

Article

Balancing Dropout Candidate Coverage and Counseling Burden in University Student Dropout Prediction

Kwan Woo Kim ¹, Cheolgi Kim ¹ and Hyeon Gyu Kim ^{2,*}¹ Department of Computer Engineering, Korea Aerospace University, Goyang 10540, Republic of Korea; westact@kau.kr (K.W.K.)² Department of VR Convergence Engineering, Duksung Women's University, Seoul 01369, Republic of Korea

* Correspondence: hgkim@duksung.ac.kr

Abstract

Many universities have developed machine learning models for student dropout prediction, and these models are commonly evaluated using accuracy-oriented metrics such as the F1 score. However, the model with the highest F1 score does not necessarily provide the most useful dropout candidate list when the institutional objective is to identify actual dropout students under limited counseling resources. This study investigates the trade-off between dropout candidate coverage and counseling burden in university student dropout prediction. We discuss how to determine the target coverage and model-specific classification thresholds required to find a solution model that provides a balanced candidate list. The proposed method uses the model with the highest recall to determine the target coverage and selects the model with the highest precision after threshold adjustment to provide a smaller candidate list under the target coverage. To reduce the risk of test-set-based threshold optimization, the target coverage, model-specific thresholds, and final solution model are determined using only the training/validation data, while the test dataset is reserved strictly for final evaluation. The method was validated using 28 candidate models implemented with various machine learning algorithms and sampling settings. In the main experimental split, the Light Gradient Boosting Machine model with the Synthetic Minority Oversampling Technique provided a candidate list with a dropout candidate coverage of 0.939 and a candidate-list size of 569. In repeated-seed analysis, the proposed method maintained dropout candidate coverage comparable to the threshold-tuned highest-F1 baseline while reducing false positives and candidate-list size. The results suggest that the proposed method provides a practical model selection strategy for improving candidate-list efficiency while reducing the risk of test-set-based threshold optimization.

Keywords: classification threshold; counseling burden; dropout candidate coverage; model selection; student dropout prediction



Academic Editors: Janos Botzheim
and Zhineng Chen

Received: 1 April 2026

Revised: 18 June 2026

Accepted: 26 June 2026

Published: 2 July 2026

Copyright: © 2026 by the authors.
Licensee MDPI, Basel, Switzerland.
This article is an open access article
distributed under the terms and
conditions of the [Creative Commons
Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

Student dropout has been recognized as one of the most complex and important issues in higher education systems [1–3]. Students who leave university without completing a degree may face limited employment opportunities and lower income prospects. From a social perspective, student dropout may increase costs related to re-employment training, unemployment support, and other welfare programs. From a university perspective, dropout can also lead to direct financial losses and reduced educational sustainability. To address this issue, many studies have developed machine learning models for predicting

student performance, graduation, or dropout in higher-education and technology-mediated learning environments [4–6].

In practical university settings, dropout prediction models are often used to generate candidate lists of students who may require academic support or counseling. Therefore, model evaluation should not be limited to prediction accuracy alone. A useful candidate list should include a sufficient number of actual dropout students while remaining manageable for institutional intervention. This requirement creates a practical trade-off between dropout candidate coverage and counseling burden.

Most existing studies have selected the best model using accuracy-oriented metrics such as accuracy, precision, recall, or the F1 score. Although these metrics are useful for evaluating classification performance, they do not directly indicate whether the resulting candidate list is suitable for intervention planning. For example, a model with the highest F1 score may produce a compact and accurate candidate list, but it may exclude some actual dropout students who could have been identified by a slightly larger list. Conversely, a model with higher recall may include more actual dropout students, but it may also increase false positives and counseling burden.

This study addresses this issue by formulating student dropout prediction as a candidate-list construction problem under limited institutional resources. Rather than assuming that prediction results directly lead to dropout reduction, we focus on whether actual dropout students are included in the candidate list. In this setting, true positives represent actual dropout students covered by the candidate list, while false positives contribute to additional counseling burden. The objective is therefore not to maximize a single performance metric, but to select a model and decision threshold that provides a favorable balance between the candidate coverage and the list size.

To achieve this, we first show that given a set of candidate models, a model with the highest recall maximizes the dropout candidate coverage. Based on this observation, the target coverage is determined using the model. We then adjust the classification thresholds of other models so that they satisfy the target value. To reduce the risk of test-set-based threshold optimization, threshold adjustment is performed using only the training and validation datasets, without using the test dataset. After threshold adjustment on the validation folds, the candidate models are evaluated under the same or similar target coverage conditions. In this case, the model with the highest precision minimizes false positives, i.e., the size of the candidate list. Therefore, we select the model as a solution to provide a balanced candidate list. The method was evaluated using 28 candidate models implemented with various machine learning algorithms and sampling settings. As learning data, academic records collected from 20,050 students at a four-year university in Seoul, Republic of Korea, were used.

The contributions of this paper can be summarized as follows:

- We review existing studies on university-level student dropout prediction and discuss their common reliance on accuracy-oriented evaluation metrics such as the F1 score.
- We formulate the practical trade-off between the dropout candidate coverage and counseling burden, emphasizing that the model with the highest F1 score alone does not necessarily provide the most useful candidate list for institutional intervention.
- We present a method to determine the target coverage and model-specific classification thresholds to find a solution model that provides a balanced candidate list. The target coverage and thresholds are determined using only the training and validation datasets, while the test dataset is reserved strictly for final evaluation.
- The method was evaluated using 28 candidate models implemented with various machine learning algorithms and sampling settings. For comparison, the solution

model was evaluated against two reference models: the default highest-F1 model and the validation-derived threshold-tuned highest-F1 model.

- We provide additional analyses, including validation-based hyperparameter tuning, repeated-seed robustness analysis, SHAP-based interpretation of the solution model, and Jaccard similarity analysis of selected candidate lists, to examine robustness, interpretability, and candidate-list stability of the proposed method.

The remainder of this paper is organized as follows. Section 2 reviews related work. Section 3 presents the materials and methods, including the problem formulation, model selection procedure, candidate models, dataset, and evaluation protocol. Section 4 reports the experimental results. Section 5 discusses the findings, implications, and limitations of the proposed method. Section 6 concludes the paper.

2. Related Work

Student Dropout Prediction (SDP) has been widely studied as an important application of machine learning in education, aiming to identify students at risk of dropout and support timely planning for interventions [2]. Previous studies have used institutional administrative and academic records, including student registration information, personal and family background variables, academic performance indicators, and grade records, to estimate students' risk of university dropout [7]. Depending on the prediction target and application context, student dropout prediction studies can be broadly categorized into course-level dropout prediction and university-level dropout prediction. Course-level dropout prediction focuses on predicting withdrawal from a specific course or online learning program and has been actively investigated in MOOC settings, where course-level learning behavior logs and public benchmark datasets are relatively accessible [8,9]. By contrast, university-level dropout prediction aims to estimate whether a student will eventually leave the university, using institutional administrative and academic records such as registration information, personal and family background variables, academic performance indicators, and grade records [7].

University-level dropout prediction often relies on institution-specific administrative and academic records, such as enrollment information, academic records, grades, attendance, scholarship records, tuition-related information, and counseling-related records [10–12]. Because these data are generated and managed within individual universities, data availability and privacy considerations become important issues in this setting [10]. Nevertheless, university-level dropout prediction is practically important because identifying potential dropout students can support retention policies, student counseling, academic assistance, and the efficient allocation of institutional resources [10,12].

Existing studies on higher-education decision support and university-level dropout prediction have applied a wide range of machine learning and deep learning algorithms to institutional academic and administrative data. These studies have demonstrated that university-held data can be used to support decision-making and identify students at risk of dropout. However, as summarized in Table 1, prior studies have mainly focused on improving predictive performance, handling class imbalance, modeling semester-level records, or enhancing explainability. Relatively few studies have explicitly formulated dropout prediction as an operational candidate-list construction problem that jointly considers dropout candidate coverage and counseling burden under limited institutional resources.

