applied sciences

MDPI

Article

Student Dropout Prediction for University with High Precision and Recall

Sangyun Kim [1], Euteum Choi [2], Yong-Kee Jun [3,4,*] and Seongjin Lee [5,*]

[1] Department of Informatics, Gyeongsang National University, Jinju-daero 501, Jinjusi 52828, Republic of Korea; summer@gnu.ac.kr
[2] Research Center for Aircraft Parts Technology, Gyeongsang National University, Jinju-daero 501, Jinjusi 52828, Republic of Korea; etchoi@gnu.ac.kr
[3] Division of Aerospace and Software Engineering, Gyeongsang National University, Jinju-daero 501, Jinjusi 52828, Republic of Korea
[4] Department of Bio & Medical Bigdata (BK4+ Program), Gyeongsang National University, Jinju-daero 501, Jinjusi 52828, Republic of Korea
[5] Department of AI Convergence Engineering, Gyeongsang National University, Jinju-daero 501, Jinjusi 52828, Republic of Korea
* Correspondence: jun@gnu.ac.kr (Y.-K.J.); insight@gnu.ac.kr (S.L.)

Featured Application: Application to student counseling and reducing the dropout rate in universities.

Abstract: Since a high dropout rate for university students is a significant risk to local communities and countries, a dropout prediction model using machine learning is an active research domain to prevent students from dropping out. However, it is challenging to fulfill the needs of consulting institutes and the office of academic affairs. To the consulting institute, the accuracy in the prediction is of the utmost importance; to the offices of academic affairs and other offices, the reason for dropping out is essential. This paper proposes a Student Dropout Prediction (SDP) system, a hybrid model to predict the students who are about to drop out of the university. The model tries to increase the dropout precision and the dropout recall rate in predicting the dropouts. We then analyzed the reason for dropping out by compressing the feature set with PCA and applying K-means clustering to the compressed feature set. The SDP system showed a precision value of 0.963, which is 0.093 higher than the highest-precision model of the existing works. The dropout recall and F1 scores, 0.766 and 0.808, respectively, were also better than those of gradient boosting by 0.117 and 0.011, making them the highest among the existing works; Then, we classified the reasons for dropping out into four categories: "Employed", "Did Not Register", "Personal Issue", and "Admitted to Other University." The dropout precision of "Admitted to Other University" was the highest, at 0.672. In post-verification, the SDP system increased counseling efficiency by accurately predicting dropouts with high dropout precision in the "High-Risk" group while including more dropouts in total dropouts. In addition, by predicting the reasons for dropouts and presenting guidelines to each department, the students could receive personalized counseling.

Keywords: dropout precision; dropout recall; machine learning; imbalanced data processing; hybrid method; big data; academic data; principle component analysis; K-means clustering

MSC: 68T01; 68U01

Citation: Kim, S.; Choi, E.; Jun, Y.-K.; Lee, S. Student Dropout Prediction for University with High Precision and Recall. *Appl. Sci.* **2023**, *13*, 6275. https://doi.org/10.3390/app13106275

Academic Editor: Andrea Prati

Received: 17 April 2023
Revised: 10 May 2023
Accepted: 16 May 2023
Published: 20 May 2023

1. Introduction

According to South Korea's 2022 Basic Education Statistics [1], the school-age population is declining. Compared to 2021, the 2022 school-age population of South Korean universities decreased by 2.6% and that of colleges decreased by 6.4%. The number of

students matriculating to universities has significantly decreased since the advent of the COVID-19 pandemic. The decline in the school-age population is a severe problem in South Korea.

When the school-age population declines, universities must be prepared to both recruit new students and prevent current ones from dropping out. One of the main concerns for universities is the student dropout rate, which not only costs individuals but also impacts universities, local communities, and the nation. For instance, in 2020 the dropout rate increased by 1.1%, with rates of 1.9% in metropolitan areas and 3% in provinces. Students in provincial universities may be more susceptible to dropping out due to factors such as employment prospects or career goals [2]. Those who drop out often transfer to higher-ranked universities in metropolitan areas. This trend is expected to exacerbate the already significant issue of population influx to urban areas, leading to a structural waste of educational and financial resources. Furthermore, the problem has additional side effects, including missed educational opportunities for those who could have enrolled, disruptions to the academic atmosphere, and a waste of management resources.

Universities can prevent students from wasting time and losing interest by predicting which students are likely to drop out and providing personalized support. In order to do so, universities require a system that can be responsive to students who experience a change of heart. Several studies [3–8] have examined the prediction of student dropout rates in universities. Two key factors crucial in predicting dropout rates are precision and recall. High dropout precision means that the model can accurately predict which students are likely to drop out, which is essential for directing counseling resources. Conversely, high dropout recall is important because it enables universities to identify all students who are at risk of dropping out. When only one of the indexes is high, there can be problems. For instance, a model may correctly predict twenty students who are about to drop out (100% precision) but fail to predict another forty (50% recall), resulting in missed counseling opportunities. Alternatively, a model may correctly predict all students who will drop out (100% recall) but have low precision, wasting university resources on students who will not actually drop out.

In this paper, we solve the problem of predicting the students who are about to drop out of the university using data-driven algorithms. The advantages of data-driven algorithms are their abilities to automatically learn from data, adapt to changing circumstances, and improve their performance over time. They uncover patterns and insights that may not be immediately apparent to human analysts and process large amounts of data quickly and accurately.

Improving precision and recall indexes depends on the quantity and quality of data accumulated by the university. Since universities typically maintain various student records, data quality is generally not a significant concern. However, ensuring an adequate amount of data that describe students who drop out is essential for predictive models' accuracy. Dealing with asymmetrical data is challenging because the average student dropout rate is low, at only 1.9% in metropolitan areas and 3% in provinces. Consequently, over 97% of the data do not describe students who drop out. To overcome this challenge, the preprocessing of the feature set is necessary to address imbalanced data and prevent them from negatively affecting machine learning processes.

The imbalanced data preprocessing methods can be categorized into an algorithmic approach, a data approach, and a cost-sensitive approach [9]. The algorithmic approach adjusts or tunes the model's hyperparameters to increase the model's performance. However, finding the appropriate values for the hyperparameters takes a long time, and it may only work with specific machine learning models. The data approach in the preprocessing process samples the available data, which may reduce the probability of overfitting the given data and the model. However, it may have low accuracy because only a tiny portion of the data are used. To address imbalanced data using the data approach, three methods are commonly used: oversampling [3,9], undersampling [3], and a combined approach [8]. Oversampling inflates minor class data, undersampling reduces major class data, and the

combined approach balances the benefits of both. The cost-sensitive approach is a method of re-learning data by giving different weights to misclassified data by exploiting other algorithms. Although the weights can be automatically learned, they can only work with some models. Note that algorithmic and cost-sensitive approaches depend on the specific algorithm in the supervised learning.

In this paper, we introduce the Student Dropout Prediction (SDP) system, which aims to enhance the precision and recall index of predicting student dropouts, providing valuable insights to academic administration and counselors. The SDP system identifies significant features through permutation importance and SHAP analysis and addresses data imbalance by utilizing a data approach. It predicts potential student dropouts by employing a hybrid model that combines the XGBoost model with the SMOTE oversampling method and the CatBoost model with the RandomOverSamplerSMOTEENN model. To further assist academic administration, the data are analyzed using a clustering method to identify distinct groups of students who require different types of support, such as mentoring, dormitory assistance, or scholarships.

Between 2015 and 2021, we obtained 67,060 student records from Gyeongsang National University and identified 27 essential feature sets from the available 40 features. Additionally, by predicting the reasons for dropouts and providing department-specific guidelines, we were able to offer personalized counseling to students. The contribution of this paper is as follows:

- We offer guidelines for designing a model based on the most recent dropout data from South Korea's Flagship National University.
- We propose the SDP system, a hybrid model that enhances dropout precision and recall while more accurately identifying the "high-risk" group and detecting a greater number of dropouts.
- To provide customized counseling to students at risk of dropping out, we employ a clustering algorithm to identify the reasons behind this tendency. These reasons are subsequently shared with counselors and departments for effective intervention.

