RECALL IN PERIL

IMPROVING CORPORATE BANKRUPTCY PREDICTION MODELS THROUGH HYBRID RESAMPLING AND FEATURE SELECTION IN IMBALANCED DATA

SYRYM TOLESH

THESIS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE IN DATA SCIENCE & SOCIETY

AT THE SCHOOL OF HUMANITIES AND DIGITAL SCIENCES

OF TILBURG UNIVERSITY

STUDENT NUMBER

2068273

COMMITTEE

dr. B. Čule
dr. Elisabeth Huis in 't Veld

LOCATION

Tilburg University
School of Humanities and Digital Sciences
Department of Cognitive Science & Artificial Intelligence
Tilburg, The Netherlands

DATE

January 15th, 2024

WORD COUNT: 8,439

ACKNOWLEDGMENTS

First and foremost, I want to thank my family for all the support given to me throughout my life.

Secondly, my heartfelt gratitude goes to Monica, who has been indispensable to me on this journey, propelling me to a place of immense value and achievement.

Finally, I express my gratitude to Dr. Čule for providing me with the advice and steering me in the right direction.

TABLE OF CONTENTS

Table of Figures

RECALL IN PERIL

IMPROVING CORPORATE BANKRUPTCY PREDICTION MODELS THROUGH HYBRID RESAMPLING AND FEATURE SELECTION IN IMBALANCED DATA

SYRYM TOLESH

Abstract

Predicting bankruptcy is significant to stakeholders such as business managers, investors, creditors, and regulators because it facilitates informed decision-making and risk management. Nevertheless, the primary challenge in this domain is the issue of class imbalance. Effectively addressing this challenge in corporate bankruptcy prediction is essential to fully harness the potential of predictive models in this critical domain.

This study investigates how the implementation of hybrid resampling techniques, along with feature selection methods, impacts the performance of supervised models, specifically Decision Trees and Random Forest, when dealing with imbalanced datasets. The study demonstrates that integrating hybrid resampling techniques, such as SMOTE-Tomek, SMOTE-ENN, and ADASYN, significantly enhances the performance of both models across selected metrics when compared to the baseline performance of these classifiers. Specifically, SMOTE-ENN emerges as the top-performing hybrid resampling method, achieving the lowest Type II errors for both models.

Additionally, applying feature selection methods, like filter-based Mutual Information and wrapper-based Recursive Feature Elimination, on resampled data using hybrid resampling methods further improves predictive performance. The Decision Tree model, utilizing ADASYN and Mutual Information, outperforms other models, while in Random Forest models, the combination of SMOTE-ENN with Mutual Information emerges as the top performer. The findings highlight Mutual Information as the most effective feature selection method, consistently yielding superior results for both models, offering insights into the synergy between resampling and feature selection in bankruptcy prediction.

DATA SOURCE, ETHICS, CODE, AND TECHNOLOGY (DSECT) STATEMENT

The dataset is covered under the terms of a Creative Commons Attribution 4.0 International Public license (CC BY 4.0), which permits for the public availability, sharing and modification of the data for any intent, under the condition that proper credit is attributed to the original creator of the dataset as required by licensor. Data for this study was gathered from the Taiwan Economic Journal in the years from 1999 to 2009, all information was collected as required by the rules and regulation stipulated by the Taiwan Stock Exchange governing body. Work on this thesis did not involve collecting data from human participants or animals. The obtained data is anonymized. The author of this study acknowledges that they do not have any legal claim to the data. The code used in this thesis is not publicly available.

Academic Attribution Link: www.archive.ics.uci.edu/dataset/572/taiwanese+bankruptcy+prediction

1. INTRODUCTION

Bankruptcy prediction is of crucial importance to a variety of stakeholders who consider the far-reaching potential consequences that this might have given the ever-growing complexity and interconnectedness of the economy. Business managers, investors, creditors, regulatory and government agencies, as well as employees and labor unions, all have a vested interest in identifying the prospective risk of insolvency. Consequently, developing efficient and robust prediction models can assist them in assessing financial risks, making informed decisions, and taking proactive and preemptive actions to safeguard their interests, investments, and livelihoods.