Table 1. Summary of prior university-level dropout prediction studies from an operational intervention perspective.

Ref. No.	Main Focus	Representative Methods	Candidate-List/Counseling Consideration
[13,14]	Machine-learning-based decision support and dropout-related prediction using institutional higher-education records	LR, DT, RF	These studies used institutional student data for higher-education decision support or dropout-related prediction, but candidate-list size or counseling workload was not explicitly controlled.
[15]	Deep learning-based dropout prediction using university administrative data	CNN, Bayesian Network	Focused on prediction accuracy; intervention burden was not directly modeled.
[16]	Stage-wise dropout prediction from pre-enrollment to later semesters	GB, RF, SVM, ensemble	Supported staged decision-making, but target coverage or list-size constraints were not formulated.
[11,12]	Imbalance-aware university dropout prediction using academic and administrative records	XGBoost, CatBoost, LightGBM, resampling	Considered precision/recall and counseling-related information, but did not formulate candidate-list size as an explicit optimization objective.
[17,18]	Semester-record modeling and temporal-pattern exploitation	LightGBM, RNN/GRU, attention	Focused on feature construction or temporal patterns; counseling burden was outside the main scope.
[19]	Explainable and fair dropout prediction	RF, Feature Tokenizer Transformer, XAI	Emphasized explainability and fairness, but not candidate-list construction under resource limits.
[20]	Retention-oriented dropout prediction in a private higher education institution	LR, DT, ANN ensemble	Aimed to support retention actions, but threshold-based counseling list construction was not explicitly defined.

Class imbalance is an important evaluation issue in student dropout prediction. In many educational settings, dropout students account for only a small proportion of the total student population, making the dropout class the minority class. In such cases, accuracy alone can be misleading because a model may appear to perform well by correctly classifying the majority class while failing to identify actual dropout students. Prior studies have therefore used evaluation metrics, such as precision, recall, F1-score, ROC curves, and precision–recall curves, or have applied data-level imbalance handling methods such as SMOTE, ADASYN, SMOTE + Tomek, and SMOTE + ENN to mitigate the class imbalance problem [11,12,16,21].

Barros et al. further showed that, under imbalanced educational data, performance interpretation can be sensitive to the label setting and evaluation perspective. Their study discusses the “accuracy paradox,” where a high accuracy value does not necessarily indicate a high-quality model when the minority class is not properly identified [22]. In a university-level setting, Kim et al. proposed an SDP system that combines an XGBoost model trained with SMOTE for high dropout precision and a CatBoost model trained with a SMOTEENN-based resampling strategy for high dropout recall [12]. Cho et al. examined SMOTE, ADASYN, and Borderline-SMOTE and reported that oversampling did not consistently improve F1-score across models [11]. These studies contributed to improving classification performance under class imbalance, but they did not explicitly formulate the counseling burden that arises when prediction results are used to construct a counseling candidate list.

In addition, the practical outcome of a dropout prediction model depends heavily on the classification threshold. Most predictive models produce a dropout probability or risk score for each student, and students whose scores exceed a given threshold are classified as dropout candidates. Lowering the threshold can include more actual dropout students

in the candidate list, thereby increasing recall and true positives. However, it can also increase false positives and expand the number of students selected for counseling, which may impose additional burdens on limited counseling staff and administrative resources. Conversely, raising the threshold can reduce the candidate-list size, but it increases the risk of excluding students who may require intervention. Therefore, threshold selection is not merely a post-processing step; rather, it determines the trade-off between dropout candidate coverage and counseling burden.

Classification threshold selection has also been discussed as an important issue in cost-sensitive learning and operational classification. Elkan argued that different types of misclassification errors can incur different costs and that optimal decisions should be made by minimizing expected cost based on class probability estimates rather than relying on a fixed default threshold [23]. Similarly, Provost and Fawcett noted that, in real-world environments, class distributions, misclassification costs, and target operating conditions may be imprecise or subject to change. They further discussed classifier evaluation in relation to various operational metrics, including accuracy, expected cost, precision, recall, and workforce utilization [24]. These studies support the view that threshold selection is not merely a post-processing step but an important procedure for adapting model outputs to operational constraints.

In university dropout prediction, a predictive model is not only used to classify students as dropouts or non-dropouts but can also support the construction of a candidate list of students who may require counseling or academic assistance. Kim et al. explicitly connected dropout prediction to student counseling and proposed an SDP system that aims to improve dropout precision and recall for university-level intervention [12]. From this perspective, true positives correspond to actual dropout students included in the candidate list, which can be interpreted as dropout candidate coverage. In contrast, false positives correspond to students who are selected as counseling candidates but do not eventually drop out, thereby increasing the counseling burden. Therefore, a practically useful dropout prediction model should not be evaluated only by a single classification metric such as F1-score. Instead, it should include a sufficient number of actual dropout students while keeping the candidate-list size within a manageable range for institutional counseling resources.

In educational data mining, several studies have used prediction outputs to prioritize students under limited intervention resources. Lakkaraju et al. proposed a machine learning framework for identifying students at risk of adverse academic outcomes and emphasized the need to rank students by risk estimates when schools can intervene with only a limited number of students [25]. Aguiar et al. also proposed a prioritization approach for students at risk of not graduating high school on time, focusing on who should be selected, when they should be supported, and why they are at risk [26]. In addition, Purdue University's Course Signals demonstrated how learning analytics-based risk indicators can be connected to early intervention actions such as feedback messages, reminders, referrals to academic advisors or resource centers, and face-to-face meetings [27]. Early warning system studies have similarly focused on identifying at-risk students early enough to support timely intervention [21,28].

However, these studies mainly focus on early warning, student prioritization, course-level intervention, or high-school-level intervention. They do not directly address the problem of selecting a university-level dropout prediction model by jointly considering dropout candidate coverage and candidate-list size under validation-derived operating conditions. Existing university-level dropout prediction studies have generally compared and selected models using classification-oriented metrics such as accuracy, precision, recall, F1-score, sensitivity, or AUC [10,12]. These metrics are useful for evaluating predictive

performance, but they do not directly answer an operational question faced by universities: how many students should be selected as counseling candidates while maintaining sufficient coverage of actual dropout students?

A model with the highest F1-score may provide a balanced trade-off between precision and recall, but it may not necessarily generate the most appropriate candidate list under limited counseling resources. Conversely, a high-recall model may include more actual dropout students, but it can also increase false positives and expand the counseling workload. Therefore, the highest-F1 model is not always the most useful model for counseling-oriented dropout intervention. This gap motivates the need to reformulate university-level dropout prediction as a candidate-list construction problem rather than a simple classification performance maximization problem.

To address this gap, this study proposes a validation-derived model selection framework that balances dropout candidate coverage and counseling burden. Instead of selecting a model solely based on the highest F1-score, the proposed framework derives target coverage and model-specific thresholds from the training and validation data and then selects a solution model that can reduce candidate-list size under the same or similar coverage conditions. In this respect, the present study extends prior university-level dropout prediction research by explicitly considering candidate-list construction for institutional intervention planning.

3. Materials and Methods

This section describes the formulation of dropout candidate-list construction, the proposed validation-derived model selection procedure, the candidate models, the dataset, and the experimental protocol.

3.1. Background

Student dropout prediction (SDP) can be formulated as a binary classification problem. Let S denote the given input data, which is divided into a training/validation dataset S_T and a test/evaluation dataset S_E . The former is used for model training, classification threshold estimation, and final model selection, while the latter is used only once for final performance evaluation. For each sample $s_i \in S$, let $t_i \in \{0, 1\}$ be the ground-truth label, where $t_i = 1$ indicates a dropout student, and $t_i = 0$ otherwise. A model M estimates a dropout probability $p_i \in \mathbb{R}$ for $s_i \in S_E$. Given a classification threshold $\tau \in \mathbb{R}$, the predicted label y_i is assigned as 1 if $p_i \geq \tau$, and 0 otherwise.