Section 2 presents the related work on predicting university dropout. The characteristics and basic statistics of the data used in this paper are described in Section 3. The proposed prediction model, the SDP (Student Dropout Prediction) system, is described in Section 4. Section 5 presents the experiment results. Section 6 discusses the applicability of the presented results and suggestions to the academic administrators. Finally, Section 7 concludes the paper.

2. Related Work

Yaacob et al. [5] conducted a study on 64 computer science students in the 1st and 2nd semesters in the year 2016, measuring their academic grades in 26 courses including mathematics and IT courses. The authors experimented with several machine learning models, such as logistic regression, KNN, random forest, artificial neural networks, and decision trees, to predict the students' performance. Although the data were imbalanced, no special imbalanced data processing was applied. Logistic regression exhibited the highest accuracy and AUC values. However, the authors did not measure the dropout precision and dropout recall metrics; instead, they evaluated their model's performance using the AUC.

Shynarbek et al. [6] collected 366 student records in the department of computer science at Suleyman Demirel University, comprising grades in mathematics and computer-related courses from 2016 to 2017. The authors created a feature set using only mathematics and computer subjects and applied several machine learning models, such as the naive Bayes model, support vector machines, logistic regression, and artificial neural networks, to predict students' academic performance. Unlike Yaacob et al. [5], they used four metrics, accuracy, recall, precision, and F1 score, to measure the prediction rate. Shynarbek et al. [6] replaced the missing values with random values in the data preparation process. The data imbalance was not mentioned in the paper. The naive Bayes and artificial neural network

methods exhibited the highest accuracy (0.96) and precision (0.94), recall (0.94), and F1 (0.94) scores, respectively.

Silva et al. [7] used 331 undergraduate students' academic grades and personal information, including 23 feature sets, from the department of computer engineering at Universidade de Trás-os-Montes e Alto Douro (UTAD) from 2011 to 2019. Of the 331 students, 124 are dropouts and 207 are students who successfully graduated from the university. The authors applied several machine learning models, such as CatBoost, random forest, XGBoost, and artificial neural networks, to predict the students' academic performance. To handle imbalanced data, they applied RandomOverSampling during preprocessing. In the preprocessing process, they scaled the data with MinMaxScaler and performed RandomOverSampling as imbalanced data processing. The authors used three metrics, precision, recall, and F1 score, to evaluate the models' performance. The training/test ratio was 8:2, and they performed 10-fold cross-validation. Artificial neural networks, XGBoost, and random forest exhibited the highest precision (0.85), recall (0.83), and F1 score (0.81), respectively.

Fernández et al. [8] collected data from 1418 undergraduate students, where 783 were dropouts and 635 were non-dropouts. The feature set comprised 19 enrollment-related fields, 14 qualification-related fields, and 4 scholarship-related fields, excluding student IDs and redundant data. The authors used numerical data with MinMaxScaler and categorical data with one-hot encoding. To handle the imbalance in the data, the authors applied the SMOTETomek method, a combination of the SMOTE and Tomek links methods, during preprocessing. The authors applied several machine learning models, such as gradient noosting, random forest, support vector machine, and ensemble models, to predict the students' dropout rate in each semester. They evaluated the models' performance using the dropout recall and dropout precision metrics. In the enrollment model, the dropout recall of gradient boosting was the highest at 72.340, and dropout precision using the support vector machine method was the highest at 65.854. In the 1st semester model, the dropout recall was 82.237 for the ensemble model and gradient boosting had the highest dropout precision of 84.277. In the 2nd semester model, the ensemble model had the highest dropout recall at 82.237, and the gradient boosting had the highest dropout precision at 79.245. In the 3rd semester model, both the dropout recall and dropout precision were the highest in the random forest model, at 88.462 and 86.792, respectively. In the 4th semester model, the dropout recall and dropout precision were the highest in the support vector machine model, at 91.549 and 89.041, respectively.

Barros et al. [3] gathered 7718 student records from the Federal Institute of Rio Grande do Norte, utilizing 6 mathematics and 19 demographics- and socio-economic-related courses as feature sets. To deal with imbalanced data, they employed downsampling, SMOTE, ADYSYN, and balanced bagging techniques for each experiment. They tested artificial neural network and decision tree models with training/test ratios of 75% and 25%, respectively. The highest precision was obtained using the oversampling (SMOTE, ADAYSYN) technique of artificial neural networks at 0.991. For recall and F1, the unprocessed decision tree method performed the best, at 0.977 and 0.976, respectively.

Baranyi et al. [4] not only acquired the university transcript and personal information but also utilized high school grades to predict dropouts. They used balanced 8319 students from 2013 to 2019 at the Budapest University of Technology and Economics, composing 30 feature sets—5 related to the university program, 21 to high school, and 4 to personal data. They tested various models, such as artificial neural networks, Tabnet, XGBoost, random forest, and BaggingFCNN. They optimized the hyperparameters of artificial neural networks using the hyperas package. The authors also used SHAP analysis to identify the most influential variables and found that "years elapsed" (the years since the matura examination) was the most influential variable, followed by grade-related features such as "University admission score". The experiment showed that artificial neural networks had the highest precision of 0.747 and recall of 0.667.

Niyogisubizo et al. [10] predicted class dropout using data from Constantine the Philosopher University in Nitra from 2016 to 2020. The authors utilized primary data, including "tests", "access", and "project", which had a high correlation. They stacked random forest, XGBoost, and gradient boosting and used the output as input for artificial neural networks. The stacking ensemble showed high performance with overall precision, recall, and F1 score values of 0.93, 0.93, and 0.92, respectively, of midpoint and midpoint deviation.

The review of related work found that many previous studies had small data sets and did not address imbalanced data. Additionally, most of the works considered academic grades as the most important predictor of student dropout. However, the methods and data sets used in these studies varied, making it challenging to compare the models. Furthermore, the results of previous works often only showed high precision or recall metrics but not both, and the reasons for dropout were not analyzed.

To address these gaps, the authors of this study used a large data set spanning five years and included student activities in addition to academic grades. We also compared the proposed approach with existing models using our data to ensure a fair comparison. We used a hybrid model to achieve high precision and recall rates for predicting student dropout. Finally, we analyzed the reasons for dropout to assist counselors and administrators in supporting students and making informed decisions. Overall, this study contributes to the field by using a comprehensive approach that considers various factors to predict student dropout and analyzes the underlying reasons for dropout.

3. Data Characteristics

3.1. Primitive Statistics on Data

We acquired the student records of Gyeongsang National University from the year 2016 to 2022. After sanitizing the data, the total number of students we used for the experiment was 67,060. Note that the university acquires consent to utilize the data when the students are admitted to the university. To protect the students' privacy, we performed data anonymization to sanitize the sensitive information before analyzing the data. The university is located in the southern region of South Korea and is one of nine Flagship National Universities of South Korea. It has 14 colleges and 375 departments. The number of students registered at the university sums up to about 13,549. Every year, about 3266 students are matriculated and about 3303 students graduate from the university. Table 1 shows the summary of the total number of students, the number of students enrolled, and the number of students who dropped out of the university during that year. The average dropout rate during the five years was 5.1%. Table 2 further distinguishes students by academic year, the female and male ratio of students, and the dropout ratio. It shows that the university has, on average, 23% more male students than female students each academic year. It shows that male students tend to break away from the university more than female students. The data show that, on average, 131% more male students drop out of the university for various reasons. It also shows that freshmen drop out of the university the most (7.2%) and seniors drop out the least (2.1%).

Table 1. The Total Number of Students and the Number of Students Dropped Out During Year 2016–2020.

Year	Total No. of Students	No. of Enrolled Students	No. of Dropouts	Ratio
2016	15,667	15,038	629	4.1%
2017	14,932	14,220	712	5%
2018	14,662	13,822	840	6%
2019	15,276	14,572	762	5.2%
2020	14,771	14,146	742	5.2%

Table 2. The Total Number of Students in Each Academical Year and The Number of Students Dropped Out in 1 March 2020.

