ISSN 2089-8673 (Print) | 2548-4265 (Online)
Volume 14, Issue 1, 2025, pp. 1-12

MULTIMODEL PREDICTION SCORE BASED ON ACADEMIC PROCRASTINATION BEHAVIOR IN E-LEARNING

Bruri Trya Sartana[1], Supeno Mardi Susiki Nugroho[1,2], Umi Laili Yuhana[3], Mauridhi Hery Purnomo[1,2,*]

[1]Department of Electrical Engineering, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia
[2]Department of Computer Engineering, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia
[3]Department of Informatics, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia

Abstract

This research investigates the impact of academic procrastination on student performance in online learning environments and explores a multimodel approach for grade prediction. Academic procrastination is a well-documented issue that negatively affects learning outcomes, often leading to lower academic performance and increased dropout rates in self-paced learning platforms. This study analyzes behavioral data from 377 students, extracted from Moodle activity logs, which record real-time student interactions with learning materials. To address the gap in understanding procrastination patterns through activity logs, key procrastination-related features were derived from timestamps of task access, submission, and engagement duration. Using K-Means clustering with the Elbow method, students were categorized into three procrastination clusters: low procrastination with high academic performance, high procrastination with low performance, and moderate procrastination with average performance. Seven machine learning models were evaluated for predicting student grades, with Random Forest (RF) achieving the highest accuracy (R^2 = 0.812, MAE = 6.248, RMSE = 8.456). These findings highlight the potential of using activity logs to analyze procrastination patterns and predict student performance, allowing educators to develop early intervention strategies that support at-risk students and improve learning outcomes.

Keywords: Academic Performance, Clustering, Online Learning, Prediction, Procrastination

Received: 18-10-2024 | **Accepted:** 20-02-2025 | **Available Online:** 31-02-2025
DOI: https://doi.org/10.23887/janapati.v14i1.85880

I. INTRODUCTION

Assignments are part of the assessment component of the online learning process. Students are often given assignments within a certain period from the start of access to the deadline to submit the results. This timeframe allows students to procrastinate for various reasons [1]. Student academic procrastination is a prevalent issue, marked by the habit of postponing academic responsibilities and delaying task completion [2,3]. Massive Open Online Courses (MOOCs) found that procrastination was the most frequently observed behavior that negatively impacted student's academic achievement and performance [4, 5].

Previous research has proven that procrastination can negatively impact learning outcomes [6–8], and increase dropout rates from online courses [9–11]. Several studies also have found a link between student failure or discouragement in online learning due to lack of time and procrastination by analyzing students' activity records [12–14].

Research found that the failure factor was the tendency to delay completing assignments [15]. According to research in [16–20], machine learning approaches can accurately identify delays in student activity records with accuracy rates ranging from 85% to 95%. An investigation of differences in learning characteristics and self-regulatory behaviors among students with varying degrees of academic procrastination is discussed in [21]. Paper [21] also examines three groups of first-year elementary teacher education program students based on their procrastination levels (low, average, and high). Six themes were identified that described how these students dealt with factors affecting their learning, such as choosing their degree program, starting, and engaging in study activities, responding to failure, self-perception, and study outcomes. Therefore, it is important to understand the factors that

*Corresponding author: hery@ee.its.ac.id (M.H. Purnomo)

contribute to procrastination and develop effective prediction methods.

In paper [22] examined video lecture data from a chemistry course with a high proportion of first-generation students. The four learning patterns used consisted of: Early Planning, Planning, Procrastination, and Low Engagement. The results showed that Early Planning was the most effective method to achieve success in the course, even for first generation students. These findings imply that self-regulation plays a critical role in the STEM achievement of first-generation students, and that targeted support could be beneficial in aiding their success.

Paper [23] proposed a novel technique for predicting students' tendency to procrastinate in online learning by examining their submission behavior patterns. This approach involves categorizing students into different procrastination clusters using an ensemble clustering approach and evaluating multiple classification methods to determine the most reliable predictor. This study was performed on a dataset of 242 students, and the findings revealed that the proposed technique could efficiently identify student procrastination with an accuracy of 97%. This study emphasizes the significance of choosing the best number of features for both clustering and classification methods.

This study aims to identify students at risk for learning delays to enable instructors and administrators to intervene, provide timely support, and improve their academic performance. Furthermore, predicting the level of delay can help instructors understand the underlying causes and design more effective intervention strategies. Understanding the factors that contribute to delays and developing ways to predict them can improve student performance and enhance the effectiveness of online learning.

The contribution of this research is as follows:

- Investigating the prediction of assignment grades as part of learning outcome indicators based on learners' procrastination behavior in completing assignments.
- Recommendations for learning instructors in providing early warning to learning participants.
- Recommendations for instructors to motivate learners to avoid failure in learning.

Unlike previous studies that rely on self-reported procrastination data, this study leverages real-time activity logs from Moodle to classify procrastination behaviors. Furthermore, this research integrates clustering with predictive modeling, demonstrating a novel approach to analyzing procrastination tendencies and forecasting student performance in e-learning environments.

This paper is organized as follows: Section 2 describes the methods used for the prediction process. Section 3 presents result and discussion. Section 4 presents conclusions.