The main symbols used in this paper are summarized in Appendix A.1. To evaluate the classification performance of a model on a given evaluation dataset, the confusion matrix consisting of TP , FP , FN , and TN is commonly used. TP is the number of true positive samples whose labels are predicted as positive ($y_i = 1$) and are actually positive ($t_i = 1$). FP is the number of false positive samples whose labels are predicted as positive ($y_i = 1$) but are actually negative ($t_i = 0$). In SDP, dropout students are treated as the positive class. Thus, TP represents the number of students who were predicted as dropout candidates and actually dropped out. Similarly, FP represents the number of students who were predicted as dropout candidates but did not drop out. TN and FN can be interpreted in the same manner. Then, the total number of samples in a given evaluation dataset can be expressed as

$$N = TP + FP + FN + TN. \quad (1)$$

The number of samples belonging to the positive and negative classes, denoted as N_P and N_N , respectively, can be represented as follows:

$$N_P = TP + FN, \quad (2)$$

$$N_N = TN + FP. \quad (3)$$

Accuracy is one of the most widely used metrics and is defined as the number of correctly predicted samples divided by the total number of samples:

$$\text{accuracy} = \frac{TP + TN}{N}. \quad (4)$$

However, accuracy can be misleading when the data are highly imbalanced. Consider a dataset with a dropout rate of 5%. A classifier that predicts all samples as non-dropout can still achieve 95% accuracy, although it fails to identify any dropout students. Conversely, a classifier that predicts all samples as dropout would achieve only 5% accuracy. This example shows that FP and FN should be considered together when evaluating model performance.

Precision reflects the reliability of positive predictions and is sensitive to false positives. It is defined as the proportion of positive predictions that are actually correct:

$$\text{precision} = \frac{TP}{TP + FP}. \quad (5)$$

Recall reflects the proportion of actual positive samples that are correctly identified and is therefore sensitive to false negatives. It is also called the true positive rate (TPR):

$$\text{recall} = \frac{TP}{N_p} = \frac{TP}{TP + FN}. \quad (6)$$

To jointly consider precision and recall, the F1 score is commonly used. It is defined as the harmonic mean of precision and recall:

$$F_1 = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}} = \frac{2TP}{2TP + FP + FN}. \quad (7)$$

As discussed in Section 2, dropout prediction studies often involve skewed class distributions and therefore commonly use the F1 score for model evaluation. However, as discussed in this paper, the F1 score alone may not fully reflect the usefulness of the candidate list for institutional intervention.

3.2. Problem Formulation

This study aims to identify a dropout prediction model that provides a practically useful candidate list for institutional intervention. Unlike conventional model selection based primarily on accuracy-oriented metrics, the proposed formulation considers two aspects simultaneously: dropout candidate coverage and candidate-list size. Dropout candidate coverage reflects how many actual dropout students are included in the candidate list, whereas the candidate-list size represents the number of students who may require counseling or academic support.

No counseling intervention was conducted in this study. Counseling burden refers to the expected institutional workload implied by the size of the predicted dropout candidate list.

Because this study does not include post-intervention counseling outcome data, we do not directly estimate the actual reduction in student dropouts. Instead, we focus on whether actual dropout students are included in the candidate list generated by the model. Under this setting, the number of true positives, TP , represents the number of actual dropout students covered by the candidate list. Therefore, increasing TP improves dropout candidate coverage, but it may also increase FP and consequently enlarge the candidate list.

Accordingly, the practical objective is to achieve a favorable trade-off between TP and $TP + FP$. A larger TP indicates that more actual dropout students are covered by the candidate list, whereas a smaller $TP + FP$ indicates a lower counseling burden. Since increasing TP often requires lowering the classification threshold, it may also increase FP . Therefore, the problem is not to maximize a single metric but to select a model and threshold that provide a suitable balance between coverage and burden.

Let U denote a set of candidate models, where each model M_j produces a dropout probability $p_{i,j}$ for each sample s_i . Given a classification threshold τ_j , s_i is included in the candidate list L_j if $p_{i,j} \geq \tau_j$. The following observations clarify the relationship among TP , recall, precision, and the candidate-list size.

Observation 1. *For a given evaluation dataset, TP and recall are proportional because the number of actual positive samples is fixed.*

Proof. Let N_p denote the number of actual positive samples in the dataset. Then, recall is defined as TP/N_p . Since N_p is fixed for the dataset, recall increases as TP increases, and vice versa. Therefore, TP and recall are proportional on the same evaluation dataset. $\square$

Observation 2. *When two candidate lists cover the same number of actual positive samples, the list with higher precision has a smaller candidate-list size.*

Proof. Precision is defined as $TP/(TP + FP)$, where $TP + FP$ is the candidate-list size. If two models have the same TP , a higher precision implies a smaller denominator, $TP + FP$. Therefore, among the candidate lists with the same TP , the list with higher precision is always smaller. $\square$

Observation 1 indicates that increasing TP is equivalent to increasing recall on the same evaluation dataset, which corresponds to improving dropout candidate coverage. Observation 2 indicates that, when the same TP or coverage is considered, precision can be used to identify a model that provides a smaller candidate list. These two observations motivate the proposed model selection scheme. In what follows, we discuss how to select a solution model that provides a balanced dropout candidate list.

3.3. Model Selection Procedure

The basic idea of the proposed method is simple. Let M_ρ denote the model with the highest recall. From Observation 1, we set the recall of M_ρ as a target value to maximize the dropout candidate coverage. We then adjust the classification thresholds of other models to satisfy the target coverage. After the threshold adjustment, the recall of all models becomes similar or nearly the same. This implies that all models provide the same or similar TP if they are evaluated on the same dataset. In this case, from Observation 2, the model with the highest precision minimizes FP . Therefore, we finally select the model as a solution and produce the dropout candidate list using the model.

The target candidate coverage and classification thresholds are determined during the training/validation stage. To obtain a stable and robust model, k -fold cross-validation was applied in this stage, and the calculation of the two factors is influenced by this process. For example, the target candidate coverage, denoted as ρ^* , is calculated as an average of k validation results. Since k -fold cross-validation is applied, the training/validation dataset S_T is divided into the k equal-sized subsets. Then, the recall of model M_j is calculated as an average of the k recall values obtained from the k subsets. After the average recall of all models is obtained, the highest value is selected as ρ^* .

The derivation of classification thresholds is also affected by the cross-validation process. To find the threshold satisfying ρ^* in M_j , we first calculate TP corresponding to ρ^* . Since each subset in S_T may have a different number of positive samples, the TP is calculated differently for each subset. Let the TP of M_j for the k -th subset be TP_j^k . Then, TP_j^k can be calculated as $\rho^* \times N_p^k$, where N_p^k is the number of positive samples in the k -th subset. After obtaining TP_j^k , a scanning process begins to find the threshold corresponding to TP_j^k . The validation samples in S_T are sorted by predicted dropout scores. If the cumulative number of actual dropout students reaches TP_j^k at a certain score, the scanning process is stopped and the score is selected as the threshold for the k -th fold, τ_j^k . Once the process for all subsets is complete, the average of the k thresholds is calculated, and this is set as a new threshold for M_j .

Since the classification threshold of the model has been updated, its performance is recalculated on S_T . After the performance of all models is updated with their new thresholds, the model with the highest precision is selected as the final solution, M_ω . Figure 1 summarizes the model selection procedure in the proposed method. For each step from 1 to 4, five-fold cross-validation was applied. The hyperparameter tuning step is discussed in Section 3.4.