Academical Year	Total No. of Students	Female/Male Ratio of Total	No. of Dropouts	Female/Male Ratio of Dropouts	Dropout Ratio
Freshman	3481	85.4% (F:1604/M:1877)	252	68% (F:102/M:150)	7.2%
Sophomore	3533	76.5% (F:1532/M:2001)	240	29.7% (F:55/M:185)	6.7%
Junior	3587	81.6% (F:1612/M:1975)	162	40.8% (F:47/M:115)	4.5%
Senior	4170	81.7% (F:1876/M:2294)	88	29.4% (F:20/M:68)	2.1%

Table 3 shows the ratio of dropouts living near the university and students from other provinces. In general, the number of students from other provinces is more than that of students living in the same province. The number of students dropping out of the university from other provinces gradually decreases, but the number stays almost indifferent in all years. From the data, we can deduce that the administrators have to take care of the first- and second-year students more than the junior and senior students. It is also advisable to take different approaches in counseling students in the different academic years.

Table 3. Rate of Dropout with Respect to Born Region in 1 March 2020.

Academical Year	Dropouts	Same Province	Other Province	Ratio
Freshman	252	39	213	18%
Sophomore	240	35	205	17%
Junior	162	43	119	36%
Senior	88	36	52	69%

Table 4 describes the reasons the freshmen gave before dropping out of the university. About 54% of the students answered that they were admitted to another university. It means the student has been preparing for the entrance exam for about a year. The second most frequent reason was that the student was going through hard times or could not continue due to various family affairs. Although the students are categorized as dropouts, their academic performance may vary significantly. For example, the students admitted to another university may have positive course grades, and those who did not register may have negative course grades.

Table 4. The Freshmen's Reasons for Dropping Out of the University (1 March 2020).

Dropout Reason	Count	Ratio
Admitted to Other University	158	54%
Personal Issues (Health, Family Affairs)	57	20%
Voluntary Dropout	39	13%
Did Not Register	22	8%
Misc.	16	5%

The data show that the number of students dropping out of university is a minor class. Identifying a small number of students who are about to drop out accurately is challenging. Highly imbalanced data create a bias towards the majority class. In some extreme cases, the minority class in imbalanced data may be completely ignored during the learning phase of the model [11].

3.2. Features

There were many features, but the existing works tell us that not all data help predict dropouts. We analyzed the data with permutation importance and SHAP analysis to reduce the features and increase the prediction performance. Some of the data had to be encoded to ordinal before applying permutation importance and SHAP analysis, and one-hot encoding was used to encode the categorical data. We ran XGBoost ten times to measure the permutation importance, and Table 5 shows the top five features that showed the highest importance; however, the numbers were not high.

Table 5. Permutation Importance of Input Data.

Rank	Feature Name	Permutation Importance
1	Grade Ranking	0.0177
2	Completed Credits	0.0070
3	Department	0.0043
4	Grade Average	0.0040
5	Univ. Main Site Login Count	0.0040

Figure 1 shows the results of the SHAP analysis. The blue and red colors indicate the low and high relationships between the input and output of the model, respectively. Just like the permutation importance, it also shows that the dropout rate is correlated with completed course credits and grade ranking. One interesting point is that the login count of the main university website shows a high correlation with the dropout rate.

We performed principal component analysis and autoencoder denoising on the data, but it had little effect on the result. We ignored the unknown category and used the SimpleInputer scikit-learn package to fill in the missing values with the median of the feature. Note that we added facility usage history along with academic records. The complete list of features used in the prediction is included in Table 6. There were five feature classes, which are as follows: academic data, academic records, personal information, facility use history, and website use history.

Figure 1. SHAP of Characteristics of Feature Set with Respect to Dropout Information.

Table 6. Feature Set Used in the Paper.

Feature Class	Feature Name	Type
Academic Data	Academic Status	Real
	Number of Rewards and Penalty	Real
	Registration Fee Installments Count	Real
	Number of Volunteer Activities	Real
	Department	Categorical
	Number of Leave of Absence	Real
	Word Count of Counsel Report	Real
	Seasonal Semester Courses Count	Real
	Collage	Categorical
Academic Records	Grade Ranking	Real
	Grade Average	Real
	Number of Majors	Real
	Completed Credits	Real
	Number of Scholarships Received	Real
	Completed Credit of Seasonal Semester Courses	Real
	Total Amount of Scholarships	Real
Personal Information	Disability	Binary
	Residence Postal Code	Categorical
	Gender	Binary
	Counseling Count	Real
Facility Use History	Library Overdue Count	Real
	Library Loan Count	Real
	Dormitory Use Count	Real
	Dormitory Penalty Count	Real
	Number of Facility Rentals	Real
Website Use History	Univ. Main Site Login Count	Real

3.3. Measurements of Existing Methods

In this paper, we are interested in devising a method that produces high precision and recall. Since the existing works used different data sets and features, comparing the performance of existing works is challenging. To understand existing works' precision and recall performance, we used our data to measure the metrics. The results of running the methods used in the existing works are illustrated in Figure 2. The methods used in the experiment are TomekLinks [12], RandomUnderSampler [13], EditNearestNeighbours [14], SMOTE [15], BorderlineSMOTE [16], ADASYN [17], SMOTEENN [18], and SMOTETomek [19]. We used 10-fold cross-validation. We chose ensemble, logistic regression, artificial neural network, and gradient boosting methods to compare with the method (SDP) proposed in this paper because, in general, they showed high precision and high recall rates in the previous works.

As shown in Figure 2, the unprocessed methods showed high precision, whereas the imbalanced data processing models showed high recall. In addition, gradient boosting and ensemble methods showed relatively high precision and recall rates. Logistic regression did not show good results on our data.

We have to emphasize that we need to have high precision and recall to correctly identify the students and all the students who are about to drop out of the university.

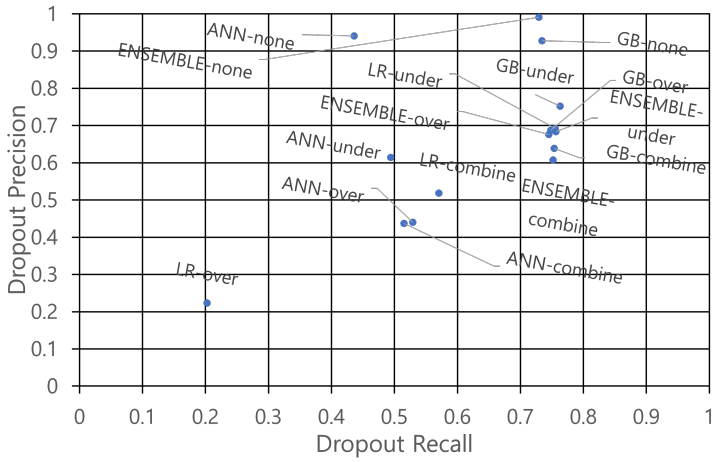

Figure 2. Dropout precision and recall performance of the existing works (GB: gradient boosting; ensemble: ensemble of gradient boosting, random forest, and support vector machine; LR: logistic regression; ANN: artificial neural network; none: unprocessed; under: undersampling; over: oversampling; combine: undersampling and oversampling).

4. SDP (Dropout Student Prediction System)

This paper presents the SDP (Student Dropout Prediction) system for predicting students who are about to drop out of universities. The architecture of the SDP system is shown in Figure 3. We first preprocess the data using SMOTE and RandomOverSamplerSMOTEENN to treat the imbalanced data. Then, we combine the dropout prediction results of XGBoost and CatBoost to produce high precision and recall. The system utilizes both models depending on the university's status and needs. A high-precision model can identify candidates who require higher priority in consultation, and a recall model can encompass a wider range of potential dropouts. By using the two different models in a system, we were able to capture the advantages of the two models in a system. We first introduce the imbalanced data processor, then present the dropout predictor that combines the result of XGBoost and CatBoost methods.

