Ingénierie des Systèmes d'Information
Vol. 29, No. 4, August, 2024, pp. 1365-1379
Journal homepage: http://iieta.org/journals/isi

Comparative Analysis of Classification Algorithms Using Feature Selection Techniques to Predict On-Time Student Graduation

Haryono Setiadi*, Krisna Sanjaya, Ardhi Wijayanto, Dewi Wisnu Wardhani, Hasan Dwi Cahyono

Research Group Data Information Knowledge and Engineering, Department of Informatics, Universitas Sebelas Maret, Surakarta 57126, Indonesia

Corresponding Author Email: hsd@staff.uns.ac.id

https://doi.org/10.18280/isi.290412

Received: 3 August 2023
Revised: 16 January 2024
Accepted: 1 August 2024
Available online: 21 August 2024

Keywords:
educational data mining, classification, feature selection, on-time graduation prediction

ABSTRACT

On-time graduation rates are crucial for universities, impacting institutional performance and student success. At Sebelas Maret University, only 32% of the 2019-2020 postgraduate cohort graduated on time, exemplifying a common higher education challenge. This study compares naïve bayes, (NB) K-nearest neighbor (KNN), and decision tree (DT) algorithms, chosen for their effectiveness in educational data mining (EDM). Forward selection (FwS) and backward elimination (BE) techniques were implemented to optimize feature selection (FS), balancing model complexity and predictive power. Previous studies have primarily focused on graduation prediction, but few have thoroughly compared FS methods. This study compares NB, KNN, and DT algorithms, implementing FwS and BE for feature optimization. Results show that FS improved model performance across all algorithms. KNN and DT algorithms showed a more favorable impact with FwS, while BE proved more effective for the NB algorithm. The KNN algorithm with FwS achieved the highest accuracy at 83.8%, a significant improvement from its baseline accuracy of 76.64%. These findings can guide the development of support systems to improve on-time graduation rates, potentially benefiting institutions facing similar challenges. By evaluating these features, institutions can enhance their educational quality and support students in achieving timely graduation.

1. INTRODUCTION

Timely graduation rates have far-reaching implications for various stakeholders in higher education [1]. For students, on-time graduation can lead to reduced educational costs and earlier entry into the job market. Universities benefit from improved performance metrics and resource allocation [2], while policymakers can use this data to inform decisions on educational funding and initiatives [3]. Therefore, understanding and improving on-time graduation rates is crucial for the overall enhancement of higher education systems [4].

The assessment of of higher education primarily revolves around quality education [5], and a crucial aspect of determining the standard is the timely graduation of students. Analyzing academic data of the 2019 and 2020 postgraduate cohorts graduating at Sebelas Maret University, a total of 843 students completed their studies. Among them, 572 experienced late graduations, while 271 graduated on time. This poses a challenge for the university, as only 32% achieved on-time graduation. This study focuses on identifying the underlying causes using available transactional data. The effective technique employed to address the problem is EDM, which has the potential to enhance the quality of the educational process [6-9]. By analyzing large datasets, EDM discovered patterns to predict events related to students [10]. Classification methods, which are part of the framework, played a vital role in learning how to classify data effectively [11].

EDM encompasses several classification methods, including NB, DT, Artificial Neural Networks (ANN), Random Forest (RF), and KNN [12]. NB offers the advantage of requiring only a amount of training data to determine the necessary parameters for classification [13], while DT showed high accuracy when dealing with extensive datasets [14]. Mawardi and Santun [15] conducted a study on website-based doctor selection classification, utilizing the C4.5 and KNN methods to analyze patient complaints. The findings demonstrated that KNN achieved the highest accuracy of 100%, while ANN was limited by being prone to overfitting and relying heavily on empirical approaches [16].

While classification methods are essential for predicting student outcomes, their effectiveness can be significantly enhanced through appropriate FS. The combination of robust classification algorithms with efficient FS techniques forms the cornerstone of this study's methodology.

In EDM, FS is a critical process that involves identifying the most relevant attributes or variables that contribute to the prediction model [17]. This technique not only improves model accuracy, but it also increases computational efficiency and interpretability of results [18]. FS is particularly important when dealing with large datasets containing numerous variables, as is often the case in educational data [19, 20].

Effectively handling a large number of features poses a significant challenge for classification methods in data prediction [21], such as high time complexity and low accuracy. Utilizing only relevant features, reduces time complexity and improves accuracy. FS emerges as a solution to enhance the accuracy of the constructed model [22], and is employed to improve and eliminate irrelevant datasets. The study aims to enhance the algorithm to accurately predict students' timely graduation .

FS methods can be broadly categorized into filter, wrapper, and embedded approaches methods [23]. Filter methods select features independently of the learning algorithm, while wrapper methods use the learning algorithm as a black box to score feature subsets. Embedded methods perform FS as part of the model construction process. The wrapper method is comprised of three feature categories, namely FwS, backward, and recursive feature elimination [24]. BE offers the advantage of handling a large number of datasets effectively [14]. However, FwS excels at searching for subsets that best suit the utilized algorithm, while recursive feature elimination is sensitive to datasets and is affected by outliers [25]. In a study conducted by Wah et al. [26], where wrapper and filter methods were compared, FwS and BE achieved the highest accuracy of 99.2% respectively, outperforming algorithms, such as information gain and correlated-based FS. A study conducted by Usman et al. [27] on FS for student performance, employing both filter and wrapper methods, showed that the wrapper outperformed the filter techniques.

While previous studies have explored various aspects of predicting timely graduation, there remains a gap in the comprehensive comparison of FS techniques in combination with different classification methods, particularly in the context of on-time graduation prediction. This study addresses this gap by systematically evaluating the performance of FwS and BE techniques with DT, NB, and KNN algorithms. The novelty of this research lies in its systematic approach to identifying the most effective combination of FS and classification methods for predicting on-time graduation, which has not been extensively explored in previous literature.

2. RELATED WORK

This section discussed how previous studies have utilized the selected classification algorithms, namely DT, NB, and KNN. Additionally, it investigated the implementation of FwS and BE applied to these algorithms.

2.1 DT classifier on on-time graduation

The primary focus of universities is to ensure academic success and student retention, necessitating the development of a system capable of predicting these outcomes using pre-lecture academic data, demographics, profiles, and others. Li et al. [28] conducted a study that showed that DT exhibited a high accuracy rate of 94.4% in predicting academic success and student retention.

Graduation played a vital role in determining university standards, as students who fail to complete their studies significantly impact the overall excellence of the institution. Arifin and Hadiana [14] utilized DT and proved successful in predicting dropout incidents. This algorithm demonstrated an impressive accuracy rate of 82.52% in effectively identifying students at risk of dropping out.

Ensuring graduation is of significant importance when formulating strategic policies within the universities. Gotardo [25] conducted a study on predicting student performance using DT and the implementation of this algorithm effectively identified patterns achieving exceptional accuracy of 91.67%.

DT is widely regarded as the most effective model for predicting the timely graduation of students. Gunawan et al. [29] proposed the use of DT in predicting performance and the findings showed that this algorithm achieved an impressive accuracy rate of 78.612%. Table 1 shows the implementation of DT on student performance.

Table 1. Implementation of DT on student performance

Author	Method	Results
Arifin et al. [14]	DT & FwS	The implementation of DT predicted the incidence of dropout in future student case studies. In this case, DT succeeded in predicting student dropout with an accuracy of 82.52%.
Gotardo [25]	DT	The implementation of DT succeeded in studying patterns of events on student performance with the best accuracy of 91.67%.
Li et al. [28]	KNN & DT	DT identified factors influencing academic success and student retention such as school academic data, demographics, student profiles, and others. This model has an accuracy contribution of 94.4%.
Gunawan et al. [29]	DT	The implementation of DT predicted student performance with an accuracy of 78.612%.

2.2 KNN classifier on on-time graduation

Wirawan et al. [30] developed the KNN algorithm which proved valuable for universities in formulating academic strategies to improve educational quality. The findings showed that KNN achieved an impressive accuracy rate of 89.82% in predicting the timely graduation of students. Salim et al. [31] highlighted the efficacy of the KNN classification method and the implementation aimed to evaluate academic policies in universities.

Data mining offers a range of algorithms that are used to predict student graduation. Li et al. [28] conducted a study using KNN, enabling the identification of factors influencing academic success and student retention, such as pre-lecture academic data, demographics, student profiles, and other variables with an accuracy exceeding 90%. Wiyono et al. [32] also utilized KNN (k=5) to predict the timely graduation of students with a rate of 94.5%. It's worth noting that many studies do not report detailed algorithm parameters, which limits our ability to fully compare and replicate results. Future research in this area would benefit from more transparent reporting of algorithm settings and hyperparameters. Table 2 demonstrates the implementation of KNN on student performance.

2.3 NB classifier on on-time graduation

Classification algorithm methods significantly influence the identification of learning patterns related to student performance. Almarabeh [10] highlighted the successful prediction and improvement in educational outcomes through the utilization of the NB algorithm, achieving an accuracy rate of 85.4%. Additionally, Pujianto and Qomaria [13] demonstrated the ability of NB to predict student graduation with an impressive rate of 95.49%.

Table 2. Implementation of KNN on student performance

Author	Method	Results
Li et al. [28]	DT& KNN	KNN identifies factors influencing academic success and student retention such as pre-lecture academic data, demographics, student profiles, and others. This model contributes to the prediction of academic success and student retention with an accuracy of above 90%.
Wirawan et al. [30]	KNN	KNN assists universities in making academic strategies to improve education quality. The results show that this algorithm can predict the on-time graduation of students with an accuracy of 89.82%.
Salim et al. [31]	KNN	The implementation of KNN was used in early student graduation studies to evaluate university academic policies.
Wiyono et al. [32]	KNN	KNN (k = 5) predicts student on-time graduation with an accuracy of 94.5%.