II. PROBLEM DESCRIPTION

In the Moodle system, users can play the roles of instructors and students. Instructors determine the task opening and deadline dates, while students begin working on the task after it is opened, and submit the task before or at the submission deadline. The task score was obtained from the Moodle system, which was graded by the instructor.

Instructors give students the freedom to decide when they will work on and submit the task, but expect them to work on the task as soon as possible and within the specified deadline. Tasks can still work if the student has not pressed the Submit button. The instructor still accepts the tasks completed after the deadline.

From the data, we analyzed students' behavior while completing tasks, particularly academic procrastination. Procrastination can occur when students begin the task (TaskStartPro), delay the time between starting and submitting the task (TaskDelay), or procrastinate submitting the task (TaskSubPro).

To better understand student groups after clustering academic procrastination behaviors in assignment completion, we intend to identify whether other behavioral variables affect students' grades, including the instructor's assignment completion time span and assignment completion time ratio. Grouping students based on similar characteristics and determining the type of procrastination of the resulting groups is essential. In addition, we conducted TaskScore predictions based on previously generated procrastination data.

III. METHODOLOGY

Fig. 1 shows the framework of investigating task score prediction based on learners' procrastination behavior by applying multiple models

A. Data Collection

The dataset consists of Moodle activity logs from 377 students enrolled in 77 online courses at a private university in Indonesia. All students were included without pre-selection based on procrastination behavior, ensuring an unbiased clustering process.

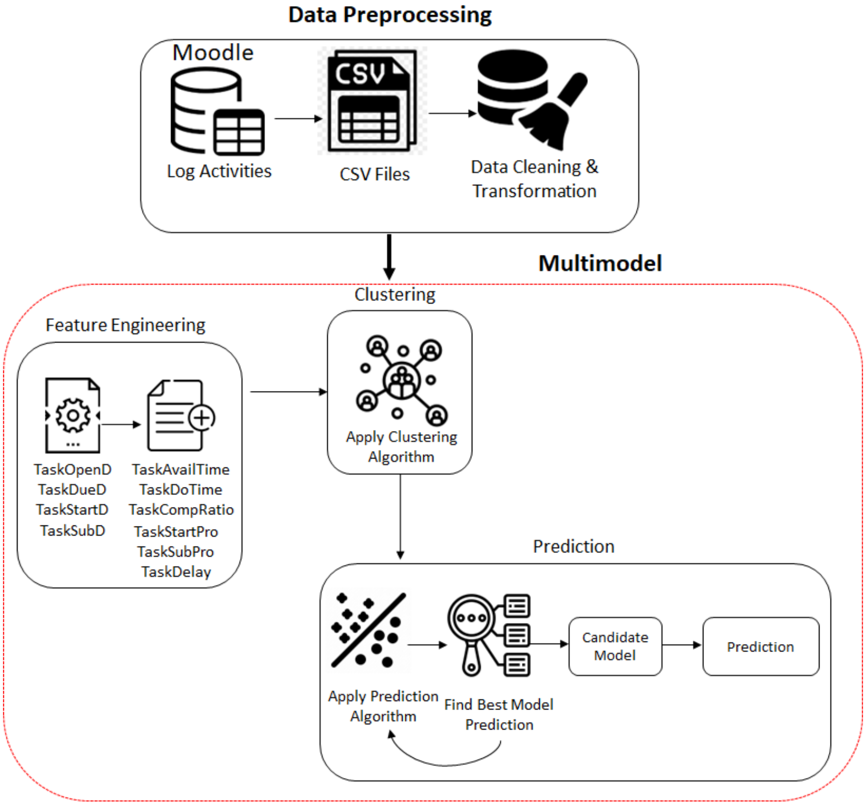

Figure 1. Framework of grade prediction based on task completion procrastination.

Procrastination tendencies were identified retrospectively using extracted features, allowing for a natural distribution of procrastination patterns among students. This approach ensures that the clustering results reflect the full range of student behaviors in an online learning environment. Due to institutional privacy policies and student confidentiality, this data is not publicly available. At the data preprocessing stage, we normalized the data using Z-score scaling to transform the data scale into a normal distribution with a mean of 0 and standard deviation of 1.

To ensure the reliability of extracted procrastination features, we conducted a preliminary statistical analysis of the dataset. The distribution of procrastination features was examined for consistency, and outliers were identified and removed where necessary. In addition, a correlation analysis was performed to verify that procrastination-related features (TaskStartPro, TaskSubPro, TaskDelay) exhibited expected relationships with academic performance (TaskScore). The high correlation between procrastination indicators and student grades supports the validity of these features as predictors of academic performance.

The clustering process was performed with K-means using the elbow method to find the optimal cluster. At the prediction stage, seven prediction models were involved: DT, RF, GB, MP, SVM, K-NN, and LGBM. From all prediction results, performance measurements were performed by applying the indicators of mean accuracy, mean absolute error (MAE), and root mean square error (RMSE).

B. Feature Engineering

Each student, identified by StudentId, was assigned multiple tasks, each with a task opening date (TaskOpenD) and a due date (TaskDueD). The time a student first accessed a task was recorded as TaskStartD, while the submission time was noted as TaskSubD. The relationship between these timestamps was used to derive procrastination-related features, such as the delay between task availability and the first access, as well as the time elapsed between task initiation and submission.

