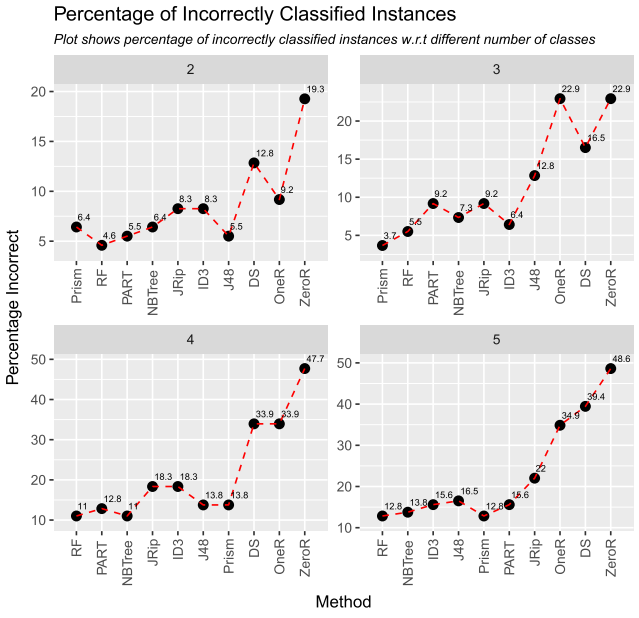


FIGURE 4. Number of incorrectly classified instances w.r.t different number of classes.

is seen in case of incorrectly classified instances. In case of kappa statistics, NBTree shows most consistent performance with lowest standard deviation, i.e., 0.017. Random forest being the second best classifier as the standard deviation for this method is second lowest, i.e., 0.023. Similarly, NBTree also showed a consistent performance in RMSE values with standard deviation of 0.01. The other methods with lowest standard deviation are Random forest and J48 with standard deviation values of 0.013 and 0.017 respectively.

Table 3 shows results of comparison between classifiers. Here, NBTree is compared with other classifiers using

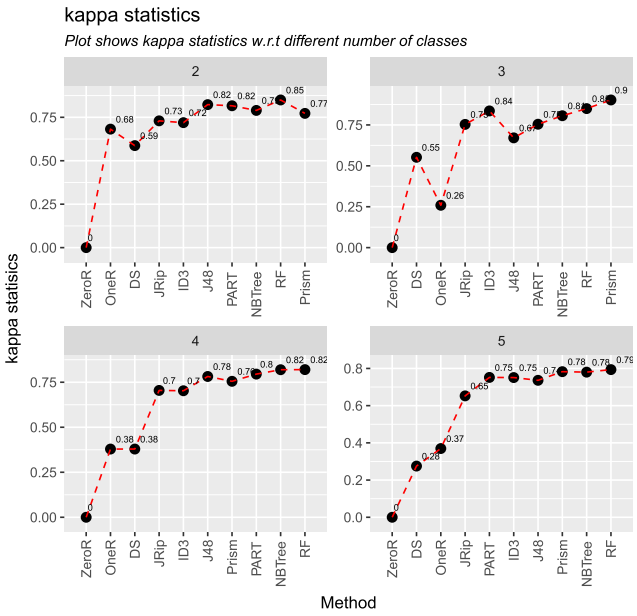


FIGURE 5. Kappa statistics w.r.t different number of classes.