In the context of machine intelligence, bankruptcy prediction models employ a binary classification system that categorizes data instances into two different classes: bankrupt and non-bankrupt, based on patterns and attributes in the training data. Most real-world datasets used by supervised learning models exhibit an uneven distribution of data across different classes, known as the class imbalance problem, which is inherent in real-world datasets (Kubat, 2000). This issue involves an unequal distribution of data, where one class has too few data points, referred to as the minority class, while the other class contains a substantially larger number of data points, known as the majority class. Without addressing the class imbalance problem, many supervised learning models experience a reduction in their predictive performance. Class imbalance persists as a challenge in numerous domains that deal with real-world data, including disease diagnosis (Luo & Xiao, 2017), fraud detection (Zakaryazad & Duman, 2016), IT security management (Bennin et al., 2018), protein detection (Herndon & Caragea, 2016), and, of course, bankruptcy prediction.

Predictive machine learning models evaluate and train on financial statements of various companies to identify patterns and relations in the data to make predictive assessments of a company’s financial condition. Leveraging these models, internal and external stakeholders can strategize and implement various initiatives aimed at alleviating the main drivers of impending bankruptcy. Furthermore, improving the predictive ability of such models is significant to the stakeholders, considering the consequences of incorrectly classifying a financially robust as insolvent or vice versa.

However, class imbalance remains a persistent challenge in corporate bankruptcy prediction, as it does in most real-world datasets (Wang & Liu, 2021). In response, researchers explored specific strategies like feature selection and various resampling techniques. While some studies have concentrated on individual methods to assess their performance in combination with classification models, others have delved into feature selection methods to improve predictive accuracy.

The literature review provided below has highlighted various methods and approaches to address the challenge of class imbalance in real-world datasets. However, the distinct lack of studies on the combination of feature selection and hybrid resampling methods on model performance and interpretability in this domain warrants further exploration. The scientific contribution of this study, which investigates the combination of feature selection and hybrid resampling methods, lies in advancing our understanding of the interplay between their combined impact on model performance and interpretability in the context of class imbalance, being able to provide valuable insights into the optimal integration of these methods, potentially uncovering new strategies for enhancing predictive modeling in imbalanced datasets, proving economic significance to stakeholders in the financial domain. Moreover, improving predictive models can significantly reduce financial risks and enhance economic stability, directly benefiting stakeholders such as creditors, investors, and employees.

1.1 Research Questions

Based on the identified research gap and promise of scientific relevance, the main research goal of this study is to explore the potential synergies of combining hybrid resampling techniques and feature selection methods in improving bankruptcy prediction model performance using imbalanced data. Therefore, the aim of this study is to answer the following main research question:

To what extent does the implementation of hybrid resampling techniques with feature selection methods enhance the predictive performance of supervised models on corporate insolvency when dealing with imbalanced datasets?

The following sub-questions aim to provide a comprehensive understanding of the factors contributing to the improvement of model performance. Firstly, this study examines the performance of the Decision Tree (DT) and Random Forest (RF) models without hybrid resampling techniques and feature selection methods. These findings will serve as a benchmark, providing insights into the strengths and weaknesses of the applied techniques on predictive performance.

Sub-Q1 *To what extent can Decision Tree and Random Forest predict the outcomes of corporate bankruptcy using an imbalanced dataset?*

Secondly, the specific impact of hybrid resampling techniques that simultaneously combine under-sampling and over-sampling is evaluated to determine their effectiveness in enhancing the predictive performance of insolvency models.

Sub-Q2 *What is the impact of implementing hybrid resampling techniques on the predictive performance of Decision Tree and Random Forest models?*

Lastly, the role of feature selection methods in conjunction with hybrid resampling methods must be assessed. Specifically, whether the inclusion of feature selection methods improves the model’s ability to make accurate predictions.