To observe task completion behavior, particularly academic procrastination behavior, we used the mathematical model in Eq.(1)-Eq.(6):

$$TaskAvailTime = TaskDueD - TaskOpenD \quad (1)$$

$$TaskDelay = \frac{TimeSubTask_i - TaskStartD_i}{TaskAvailTime} \quad (2)$$

$$TaskDoTime = TimeSubTask_i - TaskStartD_i \quad (3)$$

$$TaskCompRatio = \frac{TaskDoTime}{TaskAvailTime} \quad (4)$$

$$TaskStartPro = \frac{TaskStartD_i - TaskOpenD}{TaskAvailTime} \quad (5)$$

$$TaskSubPro = \frac{TaskDueD - TasSubD_i}{TaskAvailTime} \quad (6)$$

TaskAvailTime represents the total duration available for students to complete a task, based on the difference between the task opening and due dates. TaskDelay captures the time a student postpones submission after starting a task. TaskDoTime measures the actual time spent completing a task, reflecting the total duration between task initiation and submission.

TaskCompRatio evaluates how efficiently students utilize the available time by comparing the actual time spent to the total allocated time. TaskStartPro quantifies procrastination in starting a task, where a higher value indicates a longer delay before beginning the work. TaskSubPro assesses procrastination in task submission, where lower values represent greater delays in meeting deadlines. If a task is submitted past the deadline, TaskSubPro may be negative or zero. These features were calculated using Microsoft Excel based on predefined formulas applied to extracted Moodle activity log data.

C. Build a Task Procrastination Behavior Vector

Table 1 shows the notation of a task procrastination behavior vector. Pseudocode 1 describes the process to form the delay behavior vector based on the approach described earlier.

The dataset was extracted, focusing on tables that record student interactions with assignments and course modules. Relevant features were derived from event timestamps stored in Moodle's logs, primarily from the logstore_standard_log table. The key columns used include 'timecreated,' which is the timestamp of an event; 'component,' identifying whether the event relates to assignments, quizzes, or other activities; 'eventname,' specifying the type of action, such as 'Course module viewed,' 'Assignment submitted,' or 'Quiz attempt started'; 'objecttable,' indicating which Moodle table the event is associated with, such as 'assign_submission' or 'quiz_attempts'; and 'userid,' identifying the student performing the action.

Each procrastination-related feature was computed based on these logs. TaskStartD corresponds to the earliest Course module viewed event for a given task, while TaskSubD is derived from the Assignment submitted event. TaskOpenD and TaskDueD were extracted from the course's assignment settings, while TaskAvailTime was calculated as the difference between TaskDueD and TaskOpenD.

Pseudocode 1 is an algorithm to create vector X that measures how much students procrastinate when submitting their tasks. To create this vector, we need to know each task's start date, open date, submission time, and due date. Then, we calculate several variables related to task submissions behavior, such as the delay in starting the task and the proportion of time spent on submitting it.

D. Clustering

This section explains the steps involved in clustering academic procrastination behavior. To obtain a clearer picture of student groups, we want to cluster students based on similar characteristics and determine the type of procrastination from the clusters. We must find the optimal number of clusters and apply the clustering model using the K-means algorithm. We labelled and explained the characteristics of each cluster.

E. Data Normalization & Standardization

Before determining the optimal number of clusters, we standardized the dataset to ensure that all features were on a comparable scale and to prevent any single variable from disproportionately influencing the clustering process. Z-score scaling was applied, transforming each feature into a standardized distribution with a mean of 0 and a standard deviation of 1. This transformation was performed by subtracting the mean of each feature from its individual values and then dividing the result by the standard deviation of that feature. As a result, the dataset was rescaled to approximate a normal distribution, ensuring that all features contributed equally to the clustering process without being affected by differences in magnitude or unit scale.

$$z = \frac{x-\mu}{\sigma} \quad (7)$$

with z is the z-score, x the original value, μ the mean, and σ the standard deviation.

Table 1. Notation of a task procrastination behavior vector.

Notation	Description	Moodle Log Source
S, n	Set of students and the total number of tasks for each student	userid from logstore_standard_log
$TaskStartD$	Start date and time of taking the task	timecreated where eventname = 'Course module viewed'
$TaskSubD$	End date and time of submitting the task	timecreated where eventname = 'Assignment submitted'
$TaskOpenD$	Task opening date and time	Extracted from course settings (assign table)
$TaskDueD$	Task closing date and time	Extracted from course settings (assign table)
$TaskAvailTime$	Time provided to complete the task	Difference between TaskDueD and TaskOpenD
$TaskStartPro$	Procrastination time to start taking the task	Difference between TaskStartD and TaskOpenD
$TaskSubmitPro$	Procrastination time in submitting the task	Difference between TaskSubD and TaskStartD
$TaskScore$	The score for each task	Extracted from the grade_grades table
$TaskDoTime$	Time to complete the task	Cumulative engagement time between TaskStartD and TaskSubD
$TaskCompRatio$	Task completion ratio	Ratio of TaskDoTime to TaskAvailTime
$TaskDelay$	The delay time to complete the task from start to submission.	Difference between TaskSubD and TaskStartD
$best\_model$	The best model of classification	Model selection results
$best\_acc$	Best accuracy	Model evaluation results

Pseudocode 1:

