Academic Performance Warning System for Higher Education

Hanh Thi-Hong Duong[1,2], Linh Thi-My Tran[1,2], Huy Quoc To[1,2] and Kiet Van Nguyen[1,2*]

[1*]University of Information Technology, Ho Chi Minh City, Vietnam.
[2*]Vietnam National University, Ho Chi Minh City, Vietnam.

*Corresponding author(s). E-mail(s): kietnv@uit.edu.vn;
Contributing authors: 18520711@gm.uit.edu.vn; 18520999@gm.uit.edu.vn; huytq@uit.edu.vn;

Abstract

Academic probation at universities has become a growing source of concern in recent years, as an increasing number of students find themselves in this circumstance and face serious consequences. Confronted with the situation mentioned above, we use the power of massive data sources from the education sector, as well as the modernity of Machine Learning techniques, to undertake research on this problem in Vietnam. In this paper, the issue of academic warning is initially treated from the standpoint of academic performance—one of the main criteria that directly affects students' academic probation status at the university. Through the research process, we provide two really valuable datasets that have been painstakingly extracted and developed from raw data sources, including a wealth of information about students, subjects, scores, and so on. A two-stage academic performance warning system for smart universities is suggested with the F2-score measure of over 81% for the beginning of the semester and over 94% for before the final exam. That is the result of our extensive research and analysis with various combinations of feature generation techniques, feature selection strategies, unbalanced data handling techniques, model selection techniques, and Machine Learning algorithms. Through this research, many new things were discovered and allowed us to discuss more deeply potential future avenues of academic warning in Vietnam.

Keywords: Academic Performance Warning System, Academic Probation, Machine Learning, Higher Education

1 Introduction

In the face of an ever-increasing influx of big data, the remarkable advancement of techniques from machine learning, artificial intelligence, neural networks, database systems, and so on has enabled people to extract a wealth of valuable values from data, resulting in a more modern and all-encompassing way of life in all aspects. In this paper, we are particularly interested in opportunities and social values that can be derived from data in the field of education, particularly higher education.

Today, large amounts of student data are now stored in educational databases, giving universities a solid basis for promoting and changing development. With a competitive operating environment, a data-driven approach is widely adopted in universities around the world, which helps them to understand their students better and bring support on many fronts. However, this problem is not common in Vietnam. Based on that, we conducted this research with the goal of being able to take advantage of data sources from education to solve a persistent problem in Vietnamese universities: academic probation.

Academic probation is the most prevalent term used in the undergraduate context to signify that a student is not making the academic progress required by the institution for graduation. And the review of this matter is a periodic activity specified in the academic regulations to ensure the quality of teaching and learning in each university. Each student's academic probation status will be determined at the end of each semester based on a number of disciplinary and academic factors, with academic performance being one of the most important.

In recent years, the academic probation of students has been becoming alarmingly common in Vietnam. According to university statistics, the number of students who are placed on academic probation reaches hundreds or even thousands of students each year. And this greatly affects the quality of training, the school's output standards, as well as the academic activities of students. In particular, it can have serious academic consequences for students, such as limiting the number of credits enrolled, losing the opportunity to pursue their preferred major, or even being dismissed.

On the basis of the above, we decided to investigate and research this topic in order to assist universities in predicting the student's academic probation status, initially based on their academic performance. This research was developed with the intention of providing students with sound and timely warnings about their academic probation possibility in the new semester. Receiving warnings will make students more concerned about their current learning level so that they can adjust their learning attitudes more appropriately, strive to improve their grades, and avoid academic probation. On the other hand, based on that, the academic advisor also gains a partial understanding of the student's academic probation risk, allowing them to provide more appropriate and timely assistance to students.

In this paper, we conduct research with data provided by a university in Vietnam with policies on academic handling specified in the academic regulations of this university. Specifically, academic probation in Vietnam is divided into 3 main types of status: academic suspension, academic warning, and dismissal. However, with the goal of building a warning system based only on the academic performance of students, the models we trained will only support the classification of two states, including Academic Warning and Dismissal, because Academic Suspension is not affected by academic performance criteria. Rules related to academic performance-based warning are detailed in section 3.1.2.

Our entire research process will be detailed in the sections below. This section focuses on introducing the topic's purpose. Section 2 contains information about related research. In section 3, we show how to create datasets and information about the datasets used, and in section 4, we present a detailed description of the tasks posed to solve our problem. The following section of the paper describes our approaches to the problem. In succession, the details of our experiments, results, and evaluation will be presented in section 6. Finally, section 7 contains the conclusions and future development directions that we discovered during our research on this topic.