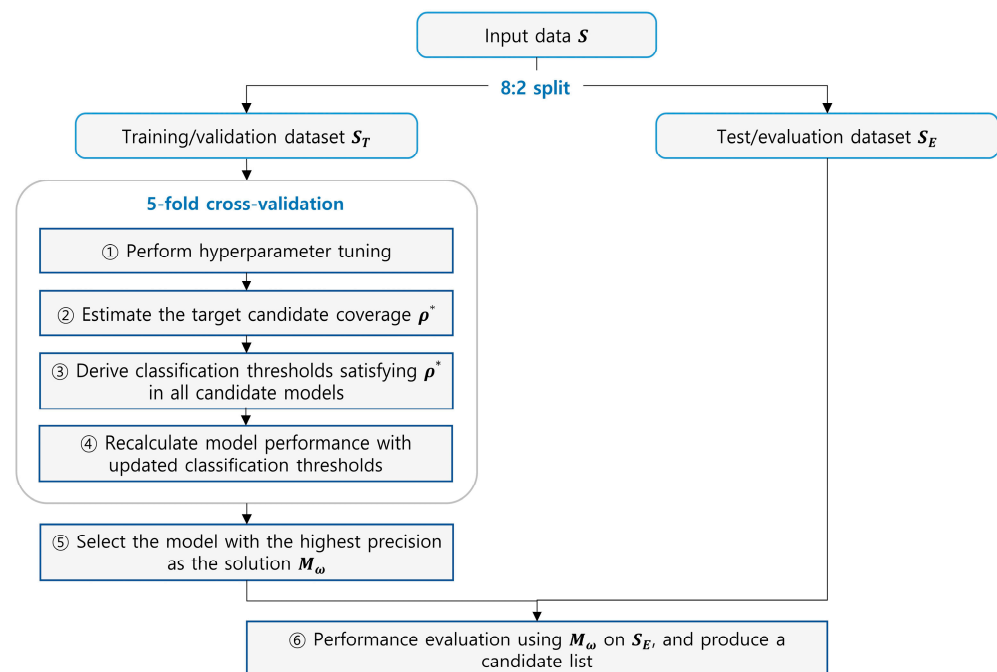


Figure 1. Model selection procedure in the proposed method.

3.4. Candidate Models and Implementation

To construct the candidate model set U , we combined seven machine learning algorithms with four sampling settings. The machine learning algorithms included the decision tree (DT), support vector machine (SVM), deep neural network (DNN), random forest (RF), XGBoost (XGB), CatBoost (CB), and LightGBM (LGBM). The sampling settings consisted of the original imbalanced data, SMOTE [29], Borderline-SMOTE (BL-SMOTE) [30], and ADASYN [31]. Thus, a total of 28 candidate models were implemented and evaluated under the proposed method.

Hyperparameter tuning was performed only within the training/validation stage. Specifically, a limited validation-based grid search was conducted for DT, RF, DNN, XGB, CB, and LGBM. The test dataset was not used during the hyperparameter tuning. For each model, candidate hyperparameter combinations were evaluated using five-fold stratified

validation, and the configuration with the highest mean validation F1 score at the default threshold of 0.5 was selected.

A limited grid search was used instead of an exhaustive, large-scale search to maintain a feasible computational cost while reducing the dependence of the results on arbitrary default hyperparameters. SVM was not included in the hyperparameter search because of its computational cost in the current experimental setting. Therefore, SVM was implemented using LinearSVC with fixed settings. Since LinearSVC does not provide calibrated probability estimates, its decision_function output was transformed into a monotonic score in the range of 0 to 1 and used for threshold-based candidate-list construction.

The hyperparameter search space and selected settings used in the experiments are provided in Appendix A.2. When the selected settings differed across sampling settings, the selected values are reported separately. When the same value was selected for all sampling settings, a single value was reported.

The tree-based and boosting models were implemented using standard Python libraries, including DecisionTreeClassifier, RandomForestClassifier, XGBClassifier, CatBoostClassifier, and LGBMClassifier. The DNN model was implemented using Dense layers with ReLU activation in the hidden layers and a sigmoid activation in the output layer. The SVM model was implemented using LinearSVC with fixed settings, and its decision_function output was used as a monotonic score for threshold-based candidate-list construction. All candidate models were trained and evaluated within the same validation-derived method. The experiments were conducted using Python 3.12.3, scikit-learn 1.8.0, imbalanced-learn 0.14.1, XGBoost 3.2.0, CatBoost 1.2.10, LightGBM 4.6.0, TensorFlow 2.21.0, Keras 3.14.1, SHAP 0.51.0, NumPy 2.4.4, and pandas 3.0.3.

3.5. Dataset

To validate the proposed method, we used academic records collected from a four-year university in Seoul, Republic of Korea. The original dataset contained 20,050 student records. After preprocessing and excluding records with missing or invalid values, 19,717 records were used in the experiments. Among these records, 2524 were dropout students, resulting in a dropout rate of approximately 12.80%.

The dataset was divided into a training/validation dataset and a test/evaluation dataset with a ratio of 8:2. The former contained 15,773 records, including 2019 dropout students. The latter contained 3944 records. The actual number of dropout students in the test dataset was 505; this value was used only for final performance evaluation and was not used to determine the target candidate coverage, classification thresholds, or final model selection.

The input data consisted of student-level variables and semester-level variables. Student-level variables were used directly as input features. Semester-level variables were expanded into fixed-length semester-wise features after arranging each student's semester records in reverse chronological order. For example, the semester-level variable GPA was expanded into GPA_Sem1 to GPA_Sem8, where Sem1 represents the most recent semester position and Sem8 represents the oldest retained semester position. This representation enables the candidate models to use semester-wise academic information while maintaining a fixed input dimension. In total, the final input contained 92 features, consisting of 4 student-level features and 88 semester-level features. The feature names listed in Table 2 are the same as those used in the SHAP analysis in Section 4.4. The target variable was the final student status, where dropout students were labeled as 1 and non-dropout students were labeled as 0. Table 2 summarizes the input variables used for model training.

Table 2. Input variables used for model training.

Group	Variable Name	Description	Expanded Feature Names
Student-level	Transfer	Whether the student entered as a transfer student	Transfer
	NumUnregisteredSemesters	Number of unregistered semesters	NumUnregisteredSemesters
	NumSem	Number of enrolled semesters	NumSem
	ChangeOfMajor	Whether the student changed major	ChangeOfMajor
Semester-level	GPA	Grade point average in each semester	GPA_Sem1–GPA_Sem8
	RegisteredCredits	Number of registered credits in each semester	RegisteredCredits_Sem1–RegisteredCredits_Sem8
	AcademicWarning	Whether the student received an academic warning in each semester	AcademicWarning_Sem1–AcademicWarning_Sem8
	MajorCoursesCompleted	Number of completed major courses in each semester	MajorCoursesCompleted_Sem1–MajorCoursesCompleted_Sem8
	GeneralEducationCoursesCompleted	Number of completed general education courses in each semester	GeneralEducationCoursesCompleted_Sem1–GeneralEducationCoursesCompleted_Sem8
	ConsecutiveLeaveOfAbsenceFlag	Whether the student had a two-year consecutive leave-of-absence pattern up to the semester	ConsecutiveLeaveOfAbsenceFlag_Sem1–ConsecutiveLeaveOfAbsenceFlag_Sem8
	CounselingCount	Number of counseling sessions in each semester	CounselingCount_Sem1–CounselingCount_Sem8
	BookLoanCount	Number of library book loans in each semester	BookLoanCount_Sem1–BookLoanCount_Sem8
	AverageOverduePeriod	Average overdue period for library book loans in each semester	AverageOverduePeriod_Sem1–AverageOverduePeriod_Sem8
	ScholarshipTotal	Binary indicator of whether the student received any scholarship in each semester, derived from the original total scholarship amount	ScholarshipTotal_Sem1–ScholarshipTotal_Sem8
	ExcellentAdmissionScholarship	Whether the student received an excellent-admission scholarship in each semester	ExcellentAdmissionScholarship_Sem1–ExcellentAdmissionScholarship_Sem8
Target	Dropout	Final student status indicating dropout or non-dropout	Dropout

3.6. Evaluation Metrics and Statistical Analysis

Model performance was evaluated using true positives (TP), false positives (FP), false negatives (FN), precision, recall, F1 score, and candidate-list size. In this study, TP represents the number of actual dropout students included in the predicted candidate list, whereas FP represents the number of non-dropout students included in the candidate list. Candidate-list size was calculated as $TP + FP$.