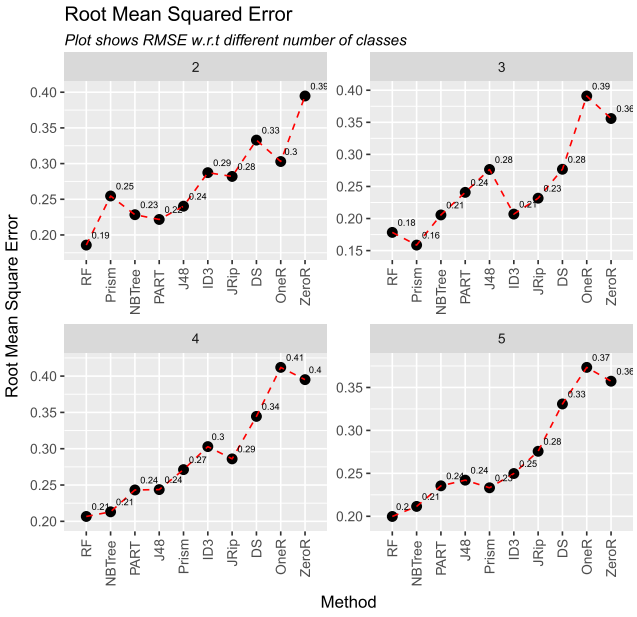


FIGURE 6. Root Mean Squared Error w.r.t different number of classes.

paired t-test. All tests were performed at 5% level of significance. The top row in Table 3 presents the four datasets used in the analysis. The columns present the classifiers. In this comparison, the results show which method performed significantly better or worse than NBTree. As it can be seen in Table 3 that no method performed significantly better than NBTree in any of the datasets. However, ZeroR and OneR performed worse than NbTree in 3*Class* dataset. Also, JRip, Decision Stump, ID3, ZeroR and OneR performed worse in 4*Class* dataset. Further, PART, J48, JRip, Decision Stump, OneR and ZeroR also performed worse than NBTree in

TABLE 2. Standard deviation w.r.t different classes.

	Correct	Incorrect	kappa	RMSE
ZeroR	15.7	15.7	0	0
OneR	12.0	12.0	0.18	0.047
ID3	5.7	5.7	0.06	0.042
J48	4.7	4.7	0.06	0.017
Random Forest	4.0	4.0	0.023	0.013
JRip	6.8	6.8	0.043	0.025
PART	4.4	4.4	0.032	0.096
Prism	7.2	5.0	0.067	0.049
Decision Stump	13	13	0.15	0.030
NBTree	3.4	3.4	0.017	0.01

TABLE 3. Classifiers comparisons.

	$2Class$	$3Class$	$4Class$	$5Class$
NBTree	94.06	92.48	87.77	87.15
Random Forest	95.52	94.85	91.46	89.46
Prism	92.19	92.75	84.87	84.61
PART	93.69	93.46	87.44	85.87*
J48	93.70	90.35	87.07	83.04*
JRip	93.15	89.06	81.02*	83.13*
Decision Stump	88.73	82.39	65.98*	62.46*
ID3	92.19	91.70	84.24*	85.99
OneR	91.95	77.45*	65.98*	65.11*
ZeroR	80.73	77.09*	52.27*	51.36*

*Performed worse statistically significant at 0.05.

5*Class* datasets. All other tests were non-significant, i.e., no conclusion can be made about better or worse performance. The evaluation matric used in this comparison was percentage of correctly classified instances.

VI. DISCUSSION

In this study, a novel method of building a feature vector using students' homework submission data is used. This feature vector represents students' homework submission behaviors during the whole semester. Three key indicators are used as building blocks of this feature vector, i.e., each homework is represented by a triple of three Boolean variables. Using homework deadline and students' homework submission dates, first, the values of three variables are calculated. The process is discussed in detail in section IV-A. This feature vector comprehensively represents students' homework submission behaviors during the semesters, i.e., how many homework a student submitted on-time, how many homework submitted late and how many homework not submitted. The feature vector is then used to group students with similar behaviors in clusters. k-means clustering algorithm is used for this purpose. The section V-B explains the process of clustering in detail. The main task of clustering is to identify procrastinating and non-procrastinating students. The Figure 1 shows standardized cluster centroids at $k = 2$, 3, 4 and 5. At $k = 2$, there are two distinct clusters.