2 Related Works

Academic warnings for students enable the university to respond to students' learning problems as soon as possible. Advisors are able to adopt various guiding measures to assist students in improving their academic performance or prevent the delayed graduation of students. As a result, more and more academics are recognizing the immense societal potential of educational data and conducting research in this area. Firstly, it is hard not to mention Huang and Fang et al.[1] with the research work in 2013, which was the first study in this field. Their research uses four different types of mathematical models to predict student academic achievement in the engineering dynamics course based on features such as cumulative GPA, results from three dynamics mid-term tests, and four prerequisite courses (statics, calculus I, calculus II, and physics). With over 2907 data points collected from 323 undergraduates over four semesters, this study resulted in the development of 24 predictive mathematical models, which led to multiple novel and significant findings. In addition, in 2018, Migueis et al.[2] collected data from 2459 students to build models to predict student overall academic performance. The research was conducted with Machine Learning algorithms such as Random Forest, Decision Trees, Support Vector Machines, and Naive Bayes. Finally, the model proposed by the author is Random Forest, with an accuracy above 95% at an early stage of the student's academic path. Both of the preceding studies demonstrate that previous semester grades are heavily weighted while building the model. In 2021, Zhai Mingyu et al.[3] used groupings of attributes concerning students' studies, living, internet activities, and basic information to solve the above problem. Prediction models are built using various machine learning algorithms such as Logistic regression, Decision Tree, Support Vector Machine, Random Forest, Gradient Boosting Decision Tree, XGBoost, LightGBM, and others and achieve the best results with the Catboost–SHAP method, obtaining MSE, MAE, and R2 of 24.976, 3.551, and 80.3%, respectively, in tenfold cross-validation. The findings of the study not only provide a method for detecting problematic students with poor expected grades but also for analyzing specific factors' impacts on students' learning outcomes. In addition to experimenting with various methods for predicting final students' grades in first-semester courses, Bujang et al.[4] resolve imbalanced data challenges to better-improved performance. They used the Synthetic Minority Oversampling Technique (SMOTE) oversampling method to solve this problem. The results were surprising, with Random Forest producing the highest F-measure of 99.5%. In general, the research has demonstrated the possibility of using classification algorithms to resolve the academic warning problem, with Random Forest and Support Vector Machine demonstrating its advantages in this domain.

With the studies mentioned above, it is readily apparent the research of study grades and warnings about students' academic results are not uncommon in the world. However, according to our knowledge, there have been no published articles about student academic warnings in Vietnam.

3 Dataset

3.1 Data Creation

In this paper, we look at how to take advantage of students' learning scores while they're at university. This is the most intuitive and clear reflection of the student's learning situation, and it is also a key factor in the school's academic warning decision-making process. However, because the source of the raw data about student scores is so large and varied, we must take steps to process, synthesize, analyze, and filter out the attributes that our system requires. In addition, we organize the data and create models in order to cover as much of the current education system in Vietnamese universities as possible. Usually, a learning year is divided into two semesters, each of which includes two important exams: a midterm exam and a final exam. Students register for the semester's subjects, which may include more than one before the semester begins. Following the completion of each subject, there will be a corresponding GPA; the semester's GPA will be the average GPA of all the subjects students complete in that semester. Each subject is assigned a certain number of credits. Furthermore, the learning grade is divided into four types, and we call them component grades. There are four types of grades: process grade (student's attendance), midterm grade (midterm exam), practice grade (practice exam), and final grade (final exam).

Each grade will be assigned a weight, and the sum of the weights of the four types of grades is equal to 1. The GPA for each subject is computed as the total of the component grades multiplied by their weights. In particular, the GPA in this paper is GPA out of 10, specifically as follows: 9.0 – 10.0 (A+), 8.0 – 9.0 (A), 7.0 – 8.0 (B+), 6.0 – 7.0 (B), 5.0 – 6.0 (C), 4.0 – 5.0 (D+), 3.0 – 4.0 (D), <3 (F). And component grades also use this scale. It is possible to view the enclosed file to see how to convert a score out of 10 to another score system: Score system.

Because data is so important in the development of our systems, we must ensure data quality while also adhering to academic warning regulations, we create the dataset through 3 stages: Calculating the required attributes, Data annotation, Evaluating and reviewing the dataset.