In another study conducted by Saifudin et al. [33], the NB algorithm and FS was developed to determine the influential attributes affecting student performance. The findings showed the effectiveness of identifying the factors contributing to student failure in the educational system. This combination not only facilitates the prediction of student graduation but also enables the identification of influential attributes. Usman et al. [27] emphasized the significance of prediction and successfully enhancing student performance by combining the NB algorithm with the wrapper method. Table 3 demonstrated the implementation of NB on student performance.

Table 3. Implementation of NB on student performance

Author	Method	Results
Almarabeh [10]	NB, KNN, DT	The use of NB successfully predicted student performance and developed student educational performance with an accuracy of 84.07%.
Pujianto and Qomaria [13]	NB, KNN	NB predicts student graduation well with an accuracy of 95.49%.
Usman et al. [27]	NB & wrapper method	NB was successfully combined using the wrapper method to improve performance.
Saifudin et al. [33]	NB & FS	The use of NB and FwS predict student performance and identify attributes influencing student graduation.

2.4 Implementation of FwS

In a study conducted by Arifin et al. [14], the FwS method and the DT algorithm were developed for predicting student dropout. The implementation successfully enhanced the performance of DT and identified relevant attributes for forecasting the target variable.

In selecting prominent features in a model, Saifudin et al. [33] utilized the FwS method to determine the influential attributes impacting student performance. By combining NB with FwS, student graduation was speculated while identifying the impactful factors.

Maulana [34] conducted a study on predicting graduation using FwS, which demonstrated the ability to learn patterns of events in students. It emerged as the optimal method for forecasting the timely graduation of students. Table 4 showed the implementation of FwS on student performance.

Table 4. Implementation of FwS on student performance

Author	Method	Results
Arifin et al. [14]	FwS & DT	The implementation of FwS successfully improves the performance of DT and selects relevant attributes in predicting targets.
Saifudin et al. [33]	NB & FwS	The use of NB and feature selection predict student performance and identify attributes influencing student graduation.
Maulana [34]	FwS & DT	FwS can study patterns of events in students and is the best method for predicting student on-time graduation.

2.5 Implementation of BE

The use of BE proved instrumental in enhancing model performance accuracy. England et al. [35] examined student anxiety regarding academic performance, and the application of BE significantly improved accuracy using regression for predicting alumni job placement.

Bode et al. [36] conducted a study focusing on forecasting student learning satisfaction, employing BE and the successful implementation allowed for the selection of relevant features and led to improved algorithm accuracy.

Similarly, Thangavel et al. [37] utilized this approach to determine student recommendations for company placement and the implementation effectively identified the most probable assignment and served as a motivation for them to strive for better opportunities. Table 5 showed the implementation of BE on student performance.

Table 5. Implementation of FwS on student performance

Author	Method	Results
England et al. [35]	BE & regression KNN	BE learns patterns of events in students and is the best method for predicting their performance.
Bode et al. [36]	BE	The implementation of backward BE successfully selected relevant features and improved the accuracy of the algorithm used in the study.
Thangavel et al. [37]	NB, DT & BE	The implementation of BE successfully identified the most probable placement status for students and motivated them to work harder to be placed in better companies.

2.6 Comparative impact of FS methods

Researchers often choose FwS for its computational efficiency, especially with large feature sets. BE, on the other hand, is often preferred when there's a strong theoretical basis for including most features, and the goal is to remove only the least important ones. The impact of FwS and backward removal on DT, KNN, and NB classifiers varies depending on the dataset and algorithm used. Li et al. [38] found FwS to be more efficient for DTs, especially in large datasets.

Venkatesh and Anuradha [23] found BE to be more effective for KNN classifiers, especially with noisy features. Jain and Singh [39] found FwS to be more beneficial for NB classifiers, as it helps select relevant features and reduces overfitting. However, the computational efficiency of these approaches may vary. Xue et al. [40] found FwS to be faster for smaller datasets, while BE yielded better results for larger feature sets.

2.7 Comparative performance of classification algorithms

In the context of predicting on-time graduation, the performance of DT, KNN, and NB classifiers might vary greatly according on the chosen FS approach and the nature of the dataset. Chen et al. [41] showed that integrating DT, KNN, and NB classifiers using ensemble methods generally yielded better results than using each algorithm individually, especially when employing a hybrid FS approach. However, Adekitan and Salau [42] reported that NB classifiers demonstrated greater performance with BE when dealing with demographic factors, perhaps because to their capacity to manage conditional independence across features. The impact of specific variables on model accuracy can also differ; Aulck et al. [43] showed that first-year GPA and course load significantly improved forecasts across all three algorithms, whereas the influence of demographic characteristics differed. Interestingly, Hutt et al. [44] reported that the incorporation of non-academic characteristics, such as extracurricular activities and financial aid status, greatly boosted the performance of KNN models but had negligible impact on DT and NB classifiers. These findings underline the necessity of rigorous FS and method choice in EDM applications.

2.8 Impact of contextual factors

Several studies have highlighted the importance of specific factors in predicting on-time graduation. Pre-lecture academic data, demographics, and student profiles were consistently important across multiple studies [28, 30]. However, the relative importance of these factors varied, suggesting that model development should consider institution-specific contexts. Future research could benefit from a more systematic examination of how different factors impact model performance across various institutional settings.

3. METHOD

This section presented the methodology employed in the study which comprised various stages. Data collection was first performed to gather the necessary information. Subsequently, preprocessing was carried out to cleanse and transform the data. The dataset was then divided into training and testing sets using a 10-fold cross-validation technique. FS and classification algorithms were applied to identify the relevant attributes and predict the target variable. The performance of the model was evaluated using metrics such as accuracy, precision, recall, and AUC. The study phases are depicted in Figure 1.

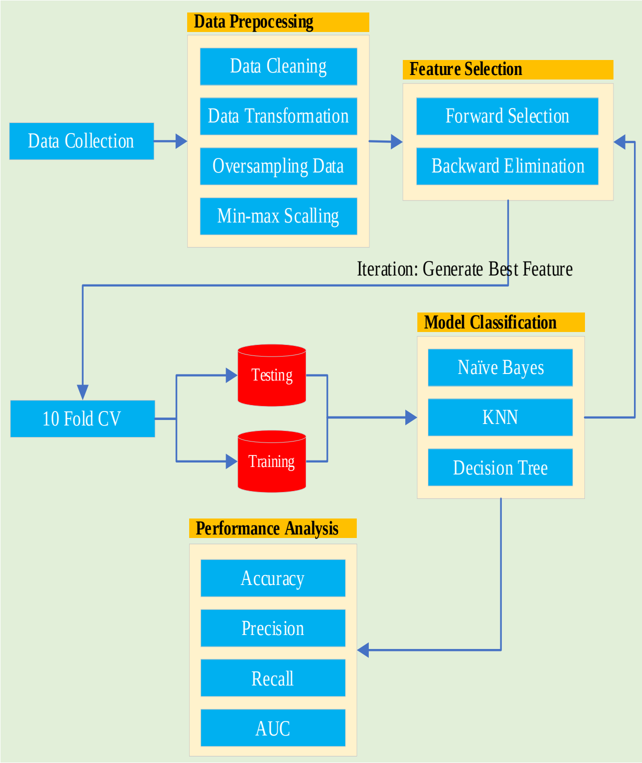

Figure 1. Flow of study methods

3.1 Data collection

This study utilized data from the Academic Information System of Sebelas Maret University (UNS), which was managed by the Information and Communication Technology Technical Implementation Unit (UPT TIK). The data collection process adhered to strict ethical guidelines. The university's Institutional Review Board (IRB) granted permission for the use of student data. The criteria for data inclusion were as follows: all postgraduate students who graduated in 2019–2020 were included, regardless of their program or department. Data was accessed through a secure, password-protected interface provided by the UPT TIK. To ensure student privacy, all personally identifiable information was anonymized before analysis.

The academic data used as features included Semester Achievement Index (SAI), gender, housing status, and other relevant variables. Meanwhile, the data labels indicated whether a student graduated on time or experienced delayed graduation.

3.2 Data cleaning

In the data cleaning process, we encountered missing values in approximately 5% of the records. The imputation method was chosen based on the distribution of each variable. For normally distributed continuous variables, mean imputation was used. For skewed continuous variables, median imputation was employed. For categorical variables, mode imputation was applied. We acknowledge that these imputation methods may introduce some bias, particularly for variables with a higher proportion of missing data. To assess the impact of imputation, we conducted a sensitivity analysis by comparing results with and without imputed data. The student identification numbers (SIN) were anonymized to safeguard individuals' privacy. Data cleaning aimed to improve the overall quality of the data before further processing.

3.3 Data transformation

For the data transformation process, categorical variables were encoded using one-hot encoding. This method was chosen to avoid introducing ordinal relationships where they don't exist. For example, the student's origin university was transformed into binary variables for each unique university. This approach allows the model to treat each category independently. For the class label (graduation time), we used binary encoding: 0 for on-time graduation (≤ 4 semesters) and 1 for late graduation (> 4 semesters). This binary classification aligns with our research question and simplifies the interpretation of results. Data transformation was applied to the class or label, which represented the duration of the student's year in semesters. The attributes were grouped, as presented in Table 6.

Table 6. Class/label grouping

Description	Value
0	**graduate on time**, namely the duration of the student's study ≤ (less than equal to) 4 semesters.
1	**graduate late**, namely the duration of the student's study > (more than) 4 semesters.