Building procrastination vector
Input: $TaskStartD$, $TaskOpenD$, $TimeSubTask$, $TaskDueD$
Output: Vector X

```
Initialize variables
X = [ ]
while i <= n:
    Calculate TaskAvailTime by (1)
    Calculate TaskDelay by (2)
    Calculate TaskDoTime by (3)
    Calculate TaskCompRatio by (4)
    Calculate TaskStartPro by (5)
    Calculate TaskSubPro by (6)
    X.append([TaskAvailTime, TaskDelay,
    TaskDoTime, TaskCompRatio,
    TaskStartPro, TaskSubPro])
    i += 1
Return X
```

F. Determining the Optimal Number of Clusters

To find the optimal number of clusters, we use the elbow method [24], which is to calculate the inertia value (the sum of squared distances between each point and its cluster center) for different numbers of clusters, and choose the point “elbow” where the decrease in inertia value becomes smaller. The inertia value can be calculated using the following formula:

$$I = \sum_{i=1}^{n} \min_{\mu_j \in C} (|| x_i - \mu_j ||^2) \tag{8}$$

with I is the inertia value, n is the number of samples, C is the set of cluster centers, x_i is the feature vector for sample i, and μ_j is the cluster center for j. A lower inertia value indicates that the samples in the cluster are closer to their center.

After determining the optimal number of clusters, we applied the k-means algorithm to group samples based on procrastination behavior variables. After obtaining the clustering results, we labelled and explained the characteristics of each cluster.

Next, a loop with a value of k between 1 and 10 was performed to calculate the Sum of Squared Errors (SSE) of each cluster. KMeans with k clusters were used to calculate the SSE value. The SSE value determines the optimal number of clusters during the clustering process. A plot of the SSE was compared with the number of clusters and then created to visualize the process results. The plot results are used to determine the optimal number of clusters by selecting the number of clusters with the lowest SSE value that can still distinguish and describe important information in the data.

$$SSE = \sum_{i=1}^{n} \min_{\mu_j \in C} (|| x_i - \mu_j ||^2) \tag{9}$$

with n is the number of samples, indicating the number of data points in the dataset. C is the set of cluster centers that indicates the locations of the centroids that represent each cluster. x_i is the feature vector for sample i, which means the values of the variables that describe the data point i. μ_jis the cluster center for j, which means the centroid that is closest to the data point i. (||x_i-μ_j||^2 is the squared Euclidean distance between the data point i and the cluster center j, which means how far apart they are.

G. Applying the Clustering Model

After the graph is formed, we determine the optimal number of clusters and perform clustering using the k-means algorithm. Unlike previous studies, we included other variables, namely TaskStartPro, TaskSubPro, TaskDelay, TaskCompRatio, and TaskScore, to better understand the characteristics of students in each cluster. Subsequently, the characteristics of each cluster are labeled and explained. To process the selection of the number of clusters k, we set k to no more than 10 to save computing time and resources. Pseudocode 2 is an algorithm for clustering procrastination behavior based on the best number of groups (clusters) for the procrastination data. We need a list of procrastination data called Vector X from Pseudocode 1 as the input. We then calculate the inertia for each number of clusters using the K-means clustering algorithm and keep track of the minimum inertia value and index. After determining the optimal number of clusters, we display an inertia plot to visualize the results. Then, we performed clustering with this optimal number of clusters and displayed the mean value characteristics for each cluster.

In summary, this algorithm helps us determine how many groups are needed to group similar procrastination behavior data and provides insights into each group's characteristics.

Pseudocode 2.
Clustering Behavior Procrastination

```
Input: Vector X from Pseudocode 1
Output: Mean value characteristics of each
cluster, inertia plot elbow method, an optimal
number of cluster
    Initialize variables
    Standardize the data by (7)
    Find optimal k
    while k <= 10:
        Calculate inertia by (8)
          k += 1
    Display inertia plot
    Perform clustering with an optimal
    number of clustering using k-Means
    Display means value characteristics of
    each cluster.
    Display SSE using (9)
```

H. Prediction

This section discusses the prediction of the task scores using machine learning. First, we used feature selection to select the most relevant features for predicting scores. We then compared seven machine learning models to find the best model for predicting scores. Data were normalized, split into training and test sets, and cross-validated to avoid overfitting. The model with the highest R-squared score was selected as the best model. Finally, pseudocode 3 was provided as an algorithm for choosing the best predictive model for task scores.

I. Feature selection

Feature selection improves the accuracy and efficiency of predictions by selecting the most relevant subset of features, thereby reducing overfitting and the computation time. There are several methods of feature selection. However, we used SelectKBest, which is a feature selection method used to select the best K features based on a certain score. This is a simple but effective technique for reducing data dimensionality and improving model quality. We used this technique because it is often used in regression and classification problems.

$$F_i = \frac{MSB_i}{MSE_i} \tag{10}$$

with MSB_i The Mean Square Between (MSBi) measures how much variation in the target values can be explained by feature i(F_i). MSE_i (Mean Square Error) measures how much variation in target values cannot be explained by F_i.

J. Applying the Prediction Model

We used seven machine learning methods and compared them to predict the TaskScore value: decision tree regressor (DT), Random Forest Regressor (RF), Gradient Boosting Regressor (GB), Multilayer Perceptron Regressor (MP), Support Vector Regressor (SV), K-Nearest Neighbors Regressor (K-NN), and LightGBM Regressor (LGBM). The data were obtained from the dataset from several features selected in the previous explanation. Before applying the model, the data were normalized on the same scale using Z-Score Scaling.