3.1.1 Calculating the required attributes

Base on the raw database, including the subject's grade for each student and a list of previous subjects, we process and get the required attributes according to the following formula:

Feature pre_avg: The average grade of all subjects is considered as the previous subject of the subject which student enrolled in the current semester.

Group 1: GPA of previous semesters (s1, s2, s3,..).

A student in the p-th semester will have $p - 1$ GPA attributes. The name format for these attributes is s(m) for $m \in \{x \in \mathbb{N};\ 1 \leq x \leq p - 1\}$ with the index m representing the semester student's m-th. For example, a sixth-semester student has five GPA attributes: s1, s2, s3, s4 and s5.

$$s(m) = \frac{\sum_1^n score_{m,i}.credit_{m,i}}{\sum_1^n credit_{m,i}} \tag{1}$$

Group 2: The average grade in the i-th known component grade of all subjects in the current semester (avg1, avg2, avg3).

$$avg(j) = \frac{\sum_1^n score(j)_{p,i}.coef(j)_{p,i}.credit_{p,i}}{\sum_1^n credit_{p,i}} \tag{2}$$

Group 3: The corresponding weigth to group 2 (coef1, coef2, coef3).

$$coef(j) = \frac{\sum_1^n coef(j)_{p,i}.credit_{p,i}}{\sum_1^n credit_{p,i}} \tag{3}$$

Where:

- p : is representative of the student's current semester (p-th semester).
- m : is representative of the m-th semester studied previously, 0<m<p.
- n : is the number of subjects relevant to the semester under consideration.
- $score_{m,i}$: is the GPA of subject i in the m-th semester.
- $credit_{m,i}$: is the number of credits of subject i in semester m-th.
- $score(j)_{p,i}$: is the j-th component grade for subject i in current semester.

(j=1: process grade, j=2: practice grade, j=3: midterm grade).
- $credit_{p,i}$: is the number of credits of subject i in current semester.
- $coef(j)_{p,i}$: is the corresponding weigth to $score(j)_{p,i}$.

3.1.2 Data annotation, evaluating and reviewing the dataset

We choose the conditions that depend on academic performance specified in the policy to implement labeling with the goal of building a predictive model based on academic performance. After that, we review the data and remove the null values. In addition, because the number of semesters that students study is different, we divided the datasets into different datasets (removing those that were too small, less than 200 samples) and built the corresponding model on each dataset. Table 1 shows the labeling information, whereas section 3.2 shows the dataset specifics.

Table 1: Academic probation rules base on academic performance.

Condition	**Alert status**	**Label**
GPA in 2 continuous semesters <4.0 or GPA in the considered semester <3.0	warning	1
GPA in 2 continuous semesters = 0.0	dismissal	2
Others	normal	0

3.2 Data Information

Following the data creation process described in section 3.1, we obtain two datasets: Dataset 1 and Dataset 2. Datasets 1 and 2 are made up of 4383 students' information. However, given the list of semester scores for each student, we collect the data for each small dataset according to the approach shown in Fig 1 in order to get more data for the training process and improve the model performance.

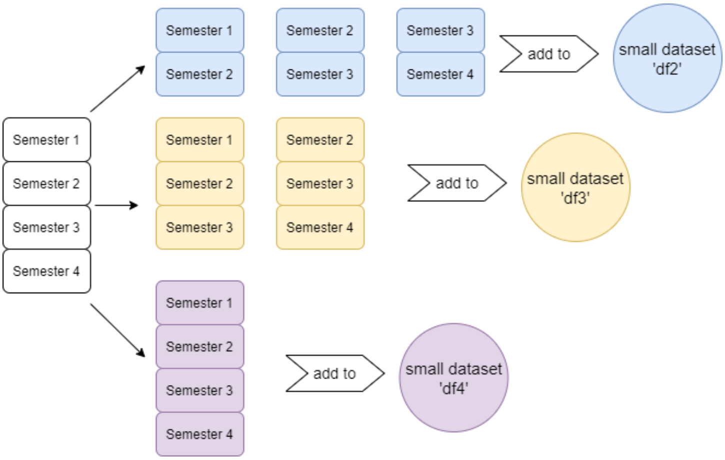

Fig. 1: Example for data collection of individual students.