Dropout candidate coverage was evaluated using recall, because recall measures the proportion of actual dropout students included in the candidate list. Candidate-list burden was evaluated using candidate-list size and FP . Precision was used to assess how efficiently the candidate list included actual dropout students under a given coverage condition.

Candidate-list overlap was assessed using Jaccard similarity. For the repeated-seed robustness analysis, the full experimental procedure was repeated using 30 different random seeds. Wilcoxon signed-rank tests were used to compare the proposed method with the threshold-tuned highest-F1 baseline across the repeated runs.

4. Results

4.1. Baseline Model Performance

Before evaluating the proposed method, we first selected a reference model for performance comparison. To simplify the discussion, the model with the highest F1 score is

adopted as the default baseline. To find the model, the 28 candidate models were evaluated on the validation folds using the default decision threshold of 0.5. In this experiment, the LightGBM + ADASYN model showed the best performance.

On the test dataset, the LightGBM + ADASYN baseline model achieved $TP = 438$, $FP = 42$, $FN = 67$, precision = 0.913, recall = 0.867, F1 score = 0.889, and a candidate-list size of 480.

Although this baseline model achieved strong precision and F1 performance, it covered only 438 of the 505 actual dropout students in the test dataset. This result indicates that the best-F1 model can provide a compact and accurate candidate list, but it may still exclude some actual dropout students.

4.2. Validation of the Proposed Method

The proposed method was applied to the 28 candidate models using only the training/validation dataset. First, we identified the model with the highest recall to determine the target candidate coverage. In this experiment, the SVM + ADASYN model achieved the highest average validation recall of 0.935. Therefore, the target coverage ρ^* was determined as 0.935.

After the model-specific thresholds were obtained and model performance was recalculated, we compared the validation precision of the models to identify the final solution. The LGBM + SMOTE model achieved the highest mean validation precision of 0.839 while maintaining a validation recall comparable to the target coverage. Therefore, the LGBM + SMOTE model was selected as the final solution model because it provided the most favorable validation-level balance between dropout candidate coverage and candidate-list size.

After the solution model was determined using the training/validation dataset, the model was applied to the test dataset to obtain the final candidate list. The model achieved $TP = 474$, $FP = 95$, $FN = 31$, precision = 0.833, recall = 0.939, F1 score = 0.883, and a candidate-list size of 569.

Compared with the baseline model with the highest F1 score, the proposed method increased dropout candidate coverage from 0.867 to 0.939, identifying 36 additional actual dropout students. This improvement was accompanied by an increase in false positives from 42 to 95 and an increase in candidate-list size from 480 to 569. This result indicates that the proposed method intentionally shifts model selection from maximizing the default F1 score alone toward increasing dropout candidate coverage under a manageable candidate-list size.

To examine whether the improvement could be obtained simply by threshold tuning of a strong single model, we also compared the proposed method with a threshold-tuned single-model baseline. For this purpose, the highest-F1 model was retained for threshold tuning, and the same threshold selection procedure discussed in Section 3.3 was applied to the model. The threshold-tuned model achieved $TP = 469$, $FP = 101$, $FN = 36$, precision = 0.823, recall = 0.929, F1 score = 0.873, and a candidate-list size of 570 on the test dataset.

Table 3 compares the performance of the proposed method with the two baseline models in the main experimental split.

Compared with the threshold-tuned highest-F1 baseline, the proposed model identified 5 additional dropout students, reduced false positives by 6, and reduced the candidate-list size by 1. It also improved precision from 0.823 to 0.833 and F1 score from 0.873 to 0.883. Although this improvement was modest, the result suggests that the proposed method can provide a favorable coverage-burden balance in the main experimental split. To further examine whether this comparison was dependent on a particular random split, we conducted an additional repeated-seed robustness analysis.

Table 3. Performance comparison of the proposed method with the two baseline models on the test dataset.

Method	Model	Threshold	TP	FP	FN	Precision	Recall	F1 Score	TP + FP
Highest-F1 baseline	LGBM + ADASYN	0.5	438	42	67	0.913	0.867	0.889	480
Threshold-tuned highest-F1 baseline	LGBM + ADASYN	Validation-derived	469	101	36	0.823	0.929	0.873	570
Proposed method	LGBM + SMOTE	Validation-derived	474	95	31	0.833	0.939	0.883	569

4.3. Hyperparameter Tuning Analysis

To reduce the reliance of experimental results on arbitrary default hyperparameters, a limited validation-based grid search was conducted within the training/validation stage. The test dataset was not used during this process. The hyperparameter search space and selected settings for the experiments are provided in Appendix A.2.

The final selected solution model belonged to the LGBM family, and the LGBM models showed consistently strong validation performance across sampling settings. Among the models evaluated at the default threshold of 0.5, the LGBM + ADASYN model achieved the highest F1 score of 0.902, followed closely by the LGBM + SMOTE model with an F1 score of 0.901. The difference between these models was small, indicating that the strong validation performance of the LGBM models was not limited to a single sampling setting.

It should be noted that the proposed method does not select the final solution model solely by maximizing the validation F1 score at the default threshold. Instead, after fold-specific validation-derived thresholds were applied to approximate the target coverage, the final model was selected based on the mean validation precision. Under this procedure, the LGBM + SMOTE model was selected as the final solution model because it achieved the highest mean validation precision under the target coverage. This result shows that the proposed method incorporates the coverage-burden trade-off after threshold derivation, rather than simply selecting the model with the highest validation F1 score.

4.4. SHAP-Based Interpretation of the Solution Model

To improve the interpretability of the selected solution model, a SHAP-based analysis for the LGBM + SMOTE model was conducted. This analysis was conducted only as a post hoc interpretation of the already fixed final model. It was not used for hyperparameter tuning, threshold selection, target coverage estimation, or final model selection.

The SHAP analysis was conducted using 2000 samples from the test dataset. Since the final model and its threshold had already been fixed before this analysis, the use of these samples was limited to model interpretation. The selected model used 92 input features, consisting of 4 student-level features and 88 semester-level features. The feature names used in the SHAP analysis correspond to the input variable names summarized in Table 2.

At the group level, semester-level features accounted for 58.18% of the total mean absolute SHAP importance, whereas student-level features accounted for 41.82%. This result indicates that the selected solution model used both student-level information and semester-wise academic variables, with semester-level information contributing slightly more to the model output.

Figure 2 presents the top 20 features ranked by mean absolute SHAP value.