The data were divided into two parts, namely training and testing data, with a ratio of 80:20, and training and model evaluation were carried out. We applied k-fold cross-validation with 5,10, and 15 folds and calculated each fold's average R2 score, MAE, and RMSE for each model. Thus, we expect the results to be more consistent and avoid overfitting the training data. The model was also trained on all available data, thus improving its performance. The evaluation results displayed each model's R2 score, mean absolute error, and root-mean-squared error. The higher the R2 score and the lower the Mean Absolute Error and Root Mean Squared Error, the better is the model performance. We also used several parameters to improve model performance and produce more accurate predictions.

The pseudocode 3 algorithm selects the best prediction model for the task scores. The algorithm performs feature selection, splits the dataset into training and testing sets, and uses several prediction models such as DT, RF, GB, MP, SV, K-NN, and LGBM regressors. This process involves training several models using a set of data. The performance of each model was evaluated using the R-squared value (r2). Finally, the best model was chosen. Overall, Pseudocode 3 helps improve the prediction by selecting the best prediction model for the task scores.

Pseudocode 3.
Selecting The Best Prediction Model

```
Input: TaskStartPro, TaskSubPro,
TaskDelay, TaskCompRatio, TaskScore
Output: The best model for prediction
TaskScore
Perform Pseudocode 1
Perform Pseudocode 2
Perform feature selection Z-Score
Scaling
features = get [features] from  (10)
X = df[features].values
y = df['TaskScore'].values
Split dataset into training and testing
sets 80:20
models = {'DT ', 'RF ', 'GB ', 'MP ',
'SVM ','K-NN ', 'LGBM'}
initialize variables i, r2, best_r2_score
while i <= len(models):
    for each models
        Training model using training
data:
        Making predictions on test data:
        Evaluating model's performance
metrics (r2)
        If r2 > best_r2_score
          Update bestmodel
  i += 1
```

The final dataset contained 7,896 rows of extracted data spanning multiple online courses. Table 2 shows the correlation matrix between the various metrics related to task completion behavior. TaskStartPro and TaskScore had a negative correlation of -0.777, suggesting that the task score decreased as the time for starting the task was postponed.

IV. RESULT AND DISCUSSION

A. Result

We employed two approaches to identify the optimal number of clusters: the elbow method and K-NN. We chose three clusters. We labeled them according to their procrastination levels and academic performance.

a. Determining the Optimal Cluster

The number of clusters was determined based on the elbow method, and the cluster method was performed with K-NN. We chose these two methods because they are commonly used in educational research and relatively easy to apply. The elbow method was used to determine the optimal cluster at this stage.

Fig. 2 shows a plot of SSE versus the number of clusters to determine the optimal number. We chose three clusters using the elbow method, which helps to determine the optimal number of clusters. We looked for the "elbow" point in the plot where the SSE decreased no further with increased clusters. This approach provided a more detailed description of each cluster.

b. Applying the Clustering Model

Table 3 presents three procrastination clusters based on TaskStartPro, TaskSubPro, and TaskDelay, categorizing students into different levels of procrastination and corresponding task performance. Cluster 1 comprises students with low procrastination and high performance; these students start tasks early, submit on time, and experience minimal delays, achieving task scores typically between 85 and 100.

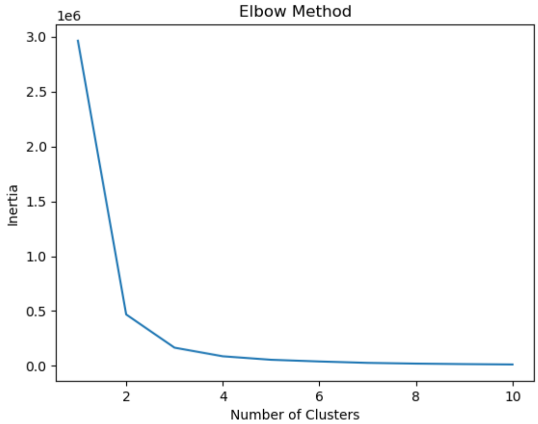

Figure 2. Determining the optimal cluster with elbow method

Cluster 2 includes students with high procrastination and poor performance; these individuals delay both starting and submitting tasks, often submitting close to or after the deadline, resulting in lower scores ranging from 40 to 55. Cluster 3 consists of students with medium procrastination and average performance; these students exhibit moderate delays in starting and submitting tasks, with performance scores ranging from 60 to 75.

In the prediction stage, we used various models to predict task scores based on multiple features, normalized the data, selected the best features, and defined different parameters for each model. The parameters of each model are shown in Table 4. We evaluate the performance of each model using metrics such as R2, MAE, and RMSE values, and the results are shown in Table 5. Based on Pseudocode 3, we experimented with predicting the Task Score based on several features, using various prediction models with different parameters. In the first step, we normalize the data using Z-Score Scaling, which involves scaling the features to a standard range to ensure equal contribution of each feature in the model. The program then selects the best features using SelectKBest with the f_regression score function and different numbers of k values (3, 5, and 7). We split the dataset into training and testing sets at 70:30. Finally, we evaluated the performance of each model using several metrics, such as the R2 score, MAE, and RMSE.