Dataset 1 was created to use for building warning models at the beginning of the semester and before the final exam. It consists of 9 individual tiny dataset named in the form df$i$ for $i \in \{x \in \mathbb{N};\ 2 \leq x \leq 10\}$. The index i is representative of the number of semesters used to compute and construct each dataset. Specifically, with df$i$, the avg$m$ and coef$m$ attributes with $m \in \{1, 2, 3\}$ will be aggregated based on known information of semester i, and this dataset will have i-1 sj attributes for $j \in \{x \in \mathbb{N};\ 1 \leq x \leq i-1\}$. Details are shown in the table 2. For more information: Dataset 1.

Table 2: Statistics on the dataset 1.

Dataset	Label 0	Label 1	Label 2
df2	29,551	2,479	26
df3	23,053	1,847	25
df4	17,975	1,491	21
df5	13,130	1,221	17
df6	9,630	945	14
df7	6,268	696	13
df8	3,835	528	10
df9	1,524	330	7
df10	422	199	6

Dataset 2 is recreated to improve the beginning of semester model results and used in Version 2. Similar to dataset 1, Dataset 2 consists of 8 small datasets. Details are shown in the table 3. For more information: Dataset 2.

Table 3: Statistics on the dataset 2.

Dataset	Label 0	Label 1
df2	24,901	1,927
df3	19,165	1,389
df4	15,272	1,168
df5	10,934	946
df6	7,842	744
df7	4,904	516
df8	2,779	389
df9	998	223

4 Task Definition

With the purpose of building an academic warning system that would assist students and universities in accurately capturing the status of their studies in a timely way, without any risk of being missed, thereby minimizing the number of students receiving academic warnings. In this paper, our goal is to issue warnings to students twice a semester:

The first warning: the alert is issued at the beginning of the semester.

The second warning: the alert is issued before the final exam.

The first warning helps students understand their current academic status early on, allowing them to adjust their learning process accordingly and avoid academic probation. At the same time, the university and advisor have a partial understanding of the state of the students' academic processing abilities, allowing them to take more appropriate and timely corrective action. However, if the system only issues a single warning at the beginning of the semester, it will ignore students who have previously qualified while neglecting to study for the current semester. As a result, the second warning will remind them to maintain a positive learning attitude and to practice for the final exam in order to improve their scores.

4.1 Task 1: The first warning

Input: List GPA of previous semesters.

Output: Alert status (normal, warning, dismissal).

For a better understanding, look in table 4.

4.2 Task 2: The second warning

Input: List GPA of previous semesters, list grade and weight of the component examination in the current semester.

Output: Alert status (normal, warning, dismissal).

The things mentioned above are detailed visualized in table 5.

Table 4: Several examples of attributes used in task 1 for the first alerting.

s1	**s2**	**s3**	**s4**	**s5**	**s6**	**status**
7.54	8.11	7.55	7.91	8.64	8.62	normal
3.56	0.55	3.83				warning
5.92	5.95	4.71	2.10	0.00		dismissal

Table 5: Several examples of attributes used for the second alerting.

s1	**s2**	**s3**	**avg1**	**avg2**	**avg3**	**coef1**	**coef2**	**coef3**	**status**
7.54	8.11		0.99	0.75	1.71	0.11	0.09	0.25	normal
5.02			0.29	0.16	0.74	0.05	0.16	0.28	warning
5.35	1.10	0.00	0.00	0.00	0.00	0.01	0.00	0.17	dismissal

4.3 Task 3: The first warning with a improved strategy

After experiencing and evaluating the results of Task 1 in the first problem, we observed that some irrational areas still negatively affect the model's performance. This is why we have made a few adjustments in Task 3.

Specifically, with the first warning, we no longer construct the problem based on the output as three labels. We propose instead to construct a binary classification problem with two output labels: normal and academic probation. In this case, the academic probation represents the alert statuses of warning and dismissal.

We are motivated to make this change for two reasons. First, based on the academic probation conditions, students who are placed on dismissal also belong on academic warning (see information in section 3.1.2 for a deeper understanding). Therefore, these students should receive both types of warnings. Second, dismissal status has a specific identifier that is rarely given by the warning status: the most recent semester's GPA is zero. This facilitates their identification without the application of a classification model.

Students who are classified as academic probation class and have had the most recent semester's GPA equal to 0 (this is a condition for determining students to be forced out of school, see 3.1.2 for details) will receive both warnings, including a warning and dismissal. The remaining cases will receive the appropriate warning based on their classification. In addition, we also add a new attribute to improve model performance: pre_avg (see part 3.1.1 for more information). Details of the new problem are as follows:

Input: List GPA of previous semesters and pre_avg.

Output: Alert status (normal, academic probation).