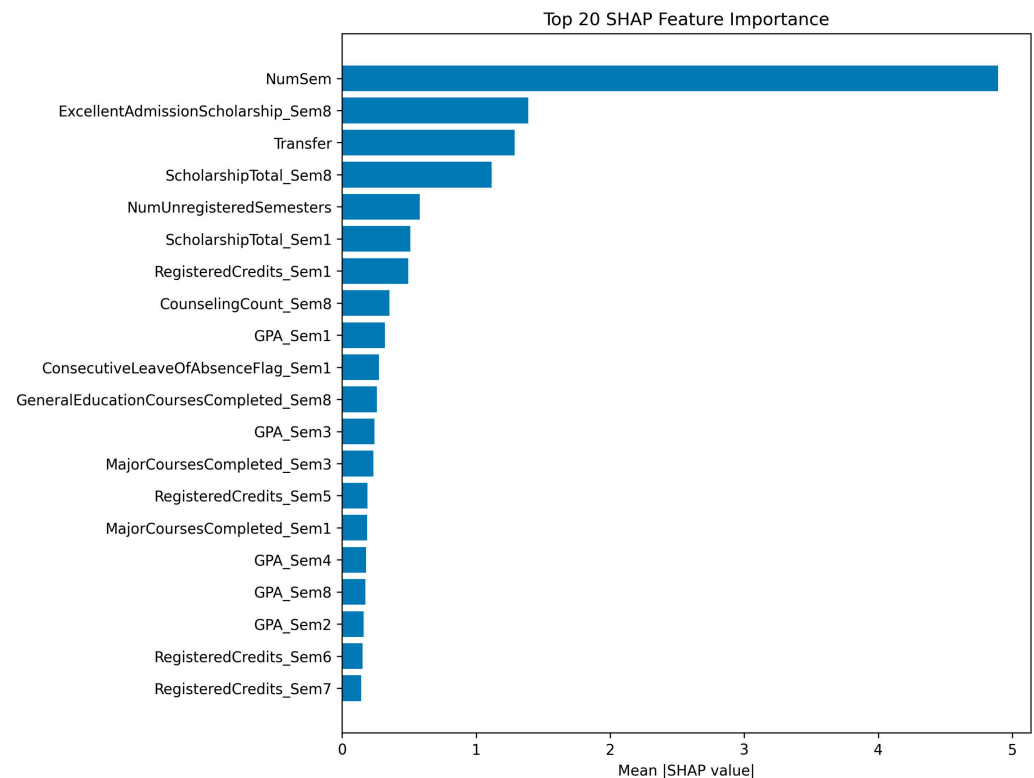


Figure 2. Top 20 SHAP feature importance values of the selected LGBM + SMOTE model.

The most influential feature was NumSem, followed by ExcellentAdmissionScholarship_Sem8, Transfer, ScholarshipTotal_Sem8, and NumUnregisteredSemesters. The remaining top-ranked features included scholarship-related variables, registered-credit variables, counseling count, GPA-related variables, course-completion variables, and a consecutive leave-of-absence flag. These results suggest that the final selected model relied on both student-level information and semester-level academic records when estimating dropout risk. Because ScholarshipTotal was binarized during preprocessing, the corresponding SHAP importance should be interpreted as the contribution of scholarship receipt rather than the raw scholarship amount.

It should be noted that SHAP values explain the contribution of features to model predictions and do not imply causal relationships. Therefore, the results should be interpreted as model-level evidence of which variables were influential in estimating dropout probability, rather than as evidence that changing a specific feature would directly reduce dropout.

4.5. Candidate-List Stability Analysis

To examine the stability of the selected candidate lists, we conducted a Jaccard similarity analysis. The Jaccard similarity measures the overlap between two candidate lists and is defined as the size of their intersection divided by the size of their union. In this study, the final candidate list generated by the proposed LGBM + SMOTE model was compared with the candidate lists generated by the two baseline models shown in Table 3.

The proposed candidate list showed a Jaccard similarity of 0.844 with the highest-F1 baseline and 0.950 with the threshold-tuned highest-F1 baseline. These values indicate that the proposed method did not generate an unstable or substantially unrelated candidate list. Instead, the proposed list largely overlapped with those of strong baseline models while improving the balance between dropout candidate coverage and counseling burden.

The overlap with the threshold-tuned highest-F1 baseline was particularly high. Although the proposed method selected LGBM + SMOTE and the threshold-tuned highest-F1

baseline used LGBM + ADASYN, the two candidate lists had 555 students in common, with a Jaccard similarity of 0.950. This result indicates that, in the main experimental split, the proposed method produced a candidate list that was highly consistent with a strong threshold-tuned baseline, while achieving higher dropout candidate coverage, fewer false positives, and a slightly smaller candidate list, as shown in Table 3.

Across all compared candidate lists, the mean Jaccard similarity with the proposed candidate list was 0.769, with a standard deviation of 0.150. The minimum and maximum values were 0.144 and 1.000, respectively. The minimum value was associated with a candidate list that was much broader than the proposed list. Overall, these results suggest that the proposed method provides a stable candidate-list construction strategy relative to the compared candidate lists.

4.6. Repeated-Seed Robustness Analysis

To examine whether the comparison with the threshold-tuned highest-F1 baseline was dependent on a particular random split, we repeated the full experimental procedure using 30 different random seeds. For each seed, the training/test split, validation-based target coverage estimation, model-specific threshold derivation, final model selection, and test evaluation were performed independently. Therefore, this analysis evaluates the robustness of the entire validation-derived selection procedure rather than the stability of a fixed threshold obtained from a single split.

As shown in Table 4, the proposed method achieved a mean *TP* of 471.13 and a mean recall of 0.933, while the threshold-tuned highest-F1 baseline achieved a mean *TP* of 471.67 and a mean recall of 0.934.

Table 4. Repeated-seed robustness analysis over 30 random seeds. Difference is calculated as proposed method minus threshold-tuned highest-F1 baseline.

Metric	Proposed Method	Threshold-Tuned Baseline	Difference	<i>p</i> -Value
TP	471.13 ± 6.79	471.67 ± 6.54	−0.53	0.366
FP	89.17 ± 11.15	94.07 ± 10.28	−4.90	0.017
Candidate-list size	560.30 ± 15.64	565.73 ± 13.65	−5.43	0.026
Recall	0.933 ± 0.013	0.934 ± 0.013	−0.001	0.311
Precision	0.841 ± 0.016	0.834 ± 0.015	+0.007	0.015
F1 score	0.885 ± 0.008	0.881 ± 0.009	+0.004	0.016

5. Discussion

The experimental results show that selecting a dropout prediction model solely by the highest F1 score may not provide the most appropriate candidate list for institutional intervention. The default highest-F1 baseline, LGBM + ADASYN, achieved high precision and a relatively compact candidate list under the default threshold of 0.5. However, it covered only 438 of the 505 dropout students in the test dataset, corresponding to a dropout candidate coverage of 0.867. This indicates that the model with the highest F1 score can provide an accurate candidate list, but it may still exclude additional actual dropout students.

The proposed method increased dropout candidate coverage compared with the default highest-F1 baseline. Specifically, the proposed LGBM + SMOTE model covered 474 actual dropout students, increasing recall from 0.867 to 0.939. This improvement required a larger candidate list and more false positives, which reflects the inherent trade-off between dropout candidate coverage and counseling burden. Therefore, the proposed method should not be interpreted as improving all classification metrics. Rather, it in-

tentionally shifts model selection toward higher candidate coverage while keeping the candidate-list size manageable.

The comparison with the threshold-tuned highest-F1 baseline further clarifies the contribution and limitation of the proposed method. In the main experimental split, the threshold-tuned LGBM + ADASYN model achieved a dropout candidate coverage of 0.929 with a candidate-list size of 570, whereas the proposed LGBM + SMOTE model achieved a higher coverage of 0.939 with a slightly smaller candidate list of 569 students. However, because the difference in the main split was modest, we additionally performed a repeated-seed robustness analysis.

Across 30 random seeds, the proposed method did not significantly increase *TP* or recall compared with the threshold-tuned highest-F1 baseline. Instead, it maintained comparable dropout candidate coverage while significantly reducing false positives and candidate-list size. It also improved precision and F1 score. Therefore, the proposed method should be interpreted as improving candidate-list efficiency under comparable coverage, rather than as consistently maximizing dropout candidate coverage beyond the threshold-tuned baseline.