Table 2. Statistical analysis of data

Metric	TaskStartPro	TaskSubPro	TaskDelay	TaskAvailTime	TaskDoTime	TaskComp Ratio	TaskScore
TaskStartPro	1.000	-0.106	-0.543	0.480	0.594	0.106	-0.777
TaskSubPro	-0.106	1.000	-0.777	0.130	-0.087	-1.000	0.213
TaskDelay	-0.543	-0.777	1.000	-0.414	-0.303	0.777	0.312
TaskAvailTime	0.480	0.130	-0.414	1.000	0.927	-0.130	-0.383
TaskDoTime	0.594	-0.088	-0.303	0.927	1.000	0.088	-0.473
TaskCompRatio	0.106	-1.000	0.777	-0.130	0.088	1.000	-0.214
TaskScore	-0.777	0.213	0.312	-0.383	-0.473	-0.214	1.000
Count	7896	7896	7896	7896	7896	7896	7896
Median	0.590	0.100	0.240	7.230	5.730	0.900	58.000
Std	0.300	0.400	0.474	3.640	3.500	0.400	19.355
Mean	0.532	0.113	0.355	7.112	6.120	0.887	68.064

Table 3. Clustering procrastination behaviour

Cluster	TaskStartPro (Normalized)	TaskSubPro (Normalized)	TaskDelay (Normalized)	TaskScore (Range)	Label
1	0.327 (Starts early)	0.109 (Submits well before the deadline)	0.564 (Minimal delay)	85 - 100	Low procrastination, high performance
2	0.733 (Starts late)	0.075 (Submits close to or after the deadline)	0.191 (Long delay)	40 - 55	High procrastination, poor performance
3	0.574 (Moderate start)	0.139 (Submits close to the deadline)	0.287 (Moderate delay)	60 - 75	Medium procrastination, average performance

Table 4. Model's parameter

Model	Parameters
DT	max_depth=5,min_samples_split=2, min_samples_leaf=1
RF	n_estimators=100, max_depth=5,min_samples_split=2
GB	learning_rate=0.1,n_estimators=100, max_depth=5
MP	hidden_layer_sizes=(100,), activation='relu',solver='adam', learning_rate_init=0.001, max_iter=1000
SV	C=1.0, epsilon=0.1
K-NN	n_neighbors=5,weights='distance'. num_leaves=31
LGBM	learning_rate=0.05, n_estimators=100

Table 5. Performance of multimodel prediction with different numbers of features and evaluation metric.

Metric	Features	DT	RF	GB	MP	SV	K-NN	LGBM
R2	3	0.784	0.789	0.794	0.785	0.739	0.707	0.794
	5	0.806	0.810	0.802	0.794	0.769	0.726	0.803
	7	0.807	0.812	0.803	0.791	0.763	0.712	0.807
Metric	**Features**	**DT**	**RF**	**GB**	**MP**	**SV**	**K-NN**	**LGBM**
MAE	3	6.654	6.575	6.456	6.627	6.310	7.230	6.499
	5	6.321	6.284	6.327	6.544	6.230	7.017	6.404
	7	6.318	6.248	6.348	6.672	6.189	7.073	6.46
Metric	**Features**	**DT**	**RF**	**GB**	**MP**	**SV**	**K-NN**	**LGBM**
RMSE	3	9.077	8.965	8.853	9.051	9.972	10.572	8.849
	5	8.593	8.501	8.678	8.868	9.380	10.213	8.665
	7	8.578	8.456	8.628	9.025	9.549	10.392	8.582

Table 6. Model performance with various K values in K-fold cross-validation.

Model	K	R^2	MAE	RMSE
DT	5	0.796	6.351	8.733
	10	0.795	6.376	8.744
	15	0.795	6.380	8.733
RF	5	0.801	6.320	8.620
	10	0.801	6.321	8.622
	15	0.801	6.325	8.618
GB	5	0.798	6.354	8.701
	10	0.798	6.336	8.680
	15	0.798	6.338	8.672
MP	5	0.782	6.638	9.026
	10	0.783	6.625	8.995
	15	0.784	6.583	8.973
SV	5	0.765	6.053	9.382
	10	0.765	6.041	9.362
	15	0.765	6.039	9.355
K-NN	5	0.713	7.097	10.355
	10	0.712	7.131	10.361
	15	0.713	7.109	10.348
LGBM	5	0.790	6.474	8.855
	10	0.792	6.445	8.804
	15	0.792	6.440	8.796

In the first experiment, only three features were used. The models demonstrated good R2 scores, ranging from 0.739-0.794, indicating their ability to predict the data. Additionally, these models had relatively low MAE and RMSE values, implying their accuracy in predicting data.

Five features were used in the second experiment. Some models displayed a performance improvement, as indicated by a slight increase in the R2 score and MAE, but a slight decrease in the RMSE. However, K-NN performed poorly, with a decrease in the R2 score and MAE, but an increase in the RMSE. In the third experiment, the RF, GB, and LGBM models exhibited a slight increase in performance. The R2 scores for these models increased from the test using five features, with the RF achieving the highest R2 score of 0.812. The MAE also decreased slightly, while the RMSE for these models remained relatively stable.