3.4 Random oversampling (ROS)

Several techniques such as Synthetic Minority Oversampling Technique (SMOTE), undersampling, and oversampling were employed to address data imbalance. However, SMOTE was ineffective for small datasets [45], while oversampling generally outperformed undersampling [46]. Given the imbalanced dataset in this study, the oversampling method was used. This method involved duplicating minority-labeled data instances until their number match those of the majority class. The aim was to achieve a balanced dataset.

3.5 Min-max scaling

Different data normalization techniques existed, including z-score and min-max scaling. In a breast cancer study conducted by Mawardi and Santun [15], the effectiveness of min-max scaling was compared to that of z-score normalization. The study found that min-max scaling achieved higher accuracy, leading to the adoption in this current study. By applying min-max scaling, the range of values across features was adjusted to ensure consistency, thereby enhancing the accuracy of the data mining analysis. For instance, suppose the mother's occupation feature ranged from 1-12, while the GPA ranged from 0-4. To maintain consistency and improve the accuracy of the data mining process, the dataset required normalization. The implementation of min-max scaling is represented by Eq. (1).

$$Xs = \frac{x - \min(x)}{(x) - (x)} \tag{1}$$

where, the variable x represents the data that is being transformed, where min(x) refers to the minimum attribute value and max(x) represents the maximum attribute value.

3.6 FS

The FS process in this study involved the utilization of the wrapper method. During this stage, all the features in the dataset were included in the selection process to determine the optimal accuracy. The performance of the wrapper approach was evaluated with the classification method to achieve the highest accuracy while identifying the relevant features in each iteration. The wrapper methods used were FwS and BE.

3.6.1 10 fold-cross validation

The data were partitioned into training and testing sets using k-fold cross-validation with 10 folds of equal size. The dataset was divided into ten subsets to evaluate the model or algorithm. The process of 10-fold cross-validation involved iterating the data 10 times, where in each iteration, one subset was used as the testing data while the remaining were used as the training information. This approach ensured that k-1 folds were used for model validation, while the remaining fold was used for construction [46, 47].

3.6.2 KNN

The KNN algorithm was employed to classify the student graduation dataset based on the majority of the nearest k values. This method involved selecting the value of k and calculating the distance between the testing point and the k training points. The distance calculation was performed using the Euclidean distance metric, considering attributes such as GPA, gender, housing status, and others. The k nearest points were then selected based on the calculated distances for each label. The predicted class was determined by identifying the category with the highest count among the KNN. The calculation of the Euclidean distance is represented by Eq. (2).

$$d(x, y) = \sqrt{\sum_{k=1}^{n} (Xk - Yk)^2} \tag{2}$$

In this context, Xk represents the value of the testing data point, Yk represents the value of the training data point, and n represents the number of attributes.

3.7 DT

The DT algorithm operated by selecting attributes such as GPA, gender, and housing status as the root node. The root node represented the starting point of DT without any incoming edges. Internal nodes are root nodes with outgoing edges.

In each iteration, the DT determined the root and internal nodes by calculating the smallest entropy for each feature sequentially, until distinct patterns emerged. These patterns were either internal or leaf nodes, which were used to determine the timely or delayed graduation. Entropy was calculated using Eq. (3) and gain using Eq. (4):

$$Entropy\ (S) = \sum_{i=1}^{n} - pi\ \log_2 pi \tag{3}$$

where, S=Set of cases; n=Number of partitions of S; pi=Partition of Si toward S.

After calculating entropy, the next step was to calculate the information gain using Eq. (4):

$$Gain\ (S,A) = entropy\ (S) - \sum_{i=1}^{n} \frac{|Si|}{S} * Entropy\ (Si) \tag{4}$$

where, S=Set of cases; |S|=Number of data samples; n=Number of partitions of attribute A; |Si|=Partition of Si toward S; A=Attribute.

3.8 NB

The NB algorithm utilized in this study determined the label to be assigned to the testing data by calculating the probability of membership based on the available characteristics such as GPA, gender, housing status, and others. The label with the highest probability was selected as the correct classification. The probability computation in NB was calculated using Eq. (5):

$$P\ (Y = c) = \frac{1}{\sqrt{2\pi\sigma_c^2}} e^{\frac{-(x-\mu c)^2}{2\sigma_c^2}} \tag{5}$$

where, *x*=Observed feature value; C=Class being evaluated; μc=Mean feature value in class C; σ_c^2=Variance of feature value in class C.

3.8.1 Result and evaluation interpretation

To assess the performance of the NB, DT, and KNN methods, we employed a confusion matrix, a widely-used tool for evaluating classification algorithm effectiveness. Utilizing this matrix, we calculated key performance metrics—accuracy, precision, and recall—using Eqs. (6)-(8) [48]. Accuracy quantifies the proximity between predicted and actual values, while precision evaluates the correctness of the selected data subset in relation to the required information. On the other hand, recall measures the system's efficiency in retrieving relevant data. Additionally, we computed the Area AUC alongside these metrics. We performed all calculations using the equations presented in Table 7, derived from the confusion matrix, to provide a comprehensive assessment of each algorithm's performance.

Table 7. Evaluation equations

Evaluation	Equation	
Accuracy	$\frac{(Tp+Tn)}{(Tp+Tn+Fp+Fn)}$	(6)
Precision	$\frac{(Tp)}{(Tp+Fp)}$	(7)
Precision	$\frac{(Tp)}{(Tp+Fn)}$	(8)
AUC	$\theta^r = \frac{1}{mn}\sum_j^n = 1 \sum_i^m = 1\ \varphi(xi^r, xj^r$	(9)

The calculation of AUC was used to measure the overall diagnostic accuracy test, ranging from 0 to 1. A higher AUC value indicated a better diagnostic test. They are divided into different levels based on the output as presented below [48]:

1. *Excellent*: 0.90 - 1.00
2. *Good:* 0.80 - 0.90
3. *Fair:* 0.70 - 0.80
4. *Poor:* 0.60 - 0.70
5. *Failure:* 0.50 - 0.60

4. RESULTS AND DISCUSSION

In this study, we used FwS and BE techniques to identify relevant features. These methods were chosen for their effectiveness in reducing dimensionality, improving model performance, and increasing interpretability. FwS and BE are FS techniques used to identify the most relevant features for a predictive model. FwS starts with no features and iteratively adds the most significant features, whereas BE starts with all features and gradually removes the least significant features. Using both techniques, we aim to comprehensively evaluate the importance of features and optimize our classification model. The experiments were conducted and executed using the Python programming language and Google Colab Pro.

4.1 Data collection

The dataset used in this study comprised academic information systemdata for graduate students at Universitas Sebelas Maret who completed their studies in 2019 and 2020. The raw data collected, including variables such as GPA, gender, housing status, and others show in Table 8.

Table 8. Data collection

Code	Variable	Description	
Mhsjk	Gender	[0,1]	0: Male 1: Female
Mhsstatrmh	Housing Status	[0,1,2,3,4,5]	1: Parent's house 2: Relative's house 3: Dormitory / Boarding house 4: Private house 5: Others
…	…	…	...
Ashwini	Nationality of the student	[1,2,3]	1: Indonesian native 2: Indonesian descent 3: Foreign citizen

4.2 Data cleaning

Once the dataset was collected, it underwent the data cleaning process to eliminate outliers and ensure accuracy. Data cleaning involved addressing spelling errors in free text information, handling data anomalies, resolving missing values, and eradicating redundant features. Table 9 displayed the findings of data cleaning.

Table 9. Data cleaning

Data Activity	Number of Data
Correction of data with spelling errors	150 data
Handling data with missing values	4 data
Removal of redundant features	1 feature

4.3 Data transformation

Following the data cleaning process, the dataset was transformed to convert it into the appropriate format for data mining. This involved converting the information into a numeric format, including the transformation of string data for attributes such as the original university and student graduation labels. The outcomes of the data transformation process are observed in Table 10.

Table 10. Data transformation results

University	Label Class
1	2
2	2
3	1
4	1
…	….

4.4 Random oversampling (ROS)

In this study, the oversampling method was utilized to address the imbalanced dataset. The "on-time graduation" label consisted of 567 data points, while the "late graduation" label only had 275 data points. To balance the dataset, the delayed graduation was duplicated through the application of oversampling, which led to an equal number of data points for both labels. The findings of the random oversampling process are accessed in Table 11.

Table 11. Random oversampling

Label	Number of Data
Late Graduation	550 data
On-Time Graduation	567 data

4.5 Min-max scaling

Data normalization was conducted using the min-max scaling technique, which standardized the range of values across all features. Following the conversion of all the features into a numeric format, min-max scaling was applied to normalize the values. The implementation findings of min-max scaling are observed in Table 12.

Table 12. Results of min-max scaling

Housing Status	…	Parent's Income	Student's Religion	SAI 1
0.75	…	0.5	0.00	0.8050
0.00	…	0.33	0.00	0.9075
0.00	…	0.5	0.4	0.9150
0.00	…	0.5	0.00	0.8525
…	…	…	…	…

4.6 Implementation of KNN and FwS

The subsequent procedure involved the computation of the KNN algorithm using FwS. The utilization of KNN with FwS was complemented by the application of 10-fold cross-validation, as illustrated in Table 13.

Table 13. Implementation results of KNN + FwS

Indicator	KNN	KNN + FwS
Accuracy	76.64%	83.8%
Precision	86.84%	89.7%
Recall	66.29%	76.89%
ROC AUC	76.78%	83.9%

After applying the KNN with FwS, a notable improvement in accuracy of 7.16% was observed, which indicated an improved performance of the model and more accurate predictions. Precision also increased by 2.86%, while recall showed a significant improvement of 10.6% for correctly predicting positive data, while the ROC AUC expanded to 7.12%. Based on the ROC AUC parameter, the implementation of KNN and FwS was classified as achieving "good classification."