The cluster A is regarded as group of non-procrastinators since their on-time submission scores are positive and late or non-submission are negative. The cluster B is a group of procrastinators since they have negative on-time submission scores and positive late and non-submission scores. These behaviors are further refined when three clusters are formed, i.e., $k = 3$. Here, cluster C is group of non-procrastinators with positive on-time submission scores and negative late and non-submission scores. However, two procrastinating groups are revealed, i.e., cluster A with negative on-time submission scores and positive late submission scores, and cluster B with high negative on-time submission scores and high positive non-submission scores. The main difference between these two groups is their achievements, i.e., cluster A has higher achievement score than cluster B. To further reveal the behaviors, now four clusters are formed, i.e., $k = 4$. Here, two procrastinating and two non-procrastinating groups are emerged. The clusters A and B is regarded as groups of non-procrastinating students, although cluster A has positive late submission scores but they also have high achievement scores. The clusters C and D are the groups of procrastinators with negative on-time submission scores and positive late or non-submission scores. The only distinguishable difference between these two groups is the higher achievement scores of cluster C. However, if the data is further divided into more clusters, i.e., $k = 5$, no significant clusters are revealed except that a procrastinating group with lowest achievement is identified, i.e., cluster B with average final score of 25.6. From above discussion, it is clear that beyond $k = 3$, no more significant clusters are achieved. Figure 2 also suggest that $k = 3$ is the optimal number of clusters for the data used in this study.

Another important task of this study is to find the most suitable classification method. Once clustering has been performed, a class label is added to the dataset. An feature vector as shown in equation 6 is used for classification purposes. Figures 3, 4, 5 and 6 shows different measure to evaluate the quality of classification results. Figure 3 shows that for the same method the number of correctly classified instances decreases as the number of classes increases. Similarly, Figure 4 also shows that number of incorrectly classified instances increase as the number of classes increases. Generally, the random forest algorithm performs best and ZeroR performs worst in these evaluation comparisons. However, kappa statistics gives a reliable estimate of most suitable method. Previously, it was mentioned that beyond $k = 3$, no significant clusters are revealed. Also, elbow method suggests $k = 3$ as the optimal number of clusters. Therefore, a comparison between different methods at $k = 3$ reveals that Prism performs best in terms of kappa statistics and RMSE values as compared to other methods. However, if number of classes are chosen other than three, there are other methods which perform well as compared to Prism. The more number of classes offers more personalization. Therefore, at some point the teacher might want to add further classes into the feature vector and make responses to the students more personalized.

Algorithm 3 Algorithm for Students' Academic Performance Enhancement Through Homework Late/ Non-Submission Detection (SAPE)

Require: $Date_{start}$, $Date_{end}$, $Date_{upload}$
Construct feature vector (without class labels)
$X_j = w_{1j}, w_{2j}, .., w_{nj}$
Apply clustering algorithm at $k = 2, 3, 4, 5$ to feature vector X
Compare clusters
Visual inspection for distinct clusters or apply Elbow Method to find optimal number of clusters
Add class labels to feature vector
$X_k \leftarrow X + class_k$
Apply classification algorithm
ZeroR, OneR, ID3, J48, Random Forest, etc.
Compare performance metrics
$M_c = m_1, m_2, m_3, ..., m_n$
while $i \leq n$ **do**
if $M_{c_i} > M_{c_{i+1}}$ **then**
$C \leftarrow c_i$
else
$C \leftarrow c_{i+1}$
end if
end while
Use classification algorithm C to identify on-time, late and non-submissions
Extend interventions

Algorithm 3 presents a generalized algorithm to build feature vector and apply clustering and classification to it. The algorithm starts by building a feature vector from $Date_{start}$, $Date_{end}$ and $Date_{upload}$. The algorithm 1 explains the process of building feature vector in detail. Once, the feature vector has been built, now it is time to add class labels. The feature vector built using algorithm 1 does not has class labels. The k-means clustering algorithm is applied to group students with similar behaviors. As discussed previously, for more personalization, more groups or clusters can be formed. Typically, visual method can be used to inspect distinct clusters. Otherwise, objective cluster evaluation measures such as Elbow Method can be used to get optimal value of k. Once, distinct clusters are identified, the class labels are added to the feature vector. However, for more classes, a higher value of k may be chosen. The k in X_k represents the number of classes in feature vector. At this stage, the feature vector is finally prepared to apply classification algorithms. As shown in section V-C, a number of different classification algorithms can be applied. However, the performance of classification algorithms vary as the number of classes are changed. So, different classification algorithms can be compared according to different number of classes to get the best one. Here, M_c represents a vector comprising of performance metrics of classification methods. For example, in present analysis, four performance measures, i.e., percentage of correctly classified instances, percentage of incorrectly classified instances,