Table 6 shows a few examples of our change in input and output in these new tasks.

Table 6: Several examples of attributes used in task 3 for the first alerting.

s1	s2	s3	pre_avg	status
7.54	8.11		8.90	normal
2.02				academic probation
5.35	0.25	0.00	1.50	academic probation

5 Our Approach

We focus on using the greatest potential and characteristics of the data, such as applying feature creation techniques or dividing the model into smaller models, to overcome the problem of data imbalance, in order to construct an effective warning system. In addition, we test a variety of machine learning algorithms to determine which is the most effective for the system.

5.1 Data Preprocessing

We do data analysis and apply appropriate strategies to improve the performance of models in this part. To begin, we construct feature 'avg10' in the second warning with dataset 1, which is the converted component average grade calculated using group features 2 (avg1, avg2, avg3) and group features 3 (coef1, coef2, coef3). We also construct the feature 'history' in the first warning with dataset 2, which is the number of times the student has been placed on academic probation in the past.

5.2 Handling Imbalanced Data

Most machine learning algorithms work best when the number of samples in each class is about equal. This is because most algorithms are designed to maximize accuracy and reduce errors. When using imbalanced data, the model is more biased to the dominant target class and tends to predict the target as the predominant target class. To deal with this issue, we apply two techniques to handle the imbalanced data: Adjusting class weights and Resampling.

We adjust the class weights by assigning the value 'balanced' to the class-weight parameter when creating an instance of the algorithm. Class-weight is a built-in parameter that assists in optimizing scoring for the minority class. When the class_weights='balanced', the model automatically assigns the class weights inversely proportional to their respective frequencies.

With the resampling method, we choose the RandomOverSampler to solve this problem. This is an over-sampling technique that increases the number of instances in the minority class by randomly replicating them in order to present a higher representation of the minority class in the sample.

Experiments are carried out with both suggested methods to choose the method that gives the best performance for each us small dataset.

5.3 New Approach - Divided Approach

In addition, to approach the form of a standard 3-label problem, we also divide it into two small binary classification problems, then we will combine two models to get the final result (fig 2).

Model 1: classifying normal students (label 0) with students who are liable to place on academic probation (label 1 and 2).

Model 2: separate students who are likely to be placed on academic probation (on model 1) into two groups: academic warning students (label 1) and dismissal students (label 2).

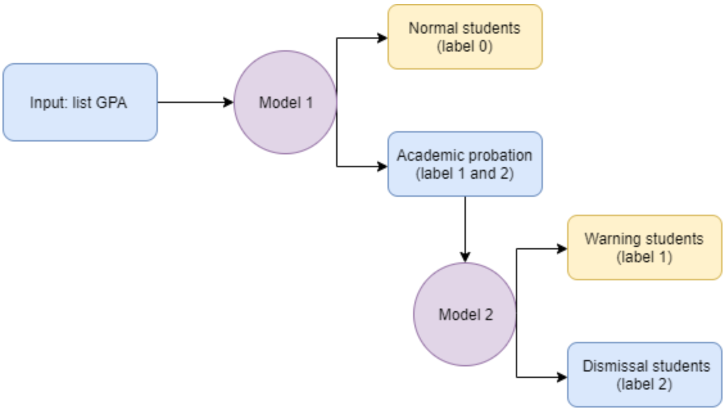

Fig. 2: Divided approach.

5.4 Algorithms

We experiment with a variety of machine-learning algorithms, both traditional and modern, and tweak the parameters to get the best model for each problem. The algorithms we use include: Decision Tree, Random Forest, Extra Trees, Gradient Boosting Decision Tree, LightGBM, Support Vector Machine, Logistic Regression, and Stochastic Gradient Descent Classifier.

Decision Tree (DT)[5] is one of the most popular machine learning algorithms today. It is used in both classification and regression problems. A decision tree is a tree where each node represents a feature, each branch represents a rule, and each leaf represents a result (specific value or a continuation branch). The decision tree is built based on many different types of algorithms such as ID3, C4.5, CART, CHAID, and others.

Random Forest (RF)[6] works based on ensemble many decision trees, but each decision tree is unique (with random factor). It allows for the resolution of overfitting and data interference issues. Each decision tree gives a classification to each object; the class with the most votes is the Random Forest's final class.

Extra Trees (ET)[7] is similar to Random Forest, except instead of bootstrapping observations, nodes are split depending on random splits among a random subset of the features chosen at each node.