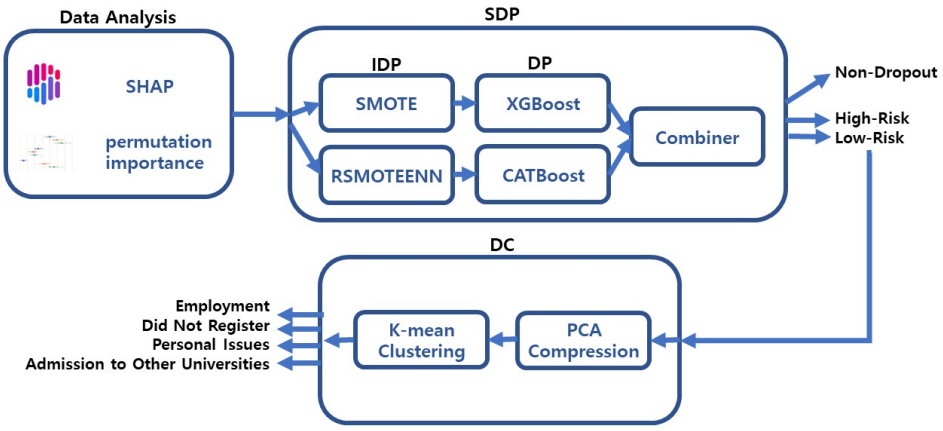

Figure 3. The University Dropout Prediction Design Structure.

4.1. Imbalanced Data Processor

We use SMOTE [15] and RandomOverSamplerSMOTEEN on the imbalanced data for producing high precision and high recall, respectively. SMOTE creates synthetic data by oversampling the minor class in the data. We use the imbalanced-learn package for the SMOTE algorithm. Since the SMOTE algorithm inflates the data of the minor class, it increases the dropout precision of the minor class in the data of Gyeongsang National University. We use RandomOverSamplerSMOTEENN to produce a better recall rate than

SMOTEENN. Listing 1 describes the pseudo-code for RandomOverSamplerSMOTEENN. We first apply RandomOverSampler and then apply SMOTE and ENN in order. The RandomOverSamplerSMOTEENN method uses data oversampling and undersampling simultaneously to reinforce the minor class data by inserting an additional 10% more random value before balancing the data. This method has the advantage of increasing the dropout recall in the dropouts class, which is a minor class in the data of Gyeongsang National University. We use make_pipeline, RandomOverSampler, and SMOTEENN libraries and the specific algorithm RandomOverSamplerSMOTEENN shown in the Listing 1.

Listing 1. RandomOverSamplerSMOTEENN.

```
input: training data (cleaned data)
output: RandomOverSamplerSMOTEENN data
method :
    set size of majority class to Smaj
    set size of minority class to Smin
    repeat until Smin >= Smaj :
        RandomOverSampler :
            repeat until Smaj * 0.1 <= Smin :
                set random data of minority class to dmin
                compute: Smin.append(dmin)
        SMOTE :
            set random data of minority class to dmin
            compute: diff = dmin and Smin by using
            KNN = computeKNearestNeighbor():
                for k in KNN:
                    compute: gap = random number between 0 and 1
                    compute: synthetick = diff * gap
                    compute: Smin.append(synthetick)
        ENN :
            set random data of minority class to dmin
            set random data to dmin
            set 3 nearest neighbors of dmin to knn3
            set (more than half of) class name of (d and knn3)
                to enn_clsName
            if enn_clsName <> class_of(dmin):
                remove knn3
```

4.2. Dropout Predictor

The SDP system uses XGBoost [20] and CatBoost for high precision and recall, respectively. XGBoost is based on the gradient boosting framework and provides a parallel tree boosting (also known as GBDT and GBM), which solves many data science problems quickly and accurately. We set values with a high metric selected using GridSearchCV for other hyperparameters in the scikit-learn package. The tree method parameter is set to `gpu_hist` to use GPU using distributed training. CatBoost [21] is also a gradient-boosting framework that attempts to solve problems by making permutations of the categorical features. We set values with high metric selections using GridSearchCV in the scikit-learn package.

Once the two models produce the output, we combine the prediction results using the combiner depicted in Figure 3. We prioritize the model producing a high prediction rate over a high recall rate for efficient counseling of the prospective candidates.

We defined the combiner's rules as "High-Risk" dropouts when the high-precision model predicted "True". Next, we defined "Low-Risk" dropouts when even one model predicted "True". Finally, we defined non-dropout as when all models predicted "False". Table 7 describes the summary of the decision.

Table 7. Risk Criteria.

Model/Risk	High Risk	High Risk	Low Risk	Non-Dropout
High-Precision Model (SMOTE+XGBoost)	True	True	False	False
High-Recall Model (RandomOverSamplerSMOTEENN+CatBoost)	True	False	True	False

5. Experiment and Analysis

5.1. Environment

As we analyzed in Section 3, the features used in the existing works vary. We ran preliminary experiments using our data with the existing methods and identified important features using permutation importance and SHAP analysis. Furthermore, we found that most existing works did not meet our criteria of producing high dropout precision and recall rates. We excluded senior students' records because they were about to graduate from the university. After sanitization and anonymization of the data, we gathered 67,060 student records. The features used in the paper are described in Table 6. Table 8 describes the hardware and software specifications and versions used for the experiment.

Table 8. Experiment Environment.

Category	Type	Description and Version
Hardware	CPU	AMD Ryzen9
	Memory	DDR4 128GByte
	Mainboard	X570 AORUS ELITE
	Storage	Samsung SDD 970 plus 1TB
	GPU	Geforce RTX 2080 super
Software	Python	3.8
	Pytorch-tabnet	3.1.1
	Catboost	0.26.1
	Xgboost	1.4.2
	Lightgbm	3.2.1
	Scikit-learn	0.24.2
	Numpy	1.19.5
	Pandas	1.2.4
	Category-encoders	2.5.1.post0
	Imbalanced-learn	0.8.0

We preprocessed the data for XGBoost and CatBoost with SMOTE and RandomOverSamplerSMOTEENN, respectively. We used 10-fold cross-validation to reduce the dependencies on the data in all experiments unless otherwise stated. The architecture of the SDP system is illustrated in Figure 3. The results of the two methods are combined in the combiner, and the results are prioritized based on the precision rate. We further analyzed the result to identify "Low-Risk", and "High-Risk" groups; the identified groups can be delivered to the counselors and administrators for decision-making and supportive actions.

The metrics used for measuring the prediction performance are based on the confusion matrix. We used the information on dropout as the label, and the definitions of the metrics we used are as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$DropoutRecall = \frac{TP}{TP + FN}$$

$$DropoutPrecision = \frac{TP}{TP + FP}$$

$$DropoutF1 = 2 * \frac{DropoutPrecision * DropoutRecall}{DropoutPrecision + DropoutRecall}$$

5.2. Analysis of Imbalanced Data Processing Methods

As we discussed in Section 3, the dropout record in our data is imbalanced and is a minor class, which has an adverse effect on producing high precision and recall rates in Figure 4a–d.