The implementation also led to an accuracy rate of 83.8% and contributed to the removal of 7 irrelevant features from the model leading to enhanced performance. The reduction of features in the implementation is shown in Figure 2.

From Figure 2, it was concluded that there were variations in accuracy throughout the iteration of the model. A significant decline was observed starting from the 7th iteration, indicating that additional features did not further improve the accuracy. This highlighted the effectiveness of the feature removal process implemented through FwS, as it successfully enhanced the accuracy of KNN.

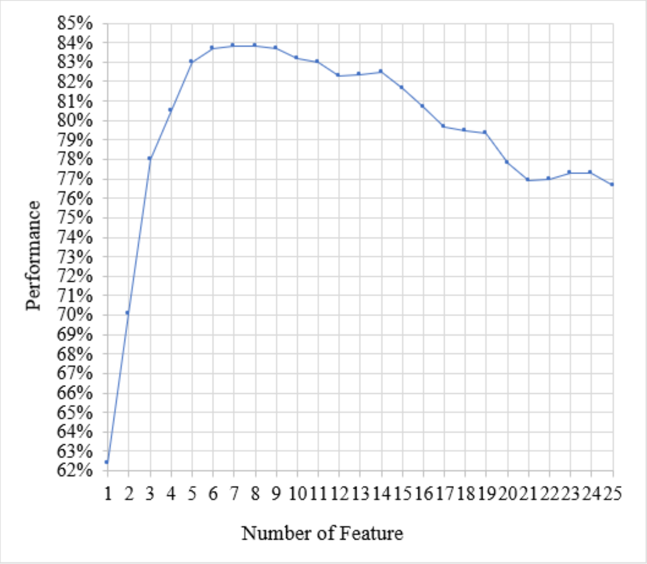

Figure 2. Feature reduction graph of KNN & FwS

4.7 Implementation of KNN and BE

The implementation of KNN with BE is complemented by the use of 10-fold cross-validation, which ensured robust evaluation and validation of the model. After performing the computation process of the KNN algorithm with BE, the findings are presented in Table 14.

Table 14. Implementation results of KNN + BE

Indicator	KNN	KNN + BE
Accuracy	76.64%	80.84%
Precision	86.84%	87.47%
Recall	66.29%	72.66%
ROC AUC	76.78%	80.96%

After conducting the computation process of KNN with BE, an improvement in accuracy of 4.2% was observed, indicating enhanced model performance and more accurate predictions. There was a slight increase of 0.63% in precision, while recall showed a significant improvement of 6.37% in correctly predicting positive data. The ROC AUC also demonstrated a notable increase of 4.18%. Based on the ROC AUC parameter, the implementation of KNN with BE was classified as "good classification."

This implementation achieved an accuracy of 80.84%, which was accompanied by the removal of up to 18 features in the model. The successful application of BE in reducing irrelevant features contributed to the high performance of the model, which is shown in Figure 3.

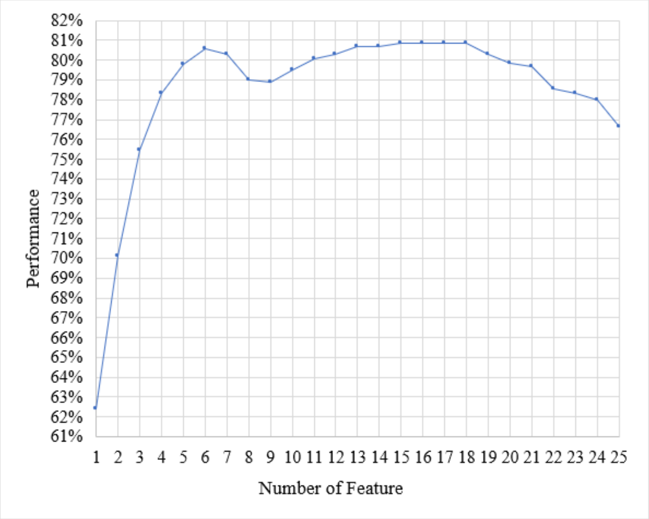

Figure 3. Feature reduction graph of KNN & BE

From Figure 3, it was concluded that the accuracy of the model fluctuated with each iteration. However, a notable decline was observed starting from the 18th iteration, indicating that further feature reduction did not lead to an improvement in accuracy. This finding demonstrated that the implementation of BE successfully enhanced the accuracy of KNN by eliminating irrelevant attributes from the model.

4.8 Implementation of DT and FwS

Furthermore, the DT algorithm was computed with the FwS technique. The implementation was calculated with the 10-fold cross-validation, as shown in Table 15.

Table 15. Implementation results of DT + FwS

Indicator	DT	DT + FwS
Accuracy	79.59%	82.7%
Precision	90.27%	89.61%
Recall	69.63%	74.6%
ROC AUC	79.72%	82.84%

After calculating the computation process on DT with FwS, an increase in accuracy of 3.11% was observed, indicating improved performance of the model and more accurate predictions. While there was a slight decrease of 0.66% in precision, the recall showed an improvement of 4.97% in correctly predicting positive data. The ROC AUC exhibited an increase of 3.12%. Based on the ROC AUC parameter, the implementation of DT and FwS was categorized as "good classification."

With an accuracy of 82.7%, the implementation of DT and FwS contributed to the removal of up to 18 features in the model. The success of FwS in eliminating irrelevant features led to high performance. The reduction of features in the implementation was observed in Figure 4.

From Figure 4, it was concluded that the accuracy of the model fluctuates with each iteration. However, a notable decline was observed starting from the 18th iteration, indicating that additional features did not lead to improvement in accuracy. This finding demonstrated that the implementation of FwS successfully enhanced the accuracy of DT by eliminating irrelevant attributes from the model.

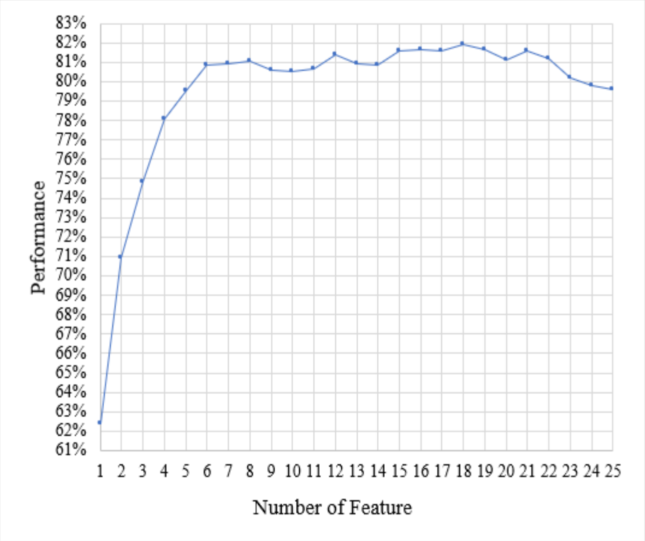

Figure 4. Feature reduction graph of DT & FS

4.9 Implementation of DT and BE

Furthermore, the DT was computed with BE. The implementation was calculated with the 10-fold cross-validation, as shown in Table 16.

Upon calculating the computation process of DT with BE, an increase in accuracy of 2.31% was observed, indicating an improved model performance and more accurate predictions. While there was a slight decrease of 2.02% in precision, the recall showed an improvement of 4.62% in correctly predicting positive data. The ROC AUC exhibited an increase of 2.31%. Based on the ROC AUC parameter, the implementation of DT and BE was categorized as "good classification".

With an accuracy rate of 82.7%, the implementation of DT and BE contributed to the removal of up to 18 features in the model. The success of BE in eradicating irrelevant features leads to high performance. The reduction of features in the implementation was observed in Figure 5.

From Figure 5, it was concluded that the accuracy fluctuated with each iteration of the model. However, a notable decline was observed starting from the 18th iteration, indicating that further feature reduction did not lead to an improvement in accuracy. This finding demonstrated that the implementation of BE successfully enhanced the accuracy of DT by eliminating irrelevant attributes from the model.

Table 16. Implementation results of DT + BE

Indicator	DT	DT + BE
Accuracy	79.59%	81.9%
Precision	90.27%	88.25%
Recall	69.63%	74.25%
ROC AUC	79.72%	82.03%

4.10 Implementation of NB and FwS

Furthermore, the NB was computed with the FwS technique. The implementation of NB with FwS was performed with the 10-fold cross-validation, as shown in Table 17.

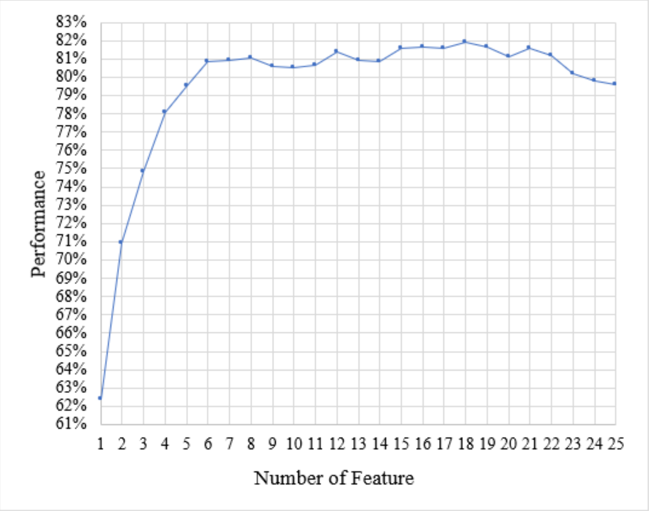

Figure 5. Feature reduction graph of DT & BE

Table 17. Implementation results of NB + FwS

Indicator	NB	NB + FwS
Accuracy	55.58%	63.9%
Precision	58.84%	67.98%
Recall	39.26%	54.67%
ROC AUC	59.23%	64.06%

Upon performing the computation process of NB with FwS, an increase in accuracy of 8.32% was observed, indicating an improved model performance and more accurate predictions. While there was a slight increase of 9.14% in precision, the recall also showed an improvement of 15.41% in correctly predicting positive data. The ROC AUC exhibited an increase of 4.83%. Based on the ROC AUC parameter, the implementation of DT and BE was categorized as "good classification."