Gradient Boosting Decision Tree (GB)[8] begins by constructing a tree to attempt to fit the data, and subsequent trees are constructed with the goal of minimizing residuals using a loss function influenced by the Gradient Descent. It accomplishes this by concentrating on areas where existing learners are underperforming.

LightGBM (LGBM)[9] implemented based on the Gradient Boosting Decision Tree; however, it uses GOSS (Gradient-based One-Side Sampling) and EFB (Exclusive Feature Bundling) to significantly speed up the computing process.

Support Vector Machine (SVM)[10] can be used for classification or regression It is, however, mostly used for classification. The algorithm's fundamental idea is to execute a hyperplane discovery that splits the classes based on margin. The margin is the distance between the hyperplane and the two closest data points that correspond to the classifiers. SVM always attempts to maximize this margin, resulting in the best hyperplane.

Logistic Regression (LR) is one of the most commonly used machine learning algorithms for binary classification problems. It uses the logistic sigmoid function to generate probabilistic estimates.

Stochastic Gradient Descent Classifier (SGD)[11] can be seen as a linear classifier (in this paper, we use Logistic Regression) optimized by Stochastic Gradient Descent.

5.5 Performance evaluation

5.5.1 Measure

In the evaluation task, we use measurements like the F2-score, the F1-score, and the recall of label 2. The F2 score and the recall of label 2 are two important measures.

With the goal of considering it much worse to miss a student who is at risk of academic probation (academic warning and dismissal) than to give a false alarm to a normal student, we choose the macro-average F2-Score measure for our model evaluation.

The F-score is a way of combining the precision and recall of the model and is commonly used for evaluating many kinds of machine learning models. This measure is particularly effective when applied to data that is unevenly distributed or where the costs of false positives and false negatives differ, as in the case of predicting disease in medicine. And the F-score is mathematically defined as follows:

$$F_\beta = (1+\beta^2)\frac{recall.precision}{recall+\beta^2.precision} \tag{4}$$

In (4), the recall and precision are calculated according to the formula below:

$$recall = \frac{TruePositives}{TruePositives + FalseNegatives} \tag{5}$$

$$precision = \frac{TruePositives}{TruePositives + FalsePositives} \tag{6}$$

In our problem, false negatives represent students whose classification is academic probation (class 1, 2) predicted as a normal student (class 0), and false positives represent normal students (class 0) predicted as students whose classification is academic probation (class 1, 2).

According to formulas (5) and (6), maximizing accuracy will minimize false positives, and maximizing recall will minimize false negatives in the predictions of a model. Therefore, we choose F2-Score (β=2) to put 2x emphasis on recall. This means putting more attention into minimizing false negatives than minimizing false positives. This allows us to select models that can minimize the number of students actually placed on academic probation but still miss

Besides, the combination of the F2-score with the macro-averages method provides an accurate evaluation of the imbalanced dataset because this method is not affected by class weights.

The F2-score, on the other hand, is solely used to assess binary classification systems. So, in order to apply this measure to 3-label classification problems, we must consider labels 1 and 2 to be members of the same class and perform the necessary conversion before evaluation. All it needs to do is convert the data points labeled 2 to label 1.

When evaluating 3-label classification problems, not only the F2-score but also the recall of label 2 are considered. This is due to the fact that the number of labels on label 2 is so small in comparison to label 1. As a result, the individual prediction result for label 2 was inadvertently ignored when evaluated with the F2 score. Several of the models provided outstanding F2-score results but completely erroneous label 2 predictions. A three-label classification model that performs well on both the F2-score and recall of label 2 is, therefore, the best choice.

5.5.2 Strategy for evaluation

The measures used will differ depending on the approach. As for the 3-label classification problem (warning, dismissal, and normal), as discussed in section 5.3, we approach this problem in two main directions, including the normal approach and the divided approach. In the normal approach, we perform an evaluation with the recall score of label 2 and the F2-score after converting the data. With the divided approach, model 1 is evaluated with an F2-score, and model 2 uses the F1-score because we consider the importance of labels 1 and 2 to be the same. Finally, perform a combination of these two binary classification models and evaluate the final result in the same manner as with the normal approach. As for the binary classification problem for academic probation and normal class, we only use the F2-score for conducting the evaluation.

5.6 Ensemble

The Ensemble technique is used in the experiment to find a model with superior performance. Specifically, using the popular label method (majority voting) to select the final predicted label from multiple model predictions.