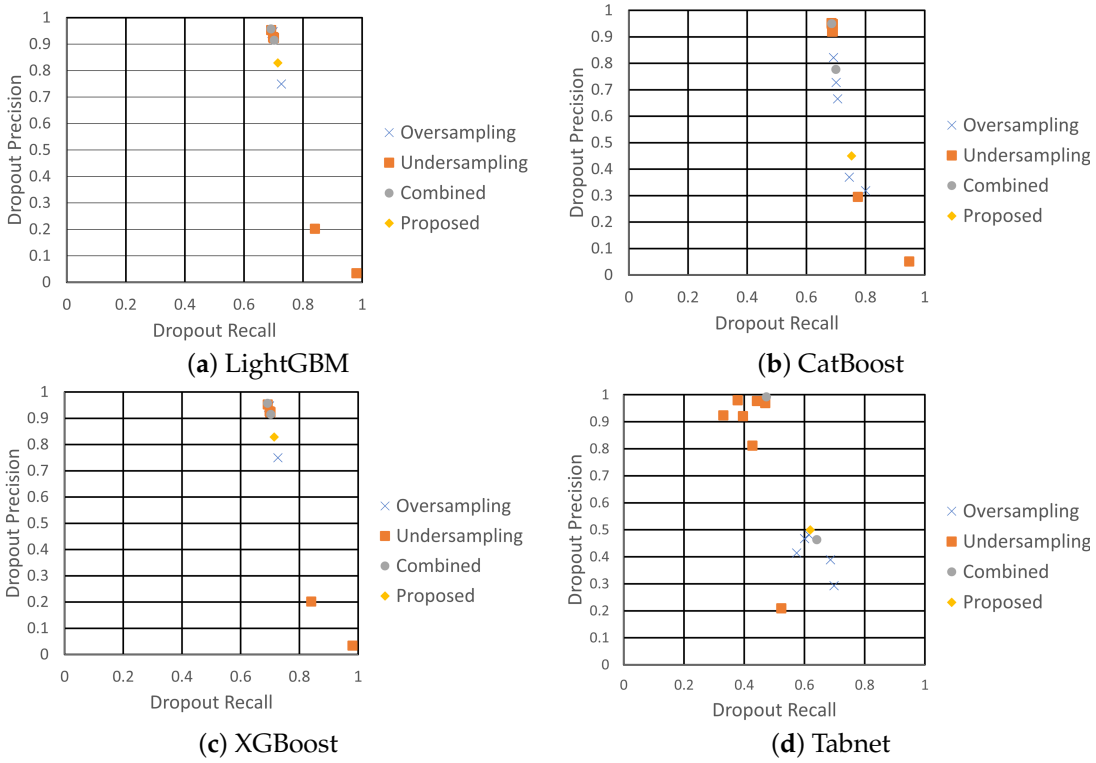

Figure 4. Comparison of Imbalanced Data Processing Methods.

To produce a high recall rate, which other methods failed to do, we used RandomOverSamplerSMOTEENN. In this section, we measure the performance (accuracy, recall, precision, F1) of XGBoost, LightGBM, CatBoost, and Tabnet combined with different over/under sampling algorithms. The undersampling methods we tested are TomekLinks [12], RandomUnderSampler [13], NeighborhoodCleaningRule [22], AllKNN [23], RepeatedEditedNearestNeighbours [23], EditedNearestNeighbours [14], and ClusterCentroids [24]. The oversampling methods we tested are SMOTE [15], BorderlineSMOTE [16], ADASYN [17], RandomOverSampler [25], and SVMSMOTE [26]. We tested the combined methods SMOTEENN [18] and SMOTETOMEK [19]. The proposed method, RandomOverSamplerSMOTEENN, combines the advantages of the RandomOverSampler and SMOTEENN methods. There were ensemble models, but we excluded them because of their complexity. Tables 9–12 show the experiment results of XGBoost, LightGBM, CatBoost, and Tabnet combined with different imbalanced data processing methods, respectively.

Table 9. Performance of XGBoost with Imbalanced Data Processing.

Category	Processing	Accuracy	Dropout Recall	Dropout Precision	Dropout F1
Undersampling	TomekLinks	0.990 (±0.00)	0.692 (±0.09)	0.952 (±0.05)	0.802 (±0.06)
	RandomUnderSampler	0.904 (±0.04)	0.840 (±0.06)	0.202 (±0.09)	0.325 (±0.11)
	NeighbourhoodCleaningRule	0.990 (±0.01)	0.701 (±0.08)	0.923 (±0.15)	0.797 (±0.10)
	AllKNN	0.990 (±0.01)	0.700 (±0.08)	0.921 (±0.12)	0.796 (±0.11)
	RepeatedEditedNearestNeighbours	0.990 (±0.02)	0.698 (±0.08)	0.926 (±0.12)	0.796 (±0.10)
	EditedNearestNeighbours	0.990 (±0.02)	0.698 (±0.08)	0.926 (±0.12)	0.796 (±0.10)
	ClusterCentroids	0.232 (±0.05)	0.981 (±0.07)	0.034 (±0.11)	0.065 (±0.12)
Oversampling	SMOTE	0.990 (±0.03)	0.698 (±0.07)	0.947 (±0.10)	0.803 (±0.10)
	BorderlineSMOTE	0.990 (±0.05)	0.695 (±0.07)	0.940 (±0.11)	0.799 (±0.12)
	ADASYN	0.990 (±0.05)	0.696 (±0.07)	0.945 (±0.11)	0.802 (±0.12)
	RandomOverSampler	0.985 (±0.05)	0.726 (±0.07)	0.749 (±0.11)	0.737 (±0.12)
	SVMSMOTE	0.990 (±0.05)	0.698 (±0.07)	0.946 (±0.11)	0.803 (±0.12)
Combined	SMOTEENN	0.989 (±0.02)	0.702 (±0.07)	0.914 (±0.10)	0.794 (±0.10)
	SMOTETOMEK	0.990 (±0.03)	0.691 (±0.07)	0.956 (±0.10)	0.803 (±0.10)
Proposed	RSMOTEENN	0.988 (±0.03)	0.714 (±0.07)	0.828 (±0.10)	0.767 (±0.10)

Table 10. Performance of LightGBM with Imbalanced Data Processing.

Category	Processing	Accuracy	Dropout Recall	Dropout Precision	Dropout F1
Undersampling	TomekLinks	0.990 (±0.00)	0.695 (±0.09)	0.965 (±0.01)	0.808 (±0.06)
	RandomUnderSampler	0.902 (±0.04)	0.845 (±0.07)	0.199 (±0.08)	0.323 (±0.09)
	NeighbourhoodCleaningRule	0.990 (±0.01)	0.697 (±0.08)	0.939 (±0.14)	0.800 (±0.11)
	AllKNN	0.990 (±0.02)	0.700 (±0.07)	0.939 (±0.10)	0.802 (±0.09)
	RepeatedEditedNearestNeighbours	0.990 (±0.01)	0.697 (±0.07)	0.936 (±0.09)	0.799 (±0.09)
	EditedNearestNeighbours	0.990 (±0.01)	0.701 (±0.07)	0.951 (±0.09)	0.807 (±0.09)
	ClusterCentroids	0.122 (±0.02)	0.988 (±0.07)	0.030 (±0.07)	0.058 (±0.08)
Oversampling	SMOTE	0.990 (±0.02)	0.694 (±0.07)	0.959 (±0.06)	0.805 (±0.06)
	BorderlineSMOTE	0.990 (±0.02)	0.695 (±0.07)	0.959 (±0.07)	0.806 (±0.08)
	ADASYN	0.990 (±0.02)	0.694 (±0.07)	0.952 (±0.07)	0.803 (±0.08)
	RandomOverSampler	0.984 (±0.02)	0.735 (±0.07)	0.703 (±0.07)	0.719 (±0.08)
	SVMSMOTE	0.990 (±0.02)	0.696 (±0.07)	0.959 (±0.07)	0.806 (±0.08)
Combined	SMOTEENN	0.990 (±0.02)	0.696 (±0.07)	0.940 (±0.06)	0.800 (±0.06)
	SMOTETOMEK	0.990 (±0.02)	0.694 (±0.07)	0.966 (±0.06)	0.808 (±0.06)
Proposed	RSMOTEENN	0.988 (±0.02)	0.719 (±0.07)	0.830 (±0.06)	0.770 (±0.06)

Table 11. Performance of CatBoost with Imbalanced Data Processing.

Category	Processing	Accuracy	Dropout Recall	Dropout Precision	Dropout F1
Undersampling	TomekLinks	0.990 (±0.04)	0.686 (±0.09)	0.948 (±0.06)	0.796 (±0.06)
	RandomUnderSampler	0.942 (±0.03)	0.774 (±0.08)	0.294 (±0.12)	0.427 (±0.12)
	NeighbourhoodCleaningRule	0.989 (±0.07)	0.687 (±0.08)	0.918 (±0.11)	0.786 (±0.08)
	AllKNN	0.990 (±0.01)	0.690 (±0.08)	0.948 (±0.07)	0.798 (±0.06)
	RepeatedEditedNearestNeighbours	0.990 (±0.01)	0.685 (±0.09)	0.951 (±0.08)	0.796 (±0.07)
	EditedNearestNeighbours	0.989 (±0.01)	0.690 (±0.09)	0.920 (±0.08)	0.788 (±0.07)
	ClusterCentroids	0.507 (±0.01)	0.948 (±0.09)	0.050 (±0.08)	0.095 (±0.08)
Oversampling	SMOTE	0.987 (±0.01)	0.691 (±0.09)	0.821 (±0.08)	0.751 (±0.08)
	BorderlineSMOTE	0.982 (±0.01)	0.705 (±0.09)	0.665 (±0.08)	0.685 (±0.08)
	ADASYN	0.957 (±0.01)	0.745 (±0.09)	0.369 (±0.08)	0.493 (±0.08)
	RandomOverSampler	0.947 (±0.01)	0.800 (±0.09)	0.318 (±0.08)	0.455 (±0.08)
	SVMSMOTE	0.984 (±0.01)	0.700 (±0.09)	0.727 (±0.08)	0.714 (±0.08)
Combined	SMOTEENN	0.986 (±0.01)	0.700 (±0.09)	0.776 (±0.09)	0.736 (±0.08)
	SMOTETOMEK	0.990 (±0.01)	0.686 (±0.09)	0.949 (±0.08)	0.796 (±0.07)
Proposed	RSMOTEENN	0.967 (±0.01)	0.752 (±0.09)	0.450 (±0.08)	0.563 (±0.07)

Table 12. Performance of Tabnet with Imbalanced Data Processing.