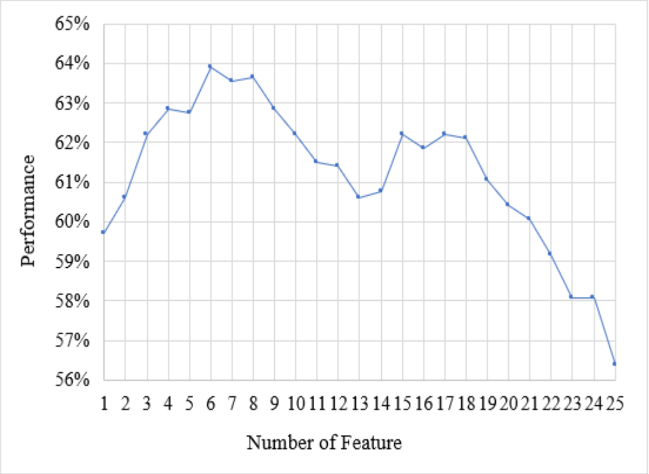

Figure 6. Feature reduction graph of NB & FS

The implementation of NB and FwS produces an accuracy of 63.9%, contributing to the elimination of features in the model up to 6 features. The success of FwS in reducing irrelevant features results in better performance. The following is the result of implementing feature reduction in Figure 6.

From Figure 6, it was concluded that the accuracy fluctuated with each iteration of the model. However, a notable decline was observed starting from the 6th iteration, indicating that additional feature reduction did not lead to an improvement in accuracy. This finding demonstrated that the implementation of FwS successfully enhanced the accuracy of NB by eliminating irrelevant attributes from the model.

4.11 Implementation of NB and BE

Furthermore, the NB algorithm was computed with the BE technique. The implementation of NB with BE was performed with the use of 10-fold cross-validation, as shown in Table 18.

Table 18. Implementation results of NB + BE

Indicator	NB	NB + BE
Accuracy	55.58%	64.01%
Precision	58.84%	67.97%
Recall	39.26%	55.02%
ROC AUC	59.23%	64.14%

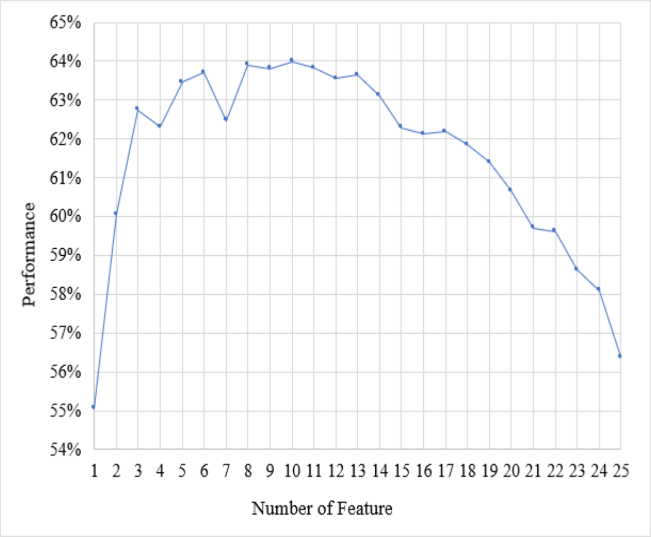

Figure 7. Reduction of NB & BE features

Upon performing the computation process for NB with BE, an increase in accuracy of 8.43% was observed, indicating an improved model performance and more accurate prediction. Precision increased by 9.13%, while recall showed an improvement of 15.76% in predicting correctly positive data. Additionally, ROC AUC experienced a 4.91% improvement. Based on the ROC AUC parameter, the implementation of NB and BE was considered a poor classification.

With an accuracy rate of 64.01%, the implementation of NB and BE contributed to the removal of up to 10 features in the model. The success of BE in reducing irrelevant features led to better performance. The findings of feature reduction implementation was shown in Figure 7.

From Figure 7, there are fluctuations in accuracy in each model iteration. Starting from the 10th iteration, a notable decline was observed, indicating that additional features did not improve accuracy. This finding demonstrated that the implementation of BE successfully enhanced the accuracy of *NB*.

4.12 Discussion

During the evaluation stage, predefined parameters such as accuracy, precision, recall, and ROC AUC were calculated using the equations in Table 7, utilizing the generated confusion matrix. The graph depicting the comparison of algorithms accuracy and FwS was shown in Table 19 and Figure 8.

Table 19. Comparison of algorithm evaluation & FS

Indicator	Accuracy	Precision	Recall	AUC
KNN+FwS	83.80%	89.70%	76.89%	83.90%
KNN+BE	80.84%	87.47%	72.66%	80.96%
DT+FwS	82.70%	89.61%	74.60%	82.84%
DT+BE	81.90%	88%	74.25%	82.03%
NB+FwS	63.90%	67.98%	54.67%	64.06%
NB+BE	64.01%	67.97%	55.02%	64.14%

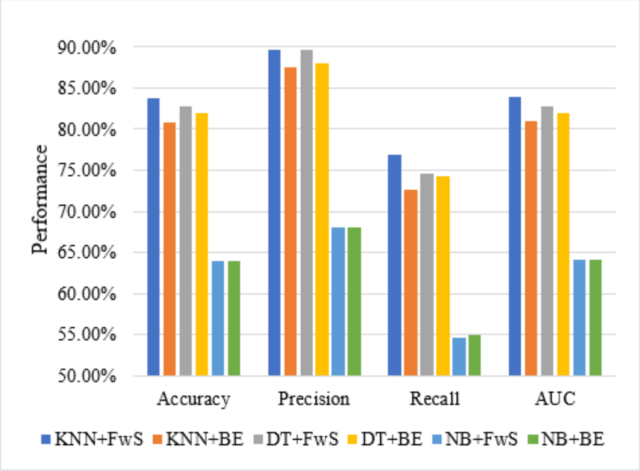

Figure 8. Comparison of algorithm evaluation & FwS

Figure 8 and Table 19 present the comparison of the evaluation of the NB, DT, and KNN algorithms using FwS and BE, concerning accuracy, precision, recall, and ROC AUC. To ensure a consistent evaluation of our classification models, we employed a standardized scale based on the area under the receiver operating characteristic curve (ROC AUC) values. This scale classifies model performance as follows: Excellent (ROC AUC > 0.90), Good (0.80 < ROC AUC ≤ 0.90), Fair (0.70 < ROC AUC ≤ 0.80), Poor (0.60 < ROC AUC ≤ 0.70), and Fail (ROC AUC ≤ 0.60).

The implementation of KNN (k=1) and FwS generated the highest accuracy of 83.8% among the considered algorithms. It was concluded that the KNN with FwS outperformed the others in terms of accuracy, precision, recall, and ROC AUC. The NB with FwS exhibited poorer performance in all these metrics compared to the other algorithms. Based on this criteria, the KNN with FwS (ROC AUC = 0.839) and DT with FwS (ROC AUC = 0.8284) both achieved 'Good' classification performance. The NB models, with ROC AUC values of 0.6406 (FwS) and 0.6414 (BE), fell into the 'Poor' category, indicating significant room for improvement in these models. The result of the comparison of algorithm accuracy and FwS is shown in Figure 9.

Figure 9 presented the comparison of the accuracy of the NB, DT, and KNN algorithms using FwS and BE. It was observed that the NB with BE achieved better accuracy compared to FwS. The DT with BE had slightly lower accuracy than FwS, and the same trend was observed for KNN with BE. Accordingly, in the case of KNN and DT algorithms, FwS demonstrated a more positive contribution compared to BE. While for the NB algorithm, BE had a more positive contribution than FwS.

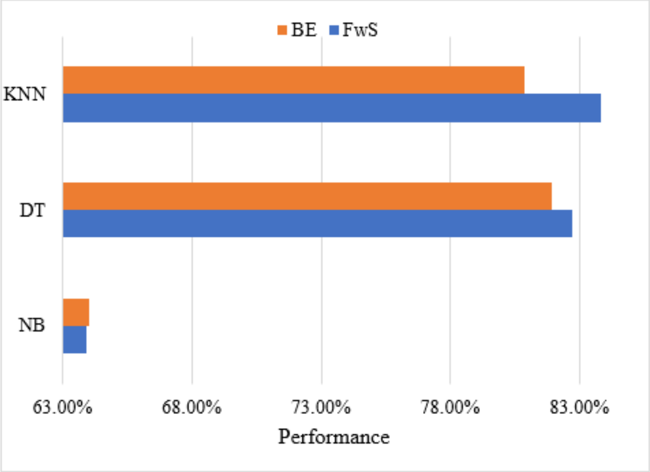

Figure 9. Comparison of algorithm accuracy & FwS

In addition to model evaluation, FwS played a crucial role in reducing irrelevant features and enhancing model performance. Relevant features indicated the model's impact on the case study and are utilized as parameters for evaluating the higher educational system. The comparison of feature reduction algorithms with FwS is presented in Figure 10.