6 Experiments and results

We conduct experiments based on the outline in section 5 with different preparation for each dataset. Specifically, dataset 1 was divided into two sets by train_test_split function with an 8:2 ratio. With dataset 2, we keep 10% as test data, while the remaining 90% will be divided into an 8:2 ratio to serve the model's training and evaluation.

To begin the modeling process, we generate baseline models with each method in order to identify the best optimal algorithm. This result is determined by the average result achieved by each algorithm across the entire set of our tiny datasets. Then, to attain even better outcomes, we make more intricate modifications based solely on the best optimal method found. Besides, the search parameter values are evaluated using the Stratified K-Folds cross-validator approach with K=5 to verify that the model performs well with imbalanced data and to improve the overfitting condition encountered. Furthermore, the above selection is done not only manually, but also through an automatic technique called HalvingGridSearchCV. It is a new model selection strategy that significantly reduces the time required for the model selection procedure to operate. Our experimental procedure is shown in detail in fig 3.

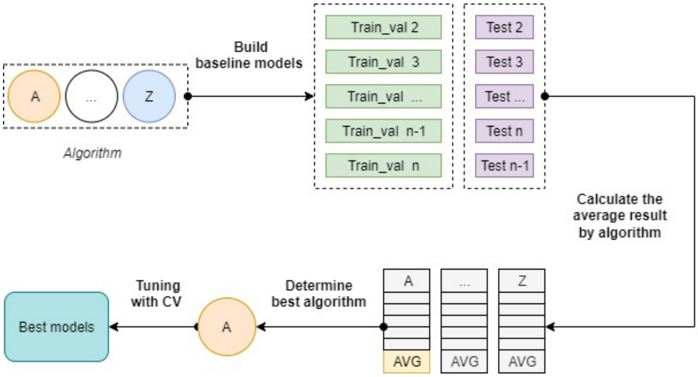

Fig. 3: Our experimental procedure.

6.1 The first warning - Beginning of each semester

Tables 7 and 8 show the results of Task 1 and Task 3 to solve the problem of the first warning, respectively.

Table 7: The best experimental results on dataset 1 with the first problem.

Methods	Algorithms	F1-macro	F2-macro
	Baseline		
Normal Approach	Decision Tree	0.6309	0.3037
	Random Forest	0.6409	0.3315
	Extra Trees	0.6423	0.3037
	Gradient Boosting Decision Tree	0.6629	0.1667
	Gradient Descent Classifier	0.6927	0.7778
	Logistic Regression	0.7060	0.8889
	Support Vector Machine (svm.LinearSVC)	0.7193	0.0370
	Support Vector Machine (svm.SVC)	0.7260	0.4907
	Proposed Method		
Normal Approach	LightGBM	0.7312	0.8889

Table 8: The best experimental results on dataset 2 with the first problem.

Algorithms	F2-macro
Baseline	
Decision Tree	0.6746
Extra Trees	0.6924
Random Forest	0.6925
Gradient Descent Classifier	0.6941
Gradient Boosting Decision Tree	0.7145
LightGBM	0.7690
Logistic Regression	0.7800
Support Vector Machine (svm.SVC)	0.7848
Proposed Method	
Support Vector Machine (svm.LinearSVC) With feature generation	0.8147

The data in the two tables above are all compiled from models that use the class-weight = "balanced" method to deal with imbalance issues. This strategy is more effective and simpler than the resampling method with RandomOverSampler in balancing the predictability of the models' labels.

Task 1 is approached on both the normal and divided approaches, but the divided approach seems not suitable in this case. This is proved by the best results in table 7 is the result of the normal approach. We conduct to determine the average suitability of each algorithm with data from dataset 1, and the algorithm that gives the best results for this challenge is LightGBM, with an F2 score of 73.12% and 89% on recall of label 2. This discovery, however, is insufficient to persuade us that the model will provide good performance value when deployed. As a result, we are urged to undertake a new task with more changes as well as more techniques applied to it, Task 3.

Table 8 clearly demonstrates that the addition of Task 3 was a fantastic idea. It outperforms what was accomplished in Task 1. In all algorithms, we can plainly observe that the results of Task 3 are significantly superior to those of Task 1. This illustrates that the reorganization, which includes a change in classification as well as the addition of new features, is really worth it. Finally, the proposed model for the first warning is a 2-label classification model trained on dataset 2 using the Linear Support Vector Classification algorithm supported by the feature creation technique, which achieves an F2-score of over 81%.

6.2 The second warning - Before the final exam

Table 9: The experimental results before the final exam.