Category	Processing	Accuracy	Dropout Recall	Dropout Precision	Dropout F1
Undersampling	TomekLinks	0.982 (±0.00)	0.396 (±0.09)	0.920 (±0.07)	0.553 (±0.06)
	RandomUnderSampler	0.981 (±0.01)	0.427 (±0.14)	0.811 (±0.20)	0.559 (±0.13)
	NeighbourhoodCleaningRule	0.982 (±0.00)	0.378 (±0.10)	0.979 (±0.11)	0.546 (±0.13)
	AllKNN	0.984 (±0.05)	0.442 (±0.06)	0.976 (±0.05)	0.609 (±0.08)
	RepeatedEditedNearestNeighbours	0.980 (±0.01)	0.331 (±0.07)	0.923 (±0.04)	0.487 (±0.15)
	EditedNearestNeighbours	0.984 (±0.01)	0.470 (±0.07)	0.969 (±0.04)	0.633 (±0.15)
	ClusterCentroids	0.932 (±0.00)	0.523 (±0.07)	0.209 (±0.14)	0.298 (±0.07)
Oversampling	SMOTE	0.970 (±0.04)	0.600 (±0.06)	0.467 (±0.35)	0.525 (±0.11)
	BorderlineSMOTE	0.971 (±0.00)	0.615 (±0.07)	0.480 (±0.14)	0.539 (±0.07)
	ADASYN	0.945 (±0.00)	0.698 (±0.07)	0.293 (±0.14)	0.412 (±0.07)
	RandomOverSampler	0.965 (±0.00)	0.574 (±0.07)	0.414 (±0.14)	0.481 (±0.07)
	SVMSMOTE	0.961 (±0.00)	0.686 (±0.07)	0.388 (±0.14)	0.496 (±0.07)
Combined	SMOTEENN	0.969 (±0.00)	0.641 (±0.07)	0.463 (±0.13)	0.537 (±0.08)
	SMOTETOMEK	0.985 (±0.04)	0.473 (±0.06)	0.992 (±0.05)	0.641 (±0.11)
Proposed	RSMOTEENN	0.972 (±0.04)	0.619 (±0.06)	0.499 (±0.15)	0.553 (±0.11)

Tables 9–12 show that ClusterCentroids have the highest recall on XGBoost, LightGBM, and CatBoost at 0.981, 0.988, and 0.948, respectively. SMOTETOMEK showed the highest precision on LightGBM, Tabnet, and XGBoost at 0.966, 0.992, and 0.956, respectively. SMOTETOMEK also showed the highest accuracy on LightGBM, Tabnet, and XGBoost at 0.99, 0.985, and 0.99, respectively. The ClusterCentroids method iteratively replaces the non-dropout data with dropout data based on the median value of the dropout data. However, this method was not suitable because the dropout precision and dropout F1 were significantly lower in all models and the matrix varied from measurement to measurement. However, the RandomOverSamplerSMOTEENN method had high dropout recall while maintaining dropout precision. As seen from Tables 9–12, since the data of Gyeongsang National University are imbalanced, we could not find a model with both high dropout precision and high dropout recall. Therefore, we selected a model with high dropout precision and a model with high dropout recall, then created a hybrid model to increase both metrics. We selected the group with the highest dropout precision and dropout recall and sorted by the highest ROC curve and dropout F1 to select the top model. XGBoost with SMOTE is the model with the highest true positive rate when the false positive rate is low on the ROC curve. Moreover, the dropout precision of XGBoost with SMOTE is 0.947 and its dropout F1 is 0.803, which makes it the highest-ranked model. CatBoost's RandomOverSamplerSMOTEENN is the model with the highest area under the curve and highest dropout recall of 0.752 on average across grades in the ROC curve comparing imbalanced data processing in Figure 5a. The model was also stable over repeated measurements for all other metrics including the dropout recall.

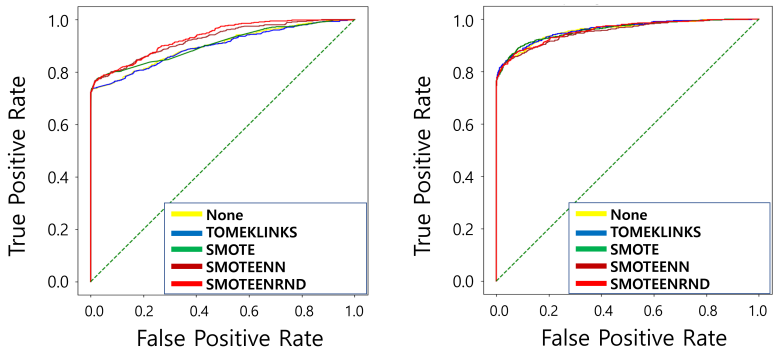

Figure 5. ROC Curve (Freshman).

Figure 5a,b show the ROC curve of XGBoost, a candidate for the high precision group, and CatBoost, a candidate for the high dropout recall group, using freshman data, which exhibits the highest dropout number. The yellow line represents "Non-Processing", the blue line represents "TomekLinks", the green line represents "SMOTE", and the brown line represents "SMOTEENN". The red represents "RandomOverSamplerSMOTEENN", denoted as "SMOTEENRND" in the graphs. CatBoost showed distinct differences depending on the imbalanced data processing method, and XGBoost showed less sensitive results. In "ROC Curves of 1st Grade CatBoost", the model using RandomOverSamplerSMOTEENN showed good performance. The SMOTE model showed a slightly higher performance in "ROC Curves of 1st Grade XGBoost." Figure 5a shows that when the false positive rate is less than 0.8, CatBoost shows high volatility in the ROC curve. In Figure 5b, XGBoost shows relatively low volatility and high dropout precision.

After analyzing various imbalanced data preprocessing methods and metrics, we chose to use XGBoost with SMOTE (Model 1) and CatBoost with RandomOverSamplerSMOTEENN (Model 2). Table 13 describes the performance of each model. To test the SDP system, 46,104 data records (70% of the total 67,060 data) were used as training data, and 20956 data records (30% of the data) were used as test data. The results show that both models predict with high accuracy. The dropout precision was higher in Model 1 by 17%,

21%, and 9% for each grade, and in the case of dropout F1, it was higher by 7%, 5%, and 3%, respectively. The dropout recall was 3%, 4%, and 2% higher in Model 2 for each grade. In summary, Model 1 had high dropout precision and dropout F1, and Model 2 had high dropout recall. When high-dropout-precision Model 1 predicts students as dropout candidates using the combiner's rule in Table 7, we categorize them as high-risk and advise counselors to consult them first. When high-dropout-recall Model 2 predicts students as dropout candidates, we categorize them as low-risk. When all models categorize students as non-dropout, we exclude them from counseling.

Table 13. Performance of SDP with Respect to Academic Year.