Based on the metrics, the RF with seven features had the highest R2 score (0.812) and the lowest MAE (6.248) and RMSE (8.456), indicating that it performed the best among all the models tested. Therefore, this model is recommended for this dataset. Furthermore, the number of features used significantly affected the model performance.

To ensure the accuracy and reliability of the predictive models, we applied the k-fold cross-validation technique with k values of 5, 10, and 15. Such an approach reduces bias and variance in model evaluation. The performance metrics R^2, Mean Absolute Error (MAE), and Root Mean Squared Error (RMSE)—for each model across different k values are presented in Table 6.

B. DISCUSSION

The results indicate that students with higher procrastination levels tend to achieve lower academic performance. This finding aligns with prior research suggesting that procrastination negatively impacts learning due to reduced study time, increased cognitive load, and last-minute task completion, which affects comprehension and performance quality. Additionally, students with high procrastination behaviors often struggle with time management and motivation, leading to increased stress and ineffective learning strategies. These factors contribute to lower task scores in Cluster 2 (high procrastination, poor performance). The moderate performance observed in Cluster 3 suggests that while some procrastination may not drastically impact academic success, excessive procrastination (as in Cluster 2) results in significantly lower grades.

V. CONCLUSION

This study successfully applied cluster analysis to categorize students based on their procrastination behaviors, determining the optimal number of clusters using the Elbow method, followed by K-Means clustering for student grouping. The results identified three distinct procrastination clusters, where Cluster 1 represents low procrastinators with good academic performance, Cluster 2 consists of high procrastinators with poor academic performance, and Cluster 3 includes medium procrastinators with moderate academic performance. The clustering results highlight the significant role of procrastination in academic outcomes, demonstrating that students with lower procrastination tendencies tend to achieve better performance, whereas higher procrastination is associated with lower scores. These insights validate the effectiveness of TaskStartPro, TaskSubPro, and TaskDelay as key procrastination indicators in analyzing student behavior.

Beyond clustering, we successfully predicted student task scores based on procrastination behavior. Among the models tested, Random Forest (RF) achieved the highest accuracy, with an R^2 score of 0.812, MAE of 6.248, and RMSE of 8.456, outperforming other machine learning models. The results suggest that integrating procrastination-based clustering with prediction models can improve student performance forecasting in online learning environments.

To further enhance the clustering model, future research could explore feature selection improvements by incorporating additional student activity logs, such as learning resource access frequency and discussion forum interactions, to refine the procrastination classification. Enhancing the dataset with these behavioral indicators may offer deeper insights into the impact of procrastination patterns on academic performance. Additionally, integrating clustering-based student analytics into Learning Management Systems (LMS) like Moodle through real-time procrastination monitoring dashboards could allow for adaptive learning interventions, enabling instructors to identify students in high procrastination clusters and provide personalized support, deadline reminders, or guided study strategies to enhance their academic performance.

Future research should expand on predicting online learner performance by integrating clickstream behavior analysis with advanced ensemble learning methods such as stacking, bagging, and boosting. This approach would enable more dynamic and personalized learning analytics, helping educators identify at-risk students earlier and refine intervention strategies for improving engagement in online learning environments.

ACKNOWLEDGMENTS

The authors express their gratitude to the Ministry of Education, Culture, Research, and Technology of the Republic of Indonesia and BRIN for supporting this research through the PDUPT funding scheme.

REFERENCES

[1] S. Ghasempour, A. Babaei, S. Nouri, M. H. Basirinezhad, and A. Abbasi, “Relationship between academic procrastination, self-esteem, and moral intelligence among medical sciences students: a cross-sectional study,” BMC Psychol., vol. 12, no. 1, pp. 1–8, 2024, doi: 10.1186/s40359-024-01731-8.

[2] X. Li et al., “The relationship between academic procrastination and anxiety symptoms among medical students during the COVID-19 pandemic: exploring the moderated mediation effects of resilience and social support,” BMC Psychiatry, vol. 24, no. 1, 2024, doi: 10.1186/s12888-024-06202-3.

[3] E. M. Peixoto, A. C. Pallini, R. J. Vallerand, S. Rahimi, and M. V. Silva, “The role of passion for studies on academic procrastination and mental health during the COVID-19 pandemic,” Soc. Psychol. Educ., vol. 24, no. 3, pp. 877–893, 2021, doi: 10.1007/s11218-021-09636-9.

[4] M. Zhu, “Enhancing MOOC Learners’ Skills for Self-Directed Learning,” Distance Educ., 2021, doi: 10.1080/01587919.2021.1956302.

[5] J. R. Rico-Juan, C. Cachero, and H. Macià, “Study regarding the influence of a student’s

personality and an LMS usage profile on learning performance using machine learning techniques," Appl. Intell., vol. 54, no. 8, pp. 6175–6197, 2024, doi: 10.1007/s10489-024-05483-1.

[6] C. K. Gadosey et al., "Vicious and virtuous relationships between procrastination and emotions: an investigation of the reciprocal relationship between academic procrastination and learning-related anxiety and hope," Eur. J. Psychol. Educ., vol. 39, no. 3, pp. 2005–2031, 2024, doi: 10.1007/s10212-023-00756-8.

[7] M. M. Pereira, M. Kubrusly, A. B. T. M. dos Santos, M. do Nascimento Oliveira, L. O. Coimbra, and H. A. L. Rocha, "Association between procrastination and learning strategies in medical students in a hybrid problem-based and lecture-based learning curriculum," BMC Med. Educ., vol. 24, no. 1, 2024, doi: 10.1186/s12909-024-06306-0.