Methods	**Algorithms**	**F2-score**	**Recall-label2**
	Baseline		
Normal Approach	LightGBM	0.8704	0.3968
	Support Vector Machine	0.8684	0.0000
	Gradient Boosting Decision Tree	0.8680	0.8571
	Logistic Regression	0.8644	0.2037
	Extra Trees	0.8605	0.1984
	Random Forest	0.8599	0.5952
	Stochastic Gradient Descent Classifier	0.8484	0.0000
	Decision Tree	0.8328	0.8519
	Proposed Method		
Normal Approach	Gradient Boosting Decision Tree With feature generation	0.9327	1.0000
Divided Approach	Random Forest & Gradient Boosting Decision Tree With feature generation	0.9623	1.0000

Table 9 shows the experimental results before the final exam, which show that adding known component scores to the model before the final exam makes the prediction results significantly more accurate., and the Gradient

Boosting Decision Tree algorithm outperforms the other algorithms when both achieve high results in both the baseline and our proposal. In particular, the feature generation technique has shown its role in the experimental process when improving the model result by approximately 10%. Furthermore, on our dataset, dividing the problem into two sub-models (section 5.3) is currently the most efficient method, with an F2-score and a recall for label 2 of 96.23% and 100%, respectively.

7 Conclusion

From a huge educational database of inanimate numbers, we develop two datasets with really valuable attributes. In addition to the semester GPA attributes, the emergence of attributes such as "pre_avg", "history" and "avg10" are outstanding results discovered from our research processes. The performance of the warning model is considerably improved as a result of these essential features, demonstrating the powerful assistance of feature generation and feature selection techniques for the modeling process. Furthermore, with the first warning, it is impossible to overlook the role of the data reorganization process in boosting model performance. With the above notion and the added features, the model's performance spiked by over 8% on the F2-score. In general, we initially achieve pretty decent results, particularly in problems prior to the final exam, when knowing the component scores of the period being forecasted greatly improves accuracy. In light of the findings, we believe that predictive models based on our suggested algorithms can help to mitigate the rise in academic probation warnings at universities, hence establishing a positive learning environment for Vietnamese students in particular and students worldwide in general.

8 Future works

In the future, we will continue to look for additional attributes in the raw database to increase the quality of the beginning of the semester model and collect new and timely data to improve model performance and check the model quality when put into reality. In addition, not stopping with warnings only based on academic performance, we always want to be able to exploit and explore more new aspects that affect students' learning status even more in order to build a more perfect alert system, such as the influences from major of a student, semesters, types of subjects and so on. Furthermore, in order to simplify the problem, we plan to discover a way to keep only one input format for all semesters while still providing good performance.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

[1] Huang, S., Fang, N.: Predicting student academic performance in an engineering dynamics course: A comparison of four types of predictive mathematical models. Comput. Educ. **61**, 133–145 (2013)

[2] Miguéis, V.L., Freitas, A., Garcia, P.J.V., Silva, A.: Early segmentation of students according to their academic performance: A predictive modelling approach. Decis. Support Syst. **115**, 36–51 (2018)

[3] Mingyu, Z., Sutong, W., Yanzhang, W., Dujuan, W.: An interpretable prediction method for university student academic crisis warning. Complex & Intelligent Systems (2021)

[4] Bujang, S.D.A., Selamat, A., Ibrahim, R., Krejcar, O., Herrera-Viedma, E., Fujita, H., Ghani, N.A.M.: Multiclass prediction model for student grade prediction using machine learning. IEEE Access **9**, 95608–95621 (2021)

[5] Quinlan, J.R.: Induction of decision trees. Machine Learning **1**, 81–106 (2004)

[6] Breiman, L.: Random forests. Machine Learning **45**, 5–32 (2004)

[7] Geurts, P., Ernst, D., Wehenkel, L.: Extremely randomized trees. Machine Learning **63**, 3–42 (2006)

[8] Friedman, J.H.: Greedy function approximation: A gradient boosting machine. Annals of Statistics **29**, 1189–1232 (2001)

[9] Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., Liu, T.-Y.: Lightgbm: A highly efficient gradient boosting decision tree. In: NIPS (2017)

[10] Hearst, M.A.: Trends & controversies: Support vector machines. IEEE Intell. Syst. **13**, 18–28 (1998)

[11] Robbins, H.E.: A stochastic approximation method. Annals of Mathematical Statistics **22**, 400–407 (2007)