kappa statistic and RMSE are used. The performance metrics for different methods are compared for particular number of classes and best method C is selected. The classification essentially identifies different students' behaviors, i.e., students who submit homework on-time, those who submit late or do not submit at all. Once, these students have been identified, the learning process can be intervened to help students facing learning difficulties.

The algorithm described above fully automates the process of identifying students having learning difficulties. The algorithm is generalized and has been made flexible to accommodate different types of behaviors. This could be helpful to offer personalized learning supports to the students. A timely detection of students' future procrastination tendencies has positive effects on students learning. This can help course instructors to implement a number of pedagogical practices to improve students' learning. It can increase and can make supervision of students much easier, i.e., [16] reported that lack of supervision from course instructors is a major reason for increased procrastination tendencies of students. The students who are flagged red by the classification model can be kept in special observation list and can be reminded repeatedly to submit homework on-time. The course instructors can provide appropriate feedback to both classes of students, i.e., the students who are regularly submitting their homework on-time are encouraged and those who fail to do so are motivated to submit their homework on-time like other students. References [40] and [41] proposed to provide regular feedback to students to enhance their performance and reduce procrastination. Reference [14] observed that students who are encouraged and motivated by the course instructors show reduced tendencies of procrastination. Reference [42] noticed that procrastination tendencies in students can be reduced if they are informed about the performance of other students. The use of social media is quite common nowadays. The students can be motivated and informed about their performance in social media groups. They can be reminded about the homework deadlines and if the deadline is passed, they can be asked to submit homework as early as possible. The importance of social media in motivation and to provide information about their learning is emphasized by [43]. Timely interventions are very important in students' learning. Instead of giving feedback at the end of the semester, if the future behaviors of students are predicted in the start of semester, a lot of difference can be made, i.e., students achievement can be increased, their behavioral discrepancies can be modified. In this study, the classification model is able to predict students' procrastination at the start of the semester, enabling course instructors to take remedial actions before it is too late.

VII. CONCLUSION AND FUTURE WORK

Three algorithms are presented in this work. The first algorithm, i.e., algorithm 1 is a novel way of building students' homework submission feature vectors. This feature vector can be used to represent a student's homework submission behavior in a semester. The algorithm 2 details the steps taken to verify the correctness of feature vector. The third algorithm, i.e., algorithm 3 presents the process of applying clustering and classification methods to predict students' procrastinating and non-procrastinating behaviors. The algorithm is generalized and flexible to fully automate the process of identify students with learning difficulties. In future, we intend to extend the present work by adding more courses with different number of homework. This would be helpful to build comprehensive algorithm to detect students with learning difficulties.

In present study, we, however, do not consider features vector of different lengths, i.e., multiple courses with different number of homework. We intend to extend present analysis by building features vectors from different courses and different number of homework.