The candidate-list stability analysis also supports this interpretation. The proposed candidate list had a Jaccard similarity of 0.950 with the threshold-tuned highest-F1 baseline, indicating that the two lists largely overlapped. However, the proposed method achieved a more efficient candidate-list construction within this highly similar candidate pool, particularly by reducing false positives and candidate-list size under comparable coverage. This suggests that the proposed method does not generate an unstable or substantially different candidate list but refines candidate-list construction by selecting a more suitable model under the validation-derived coverage target.

It is important to emphasize that the proposed method does not estimate the actual effect of counseling or academic intervention. Because post-intervention outcome data were not available, the analysis was limited to whether actual dropout students were included in the candidate list. Thus, the reported improvement should be interpreted as improved dropout candidate coverage and candidate-list construction, not as evidence of actual dropout reduction.

This study has several limitations. First, the experiments were conducted using data from a single university. Although the dataset contains a large number of student records, additional validation using datasets from other institutions is required to assess generalizability. Second, candidate-list size was evaluated as a proxy for counseling burden, but actual institutional burden may depend on staff availability, counseling intensity, intervention type, and follow-up procedures. Third, the proposed method assumes that including an actual dropout student in the candidate list is useful for intervention planning, but the actual effectiveness of the intervention depends on post-prediction institutional actions. Fourth, although the test dataset was not used for target coverage estimation, threshold derivation, or final model selection, the same validation folds were used for both threshold derivation and model selection. This may introduce limited validation-level optimism, and the results should therefore be interpreted as the final test evaluation of a validation-derived procedure rather than as evidence of a universally optimal operating point.

Overall, the results indicate that the proposed validation-derived method can support more transparent candidate-list construction by separating threshold estimation and model selection from final test evaluation.

6. Conclusions

This study investigated model selection for university student dropout prediction from the perspective of candidate-list construction. While previous studies have commonly

selected models using accuracy-oriented metrics such as the F1 score, the practical use of dropout prediction models requires consideration of both dropout candidate coverage and counseling burden. To address this issue, we proposed a validation-derived model selection method in which the target candidate coverage, model-specific classification thresholds, and final solution model are determined using only training/validation data. The test dataset was reserved strictly for final performance evaluation.

Using academic records from 20,050 students, we evaluated 28 candidate models constructed from seven machine learning algorithms and four sampling settings. The proposed method selected LGBM + SMOTE as the final solution model. The selected model achieved $TP = 474$, $FP = 95$, $FN = 31$, precision = 0.833, recall = 0.939, F1 score = 0.883, and a candidate-list size of 569 on the test dataset.

Additional repeated-seed analysis over 30 random seeds showed that the proposed method maintained dropout candidate coverage comparable to the threshold-tuned highest-F1 baseline while significantly reducing false positives and candidate-list size. These results suggest that the proposed method can provide a practical model selection strategy for improving candidate-list efficiency while maintaining the target level of dropout candidate coverage. Importantly, the method does not use the test dataset for threshold estimation or model selection, thereby reducing the risk of test-set-based threshold optimization.

This study has several limitations. First, the experiments were conducted using data from a single university, and further validation using datasets from other institutions is required. Second, candidate-list size was used as a proxy for counseling burden, although actual intervention burden may vary depending on institutional resources and counseling procedures. Third, this study did not include post-intervention counseling outcome data, which limits the evaluation to candidate-list construction rather than actual dropout reduction.

Future research will extend the proposed method in three directions. First, we will validate the method using datasets from multiple institutions to examine its generalizability. Second, we will incorporate post-intervention outcome data when available to evaluate whether improved candidate-list construction leads to actual dropout reduction. Third, we will investigate adaptive intervention strategies that consider not only dropout risk but also institutional counseling capacity and student-specific intervention needs.

Author Contributions: Conceptualization, K.W.K., C.K. and H.G.K.; Methodology, K.W.K., C.K. and H.G.K.; Software, K.W.K.; Validation, K.W.K. and H.G.K.; Formal analysis, H.G.K.; Investigation, C.K.; Writing—original draft, H.G.K.; Writing—review & editing, H.G.K.; Supervision, H.G.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding. The APC was funded by the authors.

Data Availability Statement: The data presented in this study are not publicly available because they contain institutional student academic records and are subject to privacy and institutional restrictions. Aggregated results supporting the findings are reported in the article.

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

Appendix A

Appendix A.1. Symbols Used in This Paper

Table A1 summarizes the main symbols used in the mathematical formulation of the proposed method. These symbols are defined to clarify the notation used in Sections 3.1–3.3.

Table A1. Symbols used in this paper.

Symbol	Description
S	Input data
S_T	Training/validation dataset extracted from S
S_E	Test/evaluation dataset extracted from S
N	Number of samples in S
N_P	Number of positive samples in S
N_N	Number of negative samples in S
s_i	i -th sample in a given dataset
t_i	Ground-truth label of s_i , where $t_i = 1$ for dropouts and $t_i = 0$ for non-dropouts
y_i	Predicted label for s_i
U	Set of candidate models
K	Number of models in U
M_j	j -th candidate model in U
L_j	Candidate list generated by M_j
$p_{i,j}$	Dropout probability for s_i predicted by M_j
τ_j	Classification threshold of M_j
TP_j, FP_j, FN_j	TP, FP, and FN calculated by M_j on a given evaluation dataset
M_ρ	Model with the highest recall in U
ρ^*	Average recall of k validations of M_ρ
M_ω	Solution model

Appendix A.2. Hyperparameter Search Space and Selected Settings

Table A2 summarizes the hyperparameter search space and the selected settings used for the candidate models. Hyperparameter tuning was performed only within the training/validation stage, and the test dataset was not used in this process.

Table A2. Hyperparameter search space and selected settings used in the experiments.

Algorithm	Hyperparameter	Candidate Values	Selected Setting
DT	max_depth	{5, 10, 15}	Original: 10; SMOTE: 10; BL-SMOTE: 10; ADASYN: 15
DT	min_samples_leaf	{1, 5}	Original: 1; SMOTE: 5; BL-SMOTE: 1; ADASYN: 5
RF	n_estimators	{100, 200}	Original: 100; SMOTE: 100; BL-SMOTE: 200; ADASYN: 200
RF	max_depth	{10, None}	None
RF	min_samples_leaf	{1}	1
DNN	hidden_units	{[64, 32], [128, 32], [128, 64]}	Original: [64, 32]; SMOTE: [64, 32]; BL-SMOTE: [128, 64]; ADASYN: [128, 32]
DNN	dropout	{0.0, 0.2}	Original: 0.2; SMOTE: 0.2; BL-SMOTE: 0.0; ADASYN: 0.0
DNN	learning_rate	{0.001}	0.001
XGB	n_estimators	{100, 200}	Original: 100; SMOTE: 200; BL-SMOTE: 200; ADASYN: 200
XGB	max_depth	{3, 5}	5
XGB	learning_rate	{0.1}	0.1
CB	iterations	{100, 200}	200
CB	depth	{4, 6}	6
CB	learning_rate	{0.1}	0.1
LGBM	n_estimators	{100, 200}	200
LGBM	num_leaves	{31, 63}	Original: 63; SMOTE: 63; BL-SMOTE: 31; ADASYN: 63
LGBM	learning_rate	{0.1}	0.1
SVM	model	Fixed	LinearSVC
SVM	C	Fixed	1.0
SVM	max_iter	Fixed	10,000
SVM	dual	Fixed	False