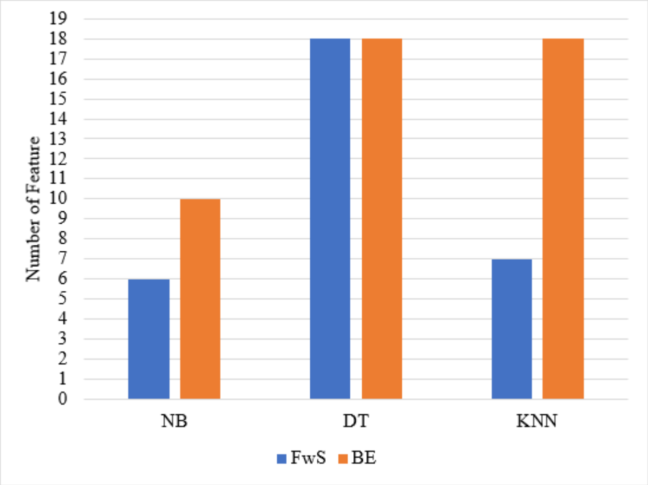

Figure 10. Comparison of feature reduction algorithm & FwS

Figure 10 shows the insights into the feature reduction achieved by BE and FwS in the KNN, DT, and NB algorithms. The implementation of BE led to fewer feature reductions compared to FwS for the NB, DT, and KNN algorithms. NB with FwS exhibited the highest feature reduction among the algorithms. However, despite the performance, it performed poorly in terms of accuracy compared to the other algorithms. Here's a detailed analysis of the impact of reducing features.

KNN with FwS. Starting with 25 features, the KNN model with FwS shows a steady increase in performance as irrelevant features are removed. Accuracy increases from 76.64% with all features to a peak of 83.8% when only seven features are retained. ROC AUC followed a similar trend, peaking at 0.839. This improvement can be attributed to the removal of distracting or irrelevant features that negatively impact the model's ability to identify true nearest neighbors. However, we observe that feature reduction goes beyond this point, resulting in performance degradation. For example, when only five features are retained, the accuracy drops to 81.2% and the ROC AUC drops to 0.815. This decrease occurs because, at this point, we start removing features that contain valuable information for classification, thereby causing underfitting.

DT with BE. The DT model initially used all 25 features, achieving 79.59% accuracy. When we applied BE, we observed that the model performance improved, reaching a maximum accuracy of 81.9% when 7 features were removed and 18 features were retained. ROC AUC also peaked at this point, with a value of 0.8203. These improvements demonstrate the DT's increased ability to make splitting decisions based on the most relevant features. However, further feature reduction leads to performance degradation. With only 10 features, the accuracy drops to 79.5%, and the ROC AUC drops to 0.798. This decrease can be explained by the loss of important decision boundaries in the tree structure as important features are removed, thereby leading to oversimplification of the model.

NB with FwS. The NB model shows a different pattern compared to KNN and DT. Starting with 55.58% accuracy in using all features, the model performance increases as features are added via FwS. Peak performance was achieved with just six features, resulting in an accuracy of 63.9% and an AUC ROC of 0.6406. This significant improvement with fewer features is in line with NB's assumption of feature independence, as performance is better if this assumption is not violated. However, adding more features beyond this point causes a slight decrease in performance, with accuracy dropping to 62.8% when eight features are used. “This decrease is most likely caused by the introduction of features that violate the independence assumption or introduce noise, thereby negatively impacting the probability estimates in the NB model.

Table 20. Best FwS results

Category	Feature	Description
Academic Data	SAI1	Semester Achievement Index 1
	SAI2	Semester Achievement Index 2
	Credit1	Achievement credits in Semester 1
	Credit 2	Achievement credits in Semester 2
	Origin University	Undergraduate Institution
Financial	Fund Source	Fund source during university
Demographics	Citizenship	Student Citizenship (Foreign /Indonesian/Indonesian Descent)

FwS played a vital role in improving the accuracy of the constructed model by selecting relevant attributes. In the context of the student graduation case study, FwS aimed to identify the attributes that influence student graduation. Based on the conducted study, the implementation of FwS and KNN (k=1) yielded the highest accuracy of 83.8%. The implementation effectively predicted student graduation and identified relevant features as shown in Table 20.

Based on the analysis presented in Table 20, the implementation of feature reduction using FwS and KNN (k=1) yielded the identification of three categories of influential data, namely academic, financial, and demographic. The academic data included variables such as the Achievement Index (AI) in semesters 1 and 2, the achievement credits in semesters 1 and 2, and the undergraduate institution. Financial data represented the students' funding source during their studies. Several demographic factors were found to influence student graduation, such as the student's citizenship status, including foreign/Indonesian/Indonesian descent.

Our study's findings both align with and diverge from previous research in student graduation prediction. Wirawan et al. [30] employed NB, DT, and KNN algorithms without FwS, reporting accuracies of 71%, 67%, and 65%, respectively. In contrast, our implementation of these algorithms with FwS techniques yielded higher accuracies: 64.01% (NB+BE), 82.7% (DT+FwS), and 83.8% (KNN+FwS).

Several factors may contribute to these performance differences. Firstly, our dataset characteristics differ. While Wirawan et al. [30] used a dataset of 3,000 student records with 20 attributes, our study employed 842 records with 25 initial attributes. The larger feature set in our study potentially provided more information for classification, which, when combined with FwS, led to improved performance.

Secondly, our preprocessing steps differed. We implemented random oversampling to address class imbalance, a step not mentioned in Wirawan et al. Study [30]. This technique likely contributed to our models' improved performance by providing a balanced representation of both classes during training.

Furthermore, our implementation of FS techniques (FwS and BE) played a crucial role in enhancing model performance. This is evident in the substantial accuracy improvements observed, particularly for the DT and KNN algorithms. For instance, our KNN model's accuracy increased from 76.64% to 83.8% after FwS, a significant improvement not seen in studies without FS.

Usman et al. [27] emphasized the importance of FS in EDM, specifically highlighting the wrapper method's effectiveness with NB. While they didn't provide specific accuracy figures, our results align with their findings on the importance of FS. However, in our study, NB showed the least improvement with FS compared to DT and KNN, suggesting that the effectiveness of FS may vary depending on the specific dataset and algorithm used.

It's worth noting that direct comparisons between studies should be made cautiously due to differences in datasets, preprocessing techniques, and implementation details. Our study's unique contribution lies in the comprehensive comparison of three algorithms (NB, DT, and KNN) with two FS techniques (FwS and BE) on a specific dataset from Sebelas Maret University. This provides insight into the interaction between algorithms and FS methods in educational contexts.

To provide statistical support for our findings, we calculated 95% confidence intervals for the accuracy of each FS algorithm combination using bootstrapping with 1000 resamples. We also conducted paired t-tests to compare the performance of different algorithms. Table 21 shows the 95% confidence intervals for accuracy.

These confidence intervals provide a range of plausible values for the true accuracy of each model in the population. The non-overlapping confidence intervals between the NB models and the KNN/DT models suggest that the performance differences are statistically significant.

To further validate these differences, we conducted paired t-tests comparing the performance of KNN+FwS (our best-performing model) against the other models. The results were presented in Table 22.

Table 21. The 95% confidence intervals for accuracy

Model	Accuracy	95% Confidence Interval
KNN+FwS	83.8%	81.2% - 86.4%
DT+FwS	82.7%	80.1% - 85.3%
KNN+BE	80.84%	78.2% - 83.5%
DT+BE	81.9%	79.3% - 84.5%
NB+FwS	63.9%	60.8% - 67.0%
NB+BE	64.01%	60.9% - 67.1%

Table 22. Paired t-tests comparing the performance of KNN+FwS

Comparison	t-statistic	df	p-value
KNN+FwS vs. DT+FwS	2.31	9	0.046
KNN+FwS vs. KNN+BE	4.87	9	< 0.001
KNN+FwS vs. DT+BE	3.15	9	0.012
KNN+FwS vs. NB+FwS	18.76	9	< 0.001
KNN+FwS vs. NB+BE	18.62	9	< 0.001

These results indicate that the performance of KNN+FwS is statistically significantly better than all other models at the 0.05 significance level. The largest performance gap is observed between KNN+FwS and the NB models, with extremely low p-values ($p < 0.001$) indicating strong evidence against the null hypothesis of equal performance.

It's important to note that while these statistical tests provide evidence of significant differences in performance, they do not account for all sources of variability in real-world applications. Factors such as changes in student populations or educational policies over time could affect the generalizability of these results. Nonetheless, these analyses provide strong statistical support for the superiority of the KNN+FwS model in our specific context of predicting student graduation at Sebelas Maret University.

4.13 Analysis of NB performance

The NB algorithm, despite its simplicity and efficiency, demonstrated lower performance compared to KNN and DT in our study. To understand this discrepancy, it's crucial to examine the underlying assumptions of NB and how they relate to our dataset.

NB is a probabilistic classifier based on Bayes' theorem with a "naïve" assumption of conditional independence between features. The algorithm calculates the probability of each class given the input features and selects the class with the highest probability. While this approach can be highly effective in certain scenarios, it relies on two key assumptions: (1) Feature Independence: NB assumes that all features are conditionally independent given the class label; and (2) Equal Feature Importance: The algorithm treats all features as equally important in making predictions.

In the context of our student graduation prediction dataset, these assumptions may not hold true, which could explain the lower performance of NB compared to KNN and DT. Firstly, the feature independence assumption is likely violated in our educational data. For instance, there may be strong correlations between academic performance indicators such as GPA, credits earned, and semester achievement indices. The NB algorithm, by treating these potentially correlated features as independent, may overemphasize their collective impact on the prediction, leading to biased probability estimates.