REFERENCES

[1] J. Mostow and J. Beck, "Some useful tactics to modify, map and mine data from intelligent tutors," *Natural Lang. Eng.*, vol. 12, no. 2, pp. 195–208, 2006.
[2] R. Azevedo, J. G. Cromley, F. I. Winters, D. C. Moos, and J. A. Greene, "Adaptive human scaffolding facilitates adolescents' self-regulated learning with hypermedia," *Instructional Sci.*, vol. 33, nos. 5–6, pp. 381–412, 2005.
[3] R. Azevedo and R. Feyzi-Behnagh, "Dysregulated learning with advanced learning technologies," in *Proc. AAAI Fall Symp. Ser. Cogn. Metacognitive Educ. Syst.*, 2010, pp. 5–10.
[4] I. E. Allen, *Changing Course: Ten Years of Tracking Online Education in the United States*. Needham, MA, USA: Sloan Consortium, 2013.
[5] J. W. You, "The relationship among academic procrastination, self-regulated learning, fear, academic self-efficacy, and perceived academic control in e-learning," *J. Educ. Inf. Media*, vol. 18, no. 3, pp. 249–271, 2012.
[6] R. Cerezo, M. Sánchez-Santillán, M, P. Paule-Ruiz, and J. C. Núñez, "Students' LMS interaction patterns and their relationship with achievement: A case study in higher education," *Comput. Educ.*, vol. 96, pp. 42–54. May 2016.
[7] M. K. Akinsola, A. Tella, and A. Tella, "Correlates of academic procrastination and mathematics achievement of university undergraduate students," *Eurasia J. Math. Sci. Technol. Educ.*, vol. 3, no. 4, pp. 363–370, 2007.
[8] C. Romero and S. Ventura, "Educational data science in massive open online courses," *Wiley Interdiscipl. Rev. Data Mining Knowl. Discovery*, vol. 7, no. 1, p. e1187, 2017.
[9] C. H. Lay, "At last, my research article on procrastination," *J. Res. Personality*, vol. 20, no. 4, pp. 474–495, 1986.
[10] L. J. Solomon and E. D. Rothblum, "Academic procrastination: Frequency and cognitive-behavioral correlates," *J. Counseling Psychol.*, vol. 31, no. 4, pp. 503–509, 1984.
[11] C. Senécal, R. Koestner, and R. J. Vallerand, "Self-regulation and academic procrastination," *J. Social Psychol.*, vol. 135, no. 5, pp. 607–619, 1995.
[12] G. Schraw, T. Wadkins, and L. Olafson, "Doing the things we do: A grounded theory of academic procrastination," *J. Educ. Psychol.*, vol. 99, no. 1, pp. 12–25, 2007.
[13] S. Piers, "The nature of procrastination: A meta-analytic and theoretical review of quintessential self-regulatory failure," *Psychol. Bull.*, vol. 133, no. 1, pp. 65–94. 2007.
[14] N. Michinov, S. Brunot, B. O. Le, J. Juhel, and M. Delaval, "Procrastination, participation, and performance in online learning environments," *Comput. Educ.*, vol. 56, no. 1, pp. 243–252, 2011.
[15] A. J. Howell and D. C. Watson, "Procrastination: Associations with achievement goal orientation and learning strategies," *Personal. Individual Differences*, vol. 43, no. 1, pp. 167–178, 2007.
[16] B. W. Tuckman, "Relations of academic procrastination, rationalizations, and performance in a Web course with deadlines," Dennis Learn. Center, Ohio State Univ., Columbus, OH, USA, Psychol. Rep. 96 (3_suppl), 2005, pp. 1015–1021.