References

- Kim, D.; Kim, S. Sustainable Education: Analyzing the Determinants of University Student Dropout by Nonlinear Panel Data Models. *Sustainability* **2018**, *10*, 954. [\[CrossRef\]](#)
- Mduma, N.; Kalegele, K.; Machuve, D. A Survey of Machine Learning Approaches and Techniques for Student Dropout Prediction. *Data Sci. J.* **2019**, *18*, 14. [\[CrossRef\]](#)
- Fierro Saltos, W.R.; Fierro Saltos, F.E.; Elizabeth Alexandra, V.S.; Rivera Guzmán, E.F. Leveraging Artificial Intelligence for Sustainable Tutoring and Dropout Prevention in Higher Education: A Scoping Review on Digital Transformation. *Information* **2025**, *16*, 819. [\[CrossRef\]](#)
- Pelima, L.R.; Sukmana, Y.; Rosmansyah, Y. Predicting University Student Graduation Using Academic Performance and Machine Learning: A Systematic Literature Review. *IEEE Access* **2024**, *12*, 23451–23465. [\[CrossRef\]](#)
- Alnasyan, B.; Basher, M.; Alassafi, M. The Power of Deep Learning Techniques for Predicting Student Performance in Virtual Learning Environments: A Systematic Literature Review. *Comput. Educ. Artif. Intell.* **2024**, *6*, 100231. [\[CrossRef\]](#)
- Colpo, M.P.; Thompsen Primo, T.; Aguiar, M.S.D.; Cechinel, C. Educational Data Mining for Dropout Prediction: Trends, Opportunities, and Challenges. *Rev. Bras. Inform. Educ.* **2024**, *32*, 220–256. [\[CrossRef\]](#)
- Araque, F.; Roldán, C.; Salguero, A. Factors Influencing University Drop out Rates. *Comput. Educ.* **2009**, *53*, 563–574. [\[CrossRef\]](#)
- Chen, M.; Wu, L. A Dropout Prediction Method Based on Time Series Model in MOOCs. *J. Phys. Conf. Ser.* **2021**, *1774*, 012065. [\[CrossRef\]](#)
- Zheng, Y.; Gao, Z.; Wang, Y.; Fu, Q. MOOC Dropout Prediction Using FWTS-CNN Model Based on Fused Feature Weighting and Time Series. *IEEE Access* **2020**, *8*, 225324–225335. [\[CrossRef\]](#)
- Kemper, L.; Vorhoff, G.; Wigger, B.U. Predicting Student Dropout: A Machine Learning Approach. *Eur. J. High. Educ.* **2020**, *10*, 28–47. [\[CrossRef\]](#)
- Cho, C.H.; Yu, Y.W.; Kim, H.G. A Study on Dropout Prediction for University Students Using Machine Learning. *Appl. Sci.* **2023**, *13*, 12004. [\[CrossRef\]](#)
- Kim, S.; Choi, E.; Jun, Y.-K.; Lee, S. Student Dropout Prediction for University with High Precision and Recall. *Appl. Sci.* **2023**, *13*, 6275. [\[CrossRef\]](#)
- Nieto, Y.; Gacia-Diaz, V.; Montenegro, C.; Gonzalez, C.C.; Gonzalez Crespo, R. Usage of Machine Learning for Strategic Decision Making at Higher Educational Institutions. *IEEE Access* **2019**, *7*, 75007–75017. [\[CrossRef\]](#)
- Ujkani, B.; Minkovska, D.; Stoyanova, L. Application of Logistic Regression Technique for Predicting Student Dropout. In *Proceedings of the 2022 XXXI International Scientific Conference Electronics (ET)*; IEEE: Sozopol, Bulgaria, 2022; pp. 1–4.
- Agrusti, F.; Mezzini, M.; Bonavolontà, G. Deep Learning Approach for Predicting University Dropout: A Case Study at Roma Tre University. *J. e-Learn. Knowl. Soc.* **2020**, *16*, 44–54. [\[CrossRef\]](#)
- Fernandez-Garcia, A.J.; Preciado, J.C.; Melchor, F.; Rodriguez-Echeverria, R.; Conejero, J.M.; Sanchez-Figueroa, F. A Real-Life Machine Learning Experience for Predicting University Dropout at Different Stages Using Academic Data. *IEEE Access* **2021**, *9*, 133076–133090. [\[CrossRef\]](#)
- Song, Z.; Sung, S.-H.; Park, D.-M.; Park, B.-K. All-Year Dropout Prediction Modeling and Analysis for University Students. *Appl. Sci.* **2023**, *13*, 1143. [\[CrossRef\]](#)
- Na, J.; Kim, K.W.; Kim, H.G. A Study on Exploiting Temporal Patterns in Semester Records for Efficient Student Dropout Prediction. *Electronics* **2025**, *14*, 4356. [\[CrossRef\]](#)
- Zanellati, A.; Zingaro, S.P.; Gabbrielli, M. Balancing Performance and Explainability in Academic Dropout Prediction. *IEEE Trans. Learn. Technol.* **2024**, *17*, 2086–2099. [\[CrossRef\]](#)
- Rabelo, A.M.; Zárate, L.E. A Model for Predicting Dropout of Higher Education Students. *Data Sci. Manag.* **2025**, *8*, 72–85. [\[CrossRef\]](#)
- Lee, S.; Chung, J.Y. The Machine Learning-Based Dropout Early Warning System for Improving the Performance of Dropout Prediction. *Appl. Sci.* **2019**, *9*, 3093. [\[CrossRef\]](#)
- 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*, 275. [\[CrossRef\]](#)
- Elkan, C. The Foundations of Cost-Sensitive Learning. In *Proceedings of the 17th International Joint Conference on Artificial Intelligence*; Morgan Kaufmann Publishers Inc.: San Francisco, CA, USA, 2001; Volume 2, pp. 973–978.
- Provost, F.; Fawcett, T. Robust Classification for Imprecise Environments. *Mach. Learn.* **2001**, *42*, 203–231. [\[CrossRef\]](#)
- Lakkaraju, H.; Aguiar, E.; Shan, C.; Miller, D.; Bhanpuri, N.; Ghani, R.; Addison, K.L. A Machine Learning Framework to Identify Students at Risk of Adverse Academic Outcomes. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*; ACM: Sydney, NSW, Australia, 2015; pp. 1909–1918.
- Aguiar, E.; Lakkaraju, H.; Bhanpuri, N.; Miller, D.; Yuhas, B.; Addison, K.L. Who, When, and Why: A Machine Learning Approach to Prioritizing Students at Risk of Not Graduating High School on Time. In *Proceedings of the Fifth International Conference on Learning Analytics and Knowledge*; ACM: Poughkeepsie, NY, USA, 2015; pp. 93–102.

27. Arnold, K.E.; Pistilli, M.D. Course Signals at Purdue: Using Learning Analytics to Increase Student Success. In *Proceedings of the 2nd International Conference on Learning Analytics and Knowledge*; ACM: Vancouver, BC, Canada, 2012; pp. 267–270.
28. Adnan, M.; Habib, A.; Ashraf, J.; Mussadiq, S.; Raza, A.A.; Abid, M.; Bashir, M.; Khan, S.U. Predicting At-Risk Students at Different Percentages of Course Length for Early Intervention Using Machine Learning Models. *IEEE Access* **2021**, *9*, 7519–7539. [[CrossRef](#)]
29. Chawla, N.V.; Bowyer, K.W.; Hall, L.O.; Kegelmeyer, W.P. SMOTE: Synthetic Minority Over-Sampling Technique. *J. Artif. Intell. Res.* **2002**, *16*, 321–357. [[CrossRef](#)]
30. Han, H.; Wang, W.-Y.; Mao, B.-H. Borderline-SMOTE: A New Over-Sampling Method in Imbalanced Data Sets Learning. In *Advances in Intelligent Computing*; Huang, D.-S., Zhang, X.-P., Huang, G.-B., Eds.; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2005; Volume 3644, pp. 878–887.
31. 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 (IEEE World Congress on Computational Intelligence)*; IEEE: New York, NY, USA, 2008; pp. 1322–1328.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.