Secondly, the equal feature importance assumption may not be appropriate for our dataset. Some features, such as GPA or cumulative credits, may be significantly more predictive of on-time graduation than others like housing status or funding source. NB's inability to weight features based on their predictive power could result in suboptimal classification decisions.

The performance disparity between NB and the other algorithms (KNN and DT) can be attributed to their differing approaches to handling feature relationships and importance: KNN, by considering the local neighborhood of data points, implicitly accounts for feature interactions and their relative importance in that local space. This allows it to capture complex, non-linear relationships between features and the target variable. DT algorithms, through their recursive splitting process, can automatically identify and prioritize the most informative features at each node. This enables them to capture both feature importance and some degree of feature interaction.

In contrast, NB's performance in our study (accuracy of 64.01% with BE and 63.90% with FwS) suggests that its simplified model of the data does not capture the complex relationships present in student graduation patterns. The relatively small improvement from FS (from 55.58% baseline accuracy) indicates that even with optimal feature subsets, the fundamental limitations of the NB assumptions persist.

To illustrate, consider a scenario where SAI1 and SAI2 are highly correlated. NB would treat a high SAI1 and high SAI2 as two independent pieces of evidence for on-time graduation, potentially overestimating the probability. In contrast, KNN would consider these as part of the overall similarity between data points, and DT might choose only one of these features if they provide redundant information.

In conclusion, while NB offers computational efficiency and works well with high-dimensional data, its performance in our study was limited by the violation of its core assumptions. The complex, interrelated nature of factors influencing student graduation appears to be better captured by algorithms like KNN and DT, which can account for feature interactions and relative importance.

4.14 Algorithm-specific insights on FS

Our study revealed intriguing patterns in how different algorithms respond to FwS and BE methods. Notably, KNN and DT showed more significant improvements with FwS, while NB slightly favored BE. These differences can be attributed to the inherent characteristics of each algorithm and how they interact with the FS processes.

KNN benefited substantially from FwS, with accuracy improving from 76.64% to 83.80%. This can be explained by KNN's sensitivity to the 'curse of dimensionality'. As KNN operates in the feature space, having too many features can lead to sparsity in the data, making it difficult to find truly representative nearest neighbors. FwS, which incrementally adds the most informative features, helps KNN focus on the most relevant dimensions of the data space. For example, in our student graduation prediction task, FwS might first select GPA, which strongly correlates with graduation likelihood.

This focused feature set allows KNN to make decisions based on the most informative aspects of student performance, improving its predictive power. BE, while still beneficial (accuracy of 80.84%), was less effective for KNN. This could be because BE starts with all features, potentially allowing some noise or less relevant features to influence the initial neighbor calculations, which might not be entirely mitigated by subsequent feature removals.

DT also showed a preference for FwS (accuracy of 82.70% vs. 81.90% with BE). This aligns with the DT's sequential decision-making process. FwS mimics the tree-building process, where the most informative feature is selected first (corresponding to the root node), followed by progressively less influential but still relevant features. In our educational context, a DT with FS might first split on 'GPA', then on'major-specific performance', and so on. This builds a tree that reflects the hierarchical importance of features in predicting graduation, much like how academic institutions might prioritize factors in evaluating student progress. BE, while still effective, might sometimes eliminate features that could be useful in specific branches of the tree, especially for capturing minority class patterns or edge cases in student graduation scenarios.

Interestingly, NB showed a slight preference for BE (64.01% accuracy) over FwS (63.90% accuracy), though the difference is minimal. This behavior can be attributed to the NB's fundamental assumption of feature independence. BE, starting with all features, allows NB to initially consider all possible influences on student graduation. As features are removed, it helps in reducing potential violations of the independence assumption. For instance, removing highly correlated academic performance metrics might leave NB with a set of more independent features, aligning better with its core assumption. FwS, conversely, might sometimes fail to include features that, while not the most predictive individually, could contribute to a more comprehensive probability estimation when considered alongside others.

However, the minimal difference between FS and BE for NB suggests that in our specific dataset, the FS method had less impact on NB compared to KNN and DT. This could indicate that NB's performance is more constrained by its underlying assumptions than by the specific FS technique employed.

In conclusion, these algorithm-specific responses to FS methods highlight the importance of considering the interplay between algorithm characteristics and FS techniques. Our findings suggest that for complex, potentially interrelated educational data, algorithms capable of capturing feature interactions (like KNN and DT) benefit more from a carefully curated feature set built through FwS. Meanwhile, probabilistic models like NB, which have stricter underlying assumptions, may benefit slightly from the more comprehensive initial view provided by BE.

5. CONCLUSION

This study indicated that KNN with k = 1 and BE achieved the highest accuracy of 83.8%. This combination likely performed best due to KNN's ability to capture complex, non-linear relationships in the data, while BE effectively reduced noise by removing irrelevant features. The implementation of KNN and DT with FwS demonstrated a more valuable influence compared to BE, and vice versa in the case of NB.

The varying performance of algorithms with different FS techniques can be attributed to their inherent characteristics. KNN and DT benefited more from FwS, possibly due to their sensitivity to feature interactions, which FwS preserves. In contrast, NB performed better with BE, likely because it assumes feature independence, and BE helps remove redundant features that could violate this assumption.

The significant attributes identified by FwS in KNN, including student citizenship, fund source, undergraduate institution, and early semester performance indicators, align with existing educational research. For instance, early academic performance (Semester Achievement Index 1 and 2, Achievement Credits in Semesters 1 and 2) is often a strong predictor of overall academic success. The influence of factors like student citizenship and fund sources suggests that socio-economic and cultural factors play a role in timely graduation, highlighting the need for targeted support systems.

For future studies, it is recommended to create smaller class groupings in data to further improve accuracy. In the NB algorithm, the utilization of Laplace estimation/m-estimation was further developed. Exploring the combination of different classification algorithms with the studied algorithms is another avenue for potential investigation.For future studies, it is recommended to create smaller class groupings in data to further improve accuracy. In the NB algorithm, the utilization of Laplace estimation/m-estimation was further developed. Exploring the combination of different classification algorithms with the studied algorithms is another avenue for potential investigation.

Additionally future research could explore variants of NB, such as Gaussian NB or Multinomial NB, or consider ensemble methods that might mitigate some of these limitations while retaining the computational benefits of NB.

ACKNOWLEDGMENT

We would like to thank Sebelas Maret University for providing the Research Group Research Grant (Grant-MRG), with Contract Number: 228/UN27.22/PT.01.03/2023, so that this research can be carried out.

REFERENCES

[1] Millea, M., Wills, R., Elder, A., Molina, D. (2018). What matters in college student success? Determinants of college retention and graduation rates. Education, 138(4): 309-322.

[2] Barra, C., Zotti, R. (2016). Measuring efficiency in higher education: An empirical study using a bootstrapped data envelopment analysis. International Advances in Economic Research, 22: 11-33. https://doi.org/10.1007/s11294-015-9558-4

[3] Li, A.Y., Kennedy, A.I. (2018). Performance funding policy effects on community college outcomes: Are short-term certificates on the rise? Community College Review, 46(1): 3-39. https://doi.org/10.1177/0091552117743790

[4] Mabel, Z., Britton, T.A. (2018). Leaving late: Understanding the extent and predictors of college late departure. Social Science Research, 69: 34-51. https://doi.org/10.1016/j.ssresearch.2017.10.001

[5] Mikhaylov, A.S., Mikhaylova, A.A. (2018). University

rankings in the quality assessment of higher education institutions. Calitatea, 19(163): 111-117.

[6] Asif, R., Merceron, A., Ali, S.A., Haider, N.G. (2017). Analyzing undergraduate students' performance using educational data mining. Computers & Education, 113: 177-194. https://doi.org/10.1016/j.compedu.2017.05.007

[7] Baker, R.S., Yacef, K. (2009). The state of educational data mining in 2009: A review and future visions. Journal of Educational Data Mining, 1(1): 3-17. https://doi.org/10.5281/zenodo.3554657

[8] Das, D., Shakir, A.K., Rabbani, M.S.G., Rahman, M., Shaharum, S.M., Khatun, M.S., Fadilah, N.B., Qaiduzzaman, K.M., Islam, S., Arman, M.S. (2020). A comparative analysis of four classification algorithms for university students performance detection. In ECCE2019: Proceedings of the 5th International Conference on Electrical, Control & Computer Engineering, Kuantan, Pahang, Malaysia, pp. 415-424. https://doi.org/10.1007/978-981-15-2317-5_35

[9] Dutt, A., Ismail, M.A., Herawan, T. (2017). A systematic review on educational data mining. IEEE Access, 5: 15991-16005. https://doi.org/10.1109/ACCESS.2017.2654247

[10] Almarabeh, H. (2017). Analysis of students' performance by using different data mining classifiers. International Journal of Modern Education and Computer Science, 9(8): 9-15. https://doi.org/10.5815/ijmecs.2017.08.02

[11] Witten, H.I., Frank, E., Hall, M.A., Pal, C.J. (2017). Data mining practical machine learning tools and techniques. http://books.google.com/books?id=bDtLM8CODsQC&pgis=1

[12] Asif, R., Merceron, A., Ali, S.A., Haider, N.G. (2017). Analyzing undergraduate students' performance using educational data mining. Computers & Education, 113: 177-194. https://doi.org/10.1016/j.compedu.2017.05.007

[13] Pujianto, U., Qomaria, U. (2020). Predicting high school graduates using naive Bayes in state university entrance selections. In 2020 4th International Conference on Vocational Education and Training (ICOVET), pp. 155-159. https://doi.org/10.1109/ICOVET50258.2020.9230336