[17] C. Grunschel, M. Schwinger, R. Steinmayr, and S. Fries, "Effects of using motivational regulation strategies on students' academic procrastination, academic performance, and well-being," *Learn. Individual Differences*, vol. 49, pp. 162–170, Jul. 2016.
[18] K. Wäschle, A. Allgaier, A. Lachner, S. Fink, and M. Nückles, "Procrastination and self-efficacy: Tracing vicious and virtuous circles in self-regulated learning," *Learn. Instruct.*, vol. 29, pp. 103–114, Feb. 2014.
[19] M. M. L. Rebetez, L. Rochat, C. Barsics, and M. Van der Linden, "Procrastination as a self-regulation failure: The role of inhibition, negative affect, and gender," *Personality Individual Differences*, vol. 101, pp. 435–439, Oct. 2016.
[20] J. Patrzek, S. Sattler, F. van Veen, C. Grunschel, and S. Fries, "Investigating the effect of academic procrastination on the frequency and variety of academic misconduct: A panel study," *Stud. Higher Educ.*, vol. 40, no. 6, pp. 1014–1029, 2015.
[21] K. Hakan, "Correlation among academic procrastination, personality traits, and academic achievement," *Anthropologist*, vol. 20, no. 1, p. 2, 2015.
[22] C. Romero, S. Ventura, M. Pechenizkiy, and R. S. J. D. Baker, Eds., *Handbook of Educational Data Mining*. Boca Raton, FL, USA: CRC Press, 2010.
[23] I. H. Witten, E. Frank, and M. A. Hall, *Data Mining: Practical Machine Learning Tools and Techniques*. San Mateo, CA, USA: Morgan Kaufmann, 2016.
[24] J. R. Quinlan, "Induction of decision trees," *Mach. Learn.*, vol. 1, no. 1, pp. 81–106, 1986.
[25] R. Quinlan, *C4.5 Programs for Machine Learning*. Amsterdam, The Netherlands: Elsevier, 2014.
[26] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The WEKA data mining software: An update," *ACM SIGKDD Explor. Newslett.*, vol. 11, pp. 10–18, Jun. 2009.
[27] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.
[28] W. W. Cohen, "Fast effective rule induction," in *Machine Learning Proceedings*. San Mateo, CA, USA: Morgan Kaufmann, 1995, pp. 115–123.
[29] E. Frank and I. H. Witten, "Generating accurate rule sets without global optimization," in *Proc. 15th Int. Conf. Mach. Learn.*, 1998, pp. 144–151.
[30] J. Cendrowska, "PRISM: An algorithm for inducing modular rules," *Int. J. Man-Mach. Stud.*, vol. 27, no. 4, pp. 349–370, 1987.
[31] W. Iba and P. Langley, "Induction of one-level decision trees," in *Proc. Mach. Learn.* San Mateo, CA, USA: Morgan Kaufmann, 1992, pp. 233–240.
[32] R. Kohavi, "Scaling up the accuracy of Naive-Bayes classifiers: A decision-tree hybrid," in *Proc. 2nd Int. Conf. Knowl. Discovery Data Mining*, vol. 96, 1996, pp. 202–207.
[33] M. Agaoglu, "Predicting instructor performance using data mining techniques in higher education," *IEEE Access*, vol. 4, pp. 2379–2387, 2016.
[34] A. A. Saa, "Educational data mining students' performance prediction," *Int. J. Adv. Comput. Sci. Appl.*, vol. 7, no. 5, pp. 212–220, 2016.
[35] P. Kaur, M. Singh, and G. S. Josan, "Classification and prediction based data mining algorithms to predict slow learners in education sector," *Proc. Comput. Sci.*, vol. 57, pp. 500–508, Mar. 2015.
[36] K. Kaur and K. Kaur, "Analyzing the effect of difficulty level of a course on students performance prediction using data mining," in *Proc. 1st Int. Conf. Next Gener. Comput. Technol. (NGCT)*, Sep. 2015, pp. 756–761.
[37] F. Ahmad, N. H. Ismail, and A. A. Aziz, "The prediction of students' academic performance using classification data mining techniques," *Appl. Math. Sci.*, vol. 9, pp. 6415–6426, Oct. 2015.
[38] M. Pandey and S. Taruna, "Towards the integration of multiple classifier pertaining to the Student's performance prediction," *Perspect. Sci.*, vol. 8, pp. 364–366, Sep. 2016.
[39] R. Asif, A. Merceron, S. A. Ali, and N. G. Haider, "Analyzing undergraduate students' performance using educational data mining," *Comput. Educ.*, vol. 113, pp. 177–194, Oct. 2017.
[40] W. Doherty, "An analysis of multiple factors affecting retention in Web-based community college courses," *Internet Higher Edu.*, vol. 9, no. 4, pp. 245–255, 2006.
[41] B. W. Tuckman, "Academic procrastinators: Their rationalizations and Web-course performance," in *Proc. APA Symp.*, 2002, pp. 22–25.
[42] N. Michinov and C. Primois, "Improving productivity and creativity in online groups through social comparison process: New evidence for asynchronous electronic brainstorming," *Comput. Hum. Behav.*, vol. 21, no. 1, pp. 11–28, Jan. 2005.
[43] J. C. Dunlap and P. R. Lowenthal, "Tweeting the night away: Using Twitter to enhance social presence," *J. Inf. Syst. Educ.*, vol. 20, no. 2, p. 129. 2009.