Academic Year	Model	Accuracy	Dropout Recall	Dropout Precision	Dropout F1
Freshman	Model 1	99%	76%	95%	85%
	Model 2	98%	79%	78%	78%
Sophomore	Model 1	98%	65%	96%	77%
	Model 2	98%	69%	75%	72%
Junior	Model 1	99%	80%	98%	88%
	Model 2	99%	82%	89%	85%

5.3. Prediction Performance of the SDP System

We compare the prediction performance of the SDP system with logistic regression, artificial neural network, gradient boosting, and ensemble (gradient boosting, random forest, and support vector machine) methods. Note that the SDP system combines the results from the two models and prioritizes the results over precision. Table 14 summarizes the results. The proposed method's accuracy, dropout recall, dropout precision, and F1 metrics show the best results compared to the other methods. The algorithm ranked in second place for accuracy and dropout recall rate is artificial neural networks. In this experiment, we organized data by combining students of all grades. In the case of SDP, we inferred the results from each model and combined them to make one result, and the metric was measured by comparing it with the validation result set. As for the SDP system, its accuracy and dropout F1 were 0.989 and 0.786, respectively, which were higher than the other models. As for the dropout precision, artificial neural networks showed the highest value at 0.870, but their dropout recall was 0.442, which was lower than average compared to the other models. The gradient boosting method had the same dropout recall value as the SDP system at 0.755, but the dropout precision and dropout F1 were 0.181 and 0.095 lower than the SDP, respectively. In this experiment, we can see that the SDP made with the hybrid model came out better than the other models.

Table 14. Comparison of related studies.

Model Type	Imbalanced Data Processing	Model Accuracy	Dropout Precision	Dropout Recall	Dropout F1
Logistic Regression	SMOTE	0.953	0.223	0.202	0.212
Artificial Neural Networks	RandomOverSampler	0.982	0.870	0.442	0.583
Gradient Boosting	SMOTETOMEK	0.980	0.638	0.755	0.691
Ensemble	SMOTETOMEK	0.978	0.606	0.749	0.670
SDP (XGBoost, Catboost) (This Paper)	SMOTE, RandomoversamplerSMOTEENN	0.989	0.819	0.755	0.786

5.4. Classification by Reason for Dropout

Once the SDP system predicted the students who are about to drop out of the university, we classified the reasons for dropping out. We used a total of 1269 dropout student records for the classification. We encoded the categorical data using ordinal encoding. We conjectured that if dropouts shared similar reasons for their dropout, their data would

also be similar. Thus, we used PCA to identify the fields associated with the reasons for dropping out of the university. We have categorized the reasons in to the following labels: "Employment", "Did Not Register", "Admitted to Other University", and "Personal Issues." We used PCA to compress the features listed in Table 6 into five principal components. Figure 6 illustrates the scatter plot of five principal components of PCA. The component in PC1 was "Number of Scholarships Received" with a value of 1.000; in PC2, there was "Residence Postal Code" and "Login Count" with values of 0.917 and 0.386, respectively; PC3 had "Login Count" with a value of 0.900; PC4 had "Department" with a value of 0.995; and PC5 had "Grade Ranking" and "Completed Credits" with values of 0.580 and 0.643, respectively.

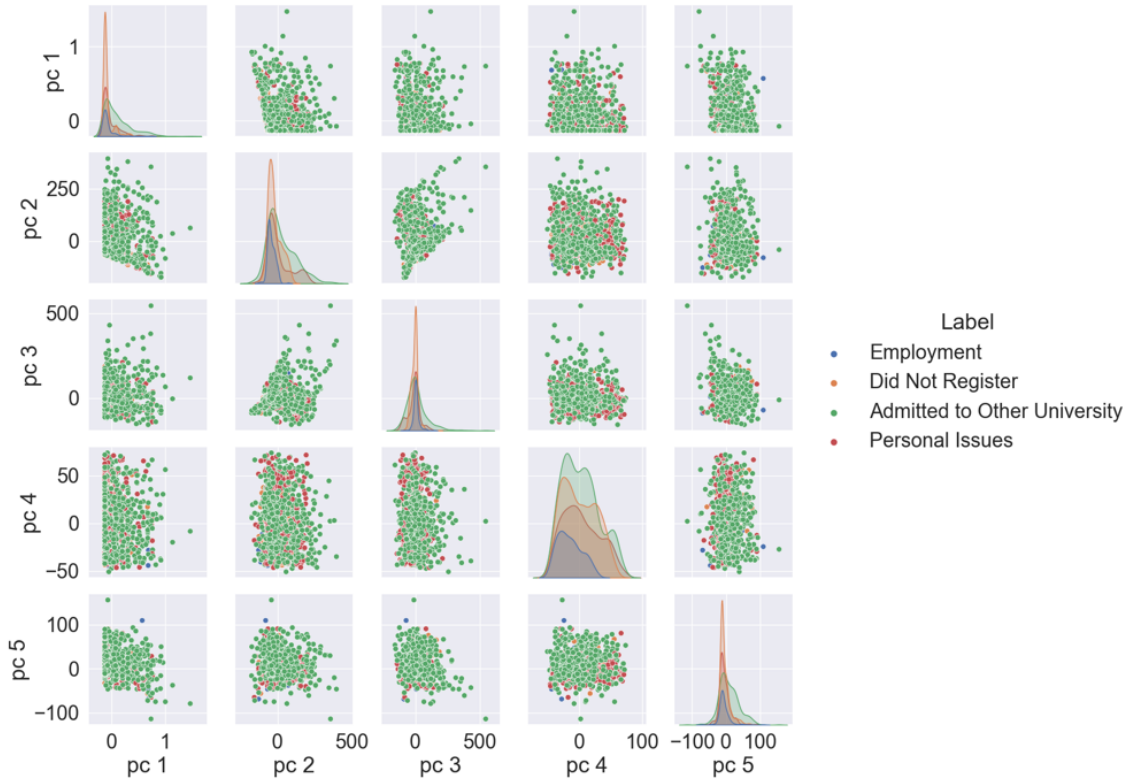

Figure 6. Scatter Plot of Top 5 Principal Components of PCA.

We used the PCA-compressed results as input to the K-means clustering algorithm to cluster the data by reason. We randomly selected the initial value of the cluster's center and used 80% of the data for training and 20% for validation with 10-fold cross-validation. We used multiple rounds of K-means with $K = 2$ to identify the reasons. By using K = 2 in the K-means, we could simplify the modeling process and improve computational efficiency. We created two groups for each reason and determined if a value was included in the corresponding group. For instance, we created the categories of "Employment" and "Other Reasons" and assigned a value of True or False depending on whether they were included. In the second round, we ran another round of K-means with the labels "Did not register" and "Other Reasons". In this way, we can understand the intention of students who have chosen multiple reasons for dropping out. After the four rounds of K-means, we identified the main reason for dropping out based on the resulting cluster. Dropout precision is crucial in this experiment because the SDP system first selects dropout students with a hybrid model and then conducts "classification by reason" for dropout. The "Employment" category showed the lowest accuracy at 0.238 because the volume of data was too small. However, we are not too concerned about the "Employment" category because these students successfully began professional careers. The "Personal Issues" category was 0.405, which did not show high dropout precision for various reasons. The precision of

"Admission to Other Universities" and "Did Not Register" was 0.672 and 0.569, respectively. We present the experiment results for each dropout reason in Table 15.

Table 15. Dropout Student K-means Clustering Classification.

Reason	Students	Accuracy	Dropout Precision	Dropout Recall	Dropout F1
Employment	180	0.848	0.238	0.161	0.192
Did Not Register	540	0.722	0.569	0.471	0.516
Personal Issues	514	0.659	0.405	0.268	0.323
Admission to Other Universities	608	0.776	0.672	0.512	0.581

6. Discussion

Since the world is moving towards personalized services, universities are also striving to provide personalized services to students. Administration and Consultation are two areas that need meticulous relationship management with the students. Until now, the universities have reacted passively to the voices or actions of the students. However, it is now required to act upon the hidden needs of the students proactively. There are many ways universities can use student data; in this paper, we focus on dropout rates of students, which are becoming a severe issue in South Korea. Once the administrators identify the students who are having trouble in academics or are about to drop out of the university, they can proactively support the students by making appointments with a counselor and other professionals. To make this happen, we can provide the risk level of the students. Using Table 7, we identified the risk levels of students in the year 2020, AS shown in Table 16. The priority order is academic year, then high-risk students, followed by low-risk students. No-risk students have the lowest priority. According to the table, 143 first-year students have the highest priority, and counselors must take action immediately upon acknowledgment of the risk group of the students. Depending on the classification group associated with a student, the counselor can decide which topic to discuss with the student. We believe such approaches are critical to the administrators and the students; however, the universities did not find the need to act upon the issue. We believe that universities can provide better student services by using the methods proposed in this paper.