[8] M. Gil, R. El Sherif, M. Pluye, B. C. M. Fung, R. Grad, and P. Pluye, "Towards a Knowledge-Based Recommender System for Linking Electronic Patient Records with Continuing Medical Education Information at the Point of Care," IEEE Access, vol. 7, pp. 15955–15966, 2019, doi: 10.1109/ACCESS.2019.2894421.

[9] J. Conde, S. López-Pernas, E. Barra, and M. Saqr, "The Temporal Dynamics of Procrastination and its Impact on Academic Performance: The Case of a Task-oriented Programming Course," Proc. ACM Symp. Appl. Comput., pp. 48–55, 2024, doi: 10.1145/3605098.3636072.

[10] C. Imhof, I. S. Comsa, M. Hlosta, B. Parsaeifard, I. Moser, and P. Bergamin, "Prediction of Dilatory Behavior in eLearning: A Comparison of Multiple Machine Learning Models," IEEE Trans. Learn. Technol., vol. 16, no. 5, pp. 648–663, 2023, doi: 10.1109/TLT.2022.3221495.

[11] J. Pecuchova and M. Drlik, "Enhancing the Early Student Dropout Prediction Model Through Clustering Analysis of Students' Digital Traces," IEEE Access, vol. 12, no. September, pp. 159336–159367, 2024, doi: 10.1109/ACCESS.2024.3486762.

[12] M. Liz-Dominguez, M. Llamas-Nistal, M. Caeiro-Rodriguez, and F. Mikic-Fonte, "LMS Logs and Student Performance: The Influence of Retaking a Course." 2022, doi: 10.1109/EDUCON52537.2022.9766691.

[13] M. A. Martinie, A. Potocki, and L. Broc, "students : role of achievement goals and learning strategies," Soc. Psychol. Educ., no. 0123456789, 2022, doi: 10.1007/s11218-022-09743-1.

[14] D. Wang, D.-Q. Lian, Y. Xing, S. Dong, X. Sun, and J. Yu, "Analysis and Prediction of Influencing Factors of College Student Achievement Based on Machine Learning," Frontiers in Psychology. 2022, doi: 10.3389/FPSYG.2022.881859.

[15] P. Steel and K. B. Klingsieck, "Academic Procrastination: Psychological Antecedents Revisited," Aust. Psychol., vol. 51, no. 1, pp. 36–46, Feb. 2016, doi: 10.1111/ap.12173.

[16] F. Rodriguez, H. R. Lee, T. Rutherford, C. Fischer, E. Potma, and M. Warschauer, "Using clickstream data mining techniques to understand and support first-generation college students in an online chemistry course," in 11th International Learning Analytics and Knowledge Conference (LAK21), 2021, pp. 313–322, doi: 10.1145/3448139.3448169.

[17] M. Goroshit, "Academic procrastination and academic performance: An initial basis for intervention," J. Prev. Interv. Community, vol. 46, no. 2, pp. 131–142, Apr. 2018, doi: 10.1080/10852352.2016.1198157.

[18] J. Melgaard, R. Monir, L. A. Lasrado, and A. Fagerstrøm, "ScienceDirect ScienceDirect Academic Procrastination and Online Learning During the COVID- 19 Pandemic Academic Procrastination and Online Learning During the COVID- 19 Pandemic," Procedia Comput. Sci., vol. 196, no. 2021, pp. 117–124, 2022, doi: 10.1016/j.procs.2021.11.080.

[19] Y. Yang, D. Hooshyar, M. Pedaste, M. Wang, Y. M. Huang, and H. Lim, "Predicting course achievement of university students based on their procrastination behaviour on Moodle," Soft Comput., vol. 24, pp. 18777–18793, 2020, doi: 10.1007/s00500-020-05110-4.

[20] D. Hooshyar, M. Pedaste, and Y. Yang, "Mining educational data to predict students' performance through procrastination behavior," Entropy, vol. 22, no. 1, p. 12, 2020, doi: 10.3390/e22010012.

[21] L. Visser, F. A. J. Korthagen, and J. Schoonenboom, "Differences in learning characteristics between students with high, average, and low levels of academic procrastination: Students' views on factors influencing their learning," Front. Psychol., vol. 9, no. MAY, May 2018, doi: 10.3389/fpsyg.2018.00808.

[22] F. Rodriguez, H. R. Lee, T. Rutherford, C. Fischer, E. Potma, and M. Warschauer, "Using clickstream data mining techniques to understand and support first-generation college students in an online chemistry course," in ACM International Conference

Proceeding Series, Apr. 2021, pp. 313–322, doi: 10.1145/3448139.3448169.

[23] Y. Yang, D. Hooshyar, M. Pedaste, M. Wang, Y. M. Huang, and H. Lim, “Prediction of students’ procrastination behaviour through their submission behavioural pattern in online learning,” J. Ambient Intell. Humaniz. Comput., 2020, doi: 10.1007/s12652-020-02041-8.

[24] H. Humaira and R. Rasyidah, “Determining The Appropiate Cluster Number Using Elbow Method for K-Means Algorithm,” no. January, 2020, doi: 10.4108/eai.24-1-2018.229238