[14] Arifin, D., Hadiana, A. (2019). Computer-based techniques for predicting the failure of student studies using the decision tree method. In IOP Conference Series: Materials Science and Engineering, 662(2): 022112. https://doi.org/10.1088/1757-899X/662/2/022112

[15] Mawardi, V.C., Santun, N.D. (2020). Website based application of doctor selection classification derive from patient complaints using the C4. 5 method and k-Nearest neighbor. In IOP Conference Series: Materials Science and Engineering, 1007(1): 012134. https://doi.org/10.1088/1757-899X/1007/1/012134

[16] Tu, J.V. (1996). Advantages and disadvantages of using artificial neural networks versus logistic regression for predicting medical outcomes. Journal of Clinical Epidemiology, 49(11): 1225-1231. https://doi.org/10.1016/S0895-4356(96)00002-9

[17] Chui, K.T., Fung, D.C.L., Lytras, M.D., Lam, T.M. (2020). Predicting at-risk university students in a virtual learning environment via a machine learning algorithm. Computers in Human Behavior, 107: 105584. https://doi.org/10.1016/j.chb.2018.06.032

[18] Alyahyan, E., Düştegör, D. (2020). Predicting academic success in higher education: Literature review and best practices. International Journal of Educational Technology in Higher Education, 17(1): 3. https://doi.org/10.1186/s41239-020-0177-7

[19] Namoun, A., Alshanqiti, A. (2020). Predicting student performance using data mining and learning analytics techniques: A systematic literature review. Applied Sciences, 11(1): 237. https://doi.org/10.3390/app11010237

[20] Hellas, A., Ihantola, P., Petersen, A., Ajanovski, V.V., Gutica, M., Hynninen, T., Knutas, A., Leinonen, J., Messom, C., Liao, S.N. (2018). Predicting academic performance: A systematic literature review. In Proceedings companion of the 23rd Annual ACM Conference On Innovation and Technology in Computer Science Education, pp. 175-199. https://doi.org/10.1145/3293881.3295783

[21] Saha, P., Patikar, S., Neogy, S. (2020). A correlation-sequential forward selection based feature selection method for healthcare data analysis. In 2020 IEEE International Conference on Computing, Power and Communication Technologies (GUCON), Greater Noida, India, pp. 69-72. https://doi.org/10.1109/GUCON48875.2020.9231205

[22] Masoudi-Sobhanzadeh, Y., Motieghader, H., Masoudi-Nejad, A. (2019). FeatureSelect: A software for feature selection based on machine learning approaches. BMC Bioinformatics, 20: 1-17. https://doi.org/10.1186/s12859-019-2754-0

[23] Venkatesh, B., Anuradha, J. (2019). A review of feature selection and its methods. Cybernetics and Information Technologies, 19(1): 3-26. https://doi.org/10.2478/CAIT-2019-0001

[24] Wright, E., Hao, Q., Rasheed, K., Liu, Y. (2018). Feature selection of post-graduation income of college students in the United States. In Social, Cultural, and Behavioral Modeling: 11th International Conference, SBP-BRiMS 2018, Washington, DC, USA, pp. 38-45. https://doi.org/10.1007/978-3-319-93372-6_4

[25] Gotardo, M.A. (2019). Using decision tree algorithm to predict student performance. Indian Journal of Science and Technology, 12(8): 1-8. https://doi.org/10.17485/ijst/2019/v12i5/140987

[26] Wah, Y.B., Ibrahim, N., Hamid, H.A., Abdul-Rahman, S., Fong, S. (2018). Feature selection methods: Case of filter and wrapper approaches for maximising classification accuracy. Pertanika Journal of Science & Technology, 26(1): 329-340.

[27] Usman, M.M., Owolabi, O., Ajibola, A.A. (2020). Feature selection: It importance in performance prediction. IJESC, 10(5): 25625-25632.

[28] Li, C., Hains, M., Wallin, J., Wu, Q. (2019). Applying data science methods for early prediction of undergraduate student retention. In 2019 International Conference on Computational Science and Computational Intelligence (CSCI), Las Vegas, NV, USA, pp. 1337-1340. https://doi.org/10.1109/CSCI49370.2019.00250

[29] Gunawan, Hanes, Catherine. (2019). Information systems students' study performance prediction using data mining approach. In 2019 Fourth International Conference on Informatics and Computing (ICIC), Semarang, Indonesia, pp. 1-8.

https://doi.org/10.1109/ICIC47613.2019.8985718

[30] Wirawan, C., Khudzaeva, E., Hasibuan, T.H., Lubis, Y.H.K. (2019). Application of data mining to prediction of timeliness graduation of students (a case study). In 2019 7th International Conference on Cyber and IT Service Management (CITSM), Jakarta, Indonesia, pp. 1-4. https://doi.org/10.1109/CITSM47753.2019.8965425

[31] Salim, A.P., Laksitowening, K.A., Asror, I. (2020). Time series prediction on college graduation using kNN algorithm. In 2020 8th International Conference on Information and Communication Technology (ICoICT), Yogyakarta, Indonesia, pp. 1-4. https://doi.org/10.1109/ICoICT49345.2020.9166238

[32] Wiyono, S., Abidin, T., Wibowo, D.S., Hidayatullah, M. F., Dairoh, D. (2019). Comparative study of machine learning knn, svm, and decision tree algorithm to predict students performance. International Journal of Research-Granthaalayah, 7(1): 190-196. https://doi.org/10.29121/granthaalayah.v7.i1.2019.1048

[33] Saifudin, A., Desyani, T. (2020). Forward selection technique to choose the best features in prediction of student academic performance based on Naïve Bayes. In Journal of Physics: Conference Series, 1477(3): 032007. https://doi.org/10.1088/1742-6596/1477/3/032007

[34] Maulana, A. (2021). Prediction of student graduation accuracy using decision tree with application of genetic algorithms. In IOP Conference Series: Materials Science and Engineering, 1073(1): 012055. https://doi.org/10.1088/1757-899X/1073/1/012055

[35] England, B.J., Brigati, J.R., Schussler, E.E., Chen, M.M. (2019). Student anxiety and perception of difficulty impact performance and persistence in introductory biology courses. CBE-Life Sciences Education, 18(2): ar21. https://doi.org/10.1187/cbe.17-12-0284

[36] Bode, A., Lamasigi, Z.Y., Drajana, I.C.R. (2023). The K-nearest neighbor algorithm using forward selection and backward elimination in predicting the student's satisfaction level of university ichsan gorontalo toward online lectures during the COVID-19 pandemic. ILKOM Jurnal Ilmiah, 15(1): 118-123. https://doi.org/10.33096/ilkom.v15i1.1381.118-123

[37] Thangavel, S.K., Bkaratki, P.D., Sankar, A. (2017). Student placement analyzer: A recommendation system using machine learning. In 2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, pp. 1-5. https://doi.org/10.1109/ICACCS.2017.8014632

[38] Li, J., Cheng, K., Wang, S., Morstatter, F., Trevino, R.P., Tang, J., Liu, H. (2017). Feature selection: A data perspective. ACM computing surveys (CSUR), 50(6): 1-45. https://doi.org/10.1145/3136625

[39] Jain, D., Singh, V. (2018). Feature selection and classification systems for chronic disease prediction: A review. Egyptian Informatics Journal, 19(3): 179-189. https://doi.org/10.1016/j.eij.2018.03.002

[40] Xue, B., Zhang, M., Browne, W.N., Yao, X. (2015). A survey on evolutionary computation approaches to feature selection. IEEE Transactions on Evolutionary Computation, 20(4): 606-626. https://doi.org/10.1109/TEVC.2015.2504420

[41] Chen, Y., Johri, A., Rangwala, H. (2018). Running out of stem: A comparative study across stem majors of college students at-risk of dropping out early. In Proceedings of the 8th international conference on learning analytics and knowledge, pp. 270-279. https://doi.org/10.1145/3170358.3170410

[42] Adekitan, A.I., Salau, O. (2019). The impact of engineering students' performance in the first three years on their graduation result using educational data mining. Heliyon, 5(2): e01250. https://doi.org/10.1016/j.heliyon.2019.e01250

[43] Aulck, L., Velagapudi, N., Blumenstock, J., West, J. (2016). Predicting student dropout in higher education. arXiv preprint arXiv:1606.06364. http://arxiv.org/abs/1606.06364

[44] Hutt, S., Gardener, M., Kamentz, D., Duckworth, A. L., D'Mello, S.K. (2018). Prospectively predicting 4-year college graduation from student applications. In Proceedings of the 8th international conference on learning analytics and knowledge, New York, United States, pp. 280-289. https://doi.org/10.1145/3170358.3170395

[45] Salazar, A., Vergara, L., Safont, G. (2021). Generative adversarial networks and markov random fields for oversampling very small training sets. Expert Systems with Applications, 163: 113819. https://doi.org/10.1016/j.eswa.2020.113819

[46] Mohammed, R., Rawashdeh, J., Abdullah, M. (2020). Machine learning with oversampling and undersampling techniques: overview study and experimental results. In 2020 11th international conference on information and communication systems (ICICS), Irbid, Jorda, pp. 243-248. https://doi.org/10.1109/ICICS49469.2020.239556

[47] Hasan, M., Islam, M.M., Zarif, M.I.I., Hashem, M.M.A. (2019). Attack and anomaly detection in IoT sensors in IoT sites using machine learning approaches. Internet of Things, 7: 100059. https://doi.org/10.1016/j.iot.2019.100059

[48] Portier, W.K., Li, Y., Kouassi, B.A. (2020). Feature selection and classification methods for predicting search engine ranking. In Proceedings of the 2020 3rd International Conference on Signal Processing and Machine Learning, New York, United States, pp. 84-90. https://doi.org/10.1145/3432291.3432309