Table 16. Risk Analysis of the Students.

Academic Year	High-Risk Students	Low-Risk Students	No-Risk Students	Total
Freshman	143	181	6406	6873
Sophomore	153	216	6412	6938
Junior	120	137	6768	7145

7. Conclusions

In this paper, we present the SDP system with high precision and recall rates to accurately predict students at risk of dropping out of universities. This system is intended to assist administrators and counselors in providing personalized support to these students. The student records used in the system were asymmetrical, so various imbalanced data processing techniques were employed to determine the best algorithm for the data. Fifteen different sampling algorithms were tested with four different prediction algorithms. The SDP system employs XGBoost with SMOTE and CatBoost with RandomOverSamplerSMOTEENN to improve the precision and recall of the predictions. The results of these two algorithms were then compared with those of four existing algorithms: logistic regression, artificial neural networks, gradient boosting, and an ensemble method. The SDP system achieved the highest scores in dropout F1. Its scores were 0.989 for accuracy, 0.819 for precision, 0.755 for recall, and 0.786 for F1. In addition, K-means clustering was used to

classify the reasons for dropping out and identify the risk levels of the students, allowing administrators and counselors to provide more targeted support.

Author Contributions: Conceptualization, S.L.; methodology, S.K.; validation, S.K. and S.L.; formal analysis, S.K.; investigation, S.K.; data curation, S.K.; writing—original draft preparation, S.K.; writing—review and editing, E.C. and S.L.; visualization, S.K. and S.L.; supervision, Y.-K.J., S.L.; project administration, E.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. NRF2021R1A2C1014163).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

GB	Gradient Boosting
Ensemble	Ensemble of Gradient Boosting, Random Forest, and Support Vector Machine
LR	Logistic Regression
ANNs	Artificial Neural Networks
None	Unprocessed
Under	Undersampling
Over	Oversampling
Combine	Undersampling and Oversampling
SDP	Student Dropout Prediction System
IDP	Imbalanced Data Processor
DP	Dropout Predictor
DC	Dropout Reason Classifier
RSMOTEENN	RandomOverSamplerSMOTEEN

References

1. South Korea's Basic Education Statistics. Available online: https://kess.kedi.re.kr/index (accessed on 15 December 2022).
2. Park, H.S. An Analysis of the Factors Affecting Local College Freshmen's Intention of Dropout: Focused on C-College. *J. Learn.-Cent. Curric. Instr.* **2017**, *17*, 423–442. [CrossRef]
3. Barros, T.M.; Souza Neto, P.A.; Silva, I.; Guedes, L.A. Predictive Models for Imbalanced Data: A School Dropout Perspective. *Educ. Sci.* **2019**, *9*, 4–275. [CrossRef]
4. Baranyi, M.; Nagy, M.; Molontay, R. Interpretable deep learning for university dropout prediction. In Proceedings of the 21st Annual Conference on Information Technology Education, Omaha, NE, USA, 7–9 October 2020; pp. 13–19.
5. Nurdaulet, S.; Alibek, O.; Yershat, S.; Shirali, K. Predicting student drop-out in higher institution using data mining techniques. *Phys. Conf.* **2020**, *1496*, 012005.
6. Shynarbek, N.; Orynbassar, A.; Sapazhanov, Y.; Kadyrov, S. Prediction of Student's Dropout from a University Program. In Proceedings of the 16th International Conference on Electronics Computer and Computation (ICECCO), Kaskelen, Kazakhstan, 25–26 November 2021; pp. 1–4.
7. da Silva, M.; Diogo, E.; Solteiro, P.; Eduardo, J.; Arsénio, R.; de Moura, O.; Paulo, B.; Barroso, J. Forecasting Students Dropout: A UTAD University Study. *Future Internet* **2022**, *14*, 3–76.
8. Ernández-García, A.J.; Preciado, J.C.; Melchor, F.; Rodriguez-Echeverria, R.; Conejero, J.M.; Sánchez-Figueroa, F. A real-life machine learning experience for predicting university dropout at different stages using academic data. *IEEE Access* **2021**, *9*, 133076–133090. [CrossRef]
9. Lee, S.; Chung, J.Y. The machine learning-based dropout early warning system for improving the performance of dropout prediction. *Educ. Sci.* **2019**, *9*, 3093. [CrossRef]
10. Niyogisubizo, J.; Liao, L.; Nziyumva, E.; Murwanashyaka, E.; Nshimyumukiza, P.C. Predicting student's dropout in university classes using two-layer ensemble machine learning approach: A novel stacked generalization. *Comput. Educ. Artif. Intell.* **2022**, *3*, 100066. [CrossRef]
11. Johnson, J.M.; Khoshgoftaar, T.M. Survey on deep learning with class imbalance. *J. Big Data* **2019**, *6*, 1–54. [CrossRef]
12. Tomek, I. Two modifications of CNN. *IEEE Trans. Syst. Man Cybern. Syst.* **1976**, *14*, 769–772.

13. Imbalanced Learn. Available online: https://imbalanced-learn.org/stable/references/generated/ (accessed on 15 December 2022).
14. Wilson, D.L. Asymptotic properties of nearest neighbor rules using edited data. *IEEE Trans. Syst. Man Cybern. Syst.* **1972**, *3*, 408–421. [CrossRef]
15. Chawla, N.V.; Bowyer, K.W.; Hall, L.O.; Kegelmeyer, W.P. SMOTE: Synthetic minority over-sampling technique. *Artif. Intell. Res.* **2002**, *16*, 321–357. [CrossRef]
16. Han, H.; Wang, W.-Y.; Mao, B.-H. Borderline-SMOTE: A new over-sampling method in imbalanced data sets learning. In Proceedings of the IEEE 2005 International Conference on Advances in Intelligent Computing, Hefei, China, 23–26 August 2005; Volume 16, pp. 878–887.
17. He, H.; Bai, Y.; Garcia, E.A.; Li, S. ADASYN: Adaptive synthetic sampling approach for imbalanced learning. In Proceedings of the 2008 IEEE International Joint Conference on Neural Networks, Hong Kong, China, 1–8 June 2008; pp. 1322–1328.
18. Batista, G.E.A.P.A.; Prati, R.C.; Monard, M.C. A study of the behavior of several methods for balancing machine learning training data. *SIGKDD Explor. Newsl.* **2004**, *1*, 20–29. [CrossRef]
19. Gustavo, E.A.P.A.; Batista, A.; Bazzan, M.C. Monard Balancing Training Data for Automated Annotation of Keywords: A Case Study. In Proceedings of the WOB, Macaé, RJ, Brazil, 3–5 December 2003; pp. 10–18.
20. dmlc XGBoost. Available online: https://xgboost.readthedocs.io/en/stable/ (accessed on 15 December 2022).
21. CatBoost. Available online: https://catboost.ai/ (accessed on 15 December 2022).
22. Laurikkala, J. Improving identification of difficult small classes by balancing class distribution. *Artif. Intell. Med.* **2001**, *35*, 63–66.
23. Tomek, I. An experiment with the edited nearest-nieghbor rule. *IEEE Trans. Syst. Man Cybern.* **1976**, *SMC-6*, 448–452.
24. ClusterCentroids, Imbalanced-Learn, Accessed 0808, 2022. Available online: https://imbalanced-learn.org/stable/references/generated/imblearn.under_sampling.ClusterCentroids.html (accessed on 15 December 2022).
25. Menardi, G.; Torelli, N. Training and assessing classification rules with imbalanced data. *Data Min. Knowl. Disc. Data Min. Knowl. Disc.* **2014**, *28*, 1, 92–122. [CrossRef]
26. Nguyen, H.M.; Cooper, E.W.; Kamei, K. Borderline over-sampling for imbalanced data classification. *Int. J. Knowl. Eng. Soft Data Parad.* **2009**, *3*, 24–29. [CrossRef]