AFTAB AKRAM received the Ph.D. degree in computer science from South China Normal University, Guangzhou, China. He has over 15 years experience in teaching in conventional and non-conventional formats. He specializes in designing algorithms for learning enhancements. He has been a Faculty Member with the University of Education, Lahore, Pakistan, since 2005. His research interests include e-learning, blended learning, educational data mining, and machine learning.

CHENGZHOU FU received the B.E., M.E., and Ph.D. degrees from South China Normal University, Guangzhou, China. He was an Engineer with Tencent (a well-known Internet company in China) and was responsible for client security research. He is currently a Lecturer with the College of Medical Information Engineering, Guangdong Pharmaceutical University. He is also a Technical Director in an SNS platform named SCHOLAT. He has been a Visiting Scholar with the Victoria University, Australia. His current research interests include social network, big data, information security, and data mining. He received the best paper award at the 6th Chinese Conference on Cloud Computing (CCCC 2015). He is also the Co-Chair of several international conferences, such as HCC 2017, HCC 2018, and UMLL 2019.

YUYAO LI is currently pursuing the Ph.D. degree with South China Normal University (SCNU) under the Supervision of Professor Y. Tang. He is also a Faculty Member with the School of Information Science and Technology/School of Cyber Security, Guangdong University of Foreign Studies. His research interests include social network, data mining, and NLP.

MUHAMMAD YAQOOB JAVED received the B.Sc. degree from the University of Central Punjab (UCP), Lahore, Pakistan, the M.Sc. degree from the University of Engineering and Technology (UET), Lahore, and the Ph.D. degree from the University of Science and Technology of China (USTC), Anhui, China. He served as an Assistant Professor for the University of Central Punjab, from 2008 to 2018. He is currently an Assistant Professor with the Department of Electrical and Computer Engineering, COMSATS University Islamabad (CUI), Lahore Campus, Lahore. He is also a Supervisor of a research group "Efficient Electrical Energy Systems" with CUI Lahore. He has authored/coauthored several research articles in leading journals and conferences of his field. His research interests include the design of renewable energy systems, standalone solar photovoltaic (PV) systems, maximum power point trackers for PV systems, partial shading effects in PV systems, economic energy dispatch, micro grid, and smart grid systems.

RONGHUA LIN is currently pursuing the master's degree with the School of Computer Science, South China Normal University, Guangzhou, China, under the Supervision of Professor Y. Tang. His current research interests include social network systems and machine learning.

YUNCHENG JIANG received the Ph.D. degree from the Institute of Computing Technology, Chinese Academy of Sciences. He is currently a Professor with the School of Computer Science, South China Normal University, Guangzhou, China. His current research interests include semantic search, semantic computing, and data science. His work is published in journals and conferences, such as *Information Processing & Management*, *Information Sciences*, *Fuzzy Sets and Systems*, *Knowledge-Based Systems*, *Engineering Applications of Artificial Intelligence*, *International Journal of Intelligent Systems*, *International Journal on Semantic Web and Information Systems*, AAAI, and KSEM.

YONG TANG received the B.S. degree in computer science from Wuhan University, in 1985, and the Ph.D. degree in computer science from the University of Science and Technology of China, in 2001. He is currently a Professor and the Dean of the School of Computer Science, South China Normal University, and also serves as the Director of the Services Computing Engineering Research Center of Guangdong Province. His research interests include database and cooperative software, temporal information processing, social network, and big data analytics. He has completed more than 30 research and development projects, and has authored or coauthored more than 100 publications in these areas.