Sub-Q3 *How does the integration of feature selection methods after implementing hybrid resampling techniques impact the predictive performance of bankruptcy prediction models?*

The answers to these sub-questions will collectively provide insights into the synergistic effects of hybrid resampling and feature selection on bankruptcy prediction, directly addressing the main research question.

In this study, Random Forest and Decision Tree models are utilized to evaluate bankruptcy prediction performance through a hybrid approach combining the outlined methods. For each step of the experiment, the F2-measure, Area Under the Curve (AUC), and analyses of Type I and Type II errors will be employed to assess predictive performance. The AUC is chosen for its robustness in scenarios with class imbalance, while the F2-measure, which combines recall and precision, is selected to effectively minimize false positives and false negatives. Analyzing Type I and Type II errors is crucial for understanding instances where the model incorrectly predicts a non-event as an event (false positive) or fails to identify an actual event (false negative), thus providing vital insights into the reliability and accuracy of the model's performance (Wang et al., 2014).

2. Related Work

2.1 Bankruptcy prediction models

In the 1960s, bankruptcy prediction models were based on statistical techniques to measure the likelihood of insolvency based on the financial ratios of the company (Beaver, 1966), which was later improved through the development of Altman's Z-score model (Altman, 1968) and Ohlson O-score (Ohlson, 1980). The generalization performance of these models to predict bankruptcy was constrained by their theoretical applicability, considering they worked under restrictive assumptions: they were not sufficiently robust in representing normality and independence among input variables (Wang et al., 2014), and consequently, further exploration was required. Continuous advancements in Artificial Intelligence and Machine Learning shifted the focus from statistical approaches and hypothesis testing to newer, more intelligent methodologies. Such methodologies were not constrained by challenges of linearity, interdependence, and sensitivity to outliers, while excelling in their ability to handle large amounts of data to find intricate patterns and relationships within a multitude of attributes (Kumar, 2007).

The problem of corporate bankruptcy prediction is viewed as a classification problem, with many studies utilizing ensemble and standalone machine learning models to classify companies as bankrupt or non-bankrupt based on financial and accounting data. There is no single best ML model for corporate bankruptcy prediction problem, as the context of each problem is different and often involves imbalanced datasets, necessitating the implementation of resampling techniques (Le et al., 2018), application of feature selection (FS) methods to capture representative data and applying hyperparameter tuning for the selected models to achieve best predictive performance (Paraschiv et al., 2023). Therefore, studies focus on these specific techniques that improve the performance of standalone and ensemble models due to their superior ability to capture complex data patterns from diverse datasets (Hung & Binh, 2021), capacity to manage the intricacies of feature importance and interaction and maintain robustness across diverse datasets. (Yotsawat et al., 2023)

Studies have successfully enhanced the predictive performance in both standalone and ensemble models to address the significant hindrance caused by imbalanced data when applying different resampling techniques based on oversampling the minority class (Sisodia & Verma, 2018), under-sampling the majority class (Wang & Liu, 2021), or a combination of the two (Le, 2021). Additionally, FS has been an active research area in corporate bankruptcy prediction as it reduces the dimensionality of feature space, removes redundant or noisy data, and improves predictive performance of the models in question (Zoričák et al., 2020; Aly et al., 2022; Faris et al., 2020). FS techniques fall into two categories: filter-based methods that assess the relevance of features based on statistical measures, and wrapper-based methods that incorporate a specific ML algorithm as a part of the selection process (Tsai et al., 2014).

Table 1 presents the current state of knowledge about the influence of FS methods and resampling methods on the performance bankruptcy prediction models. The table below provides an overview of related studies examining the performance of bankruptcy prediction models, specifically, what models were implemented, and whether they considered resampling methods and/or FS techniques. Similarly, Table 2 provides a summary of the latest research on hybrid resampling techniques in the context of corporate bankruptcy prediction.

2.2 Resampling methods

In corporate bankruptcy prediction datasets, the data often exhibits an uneven class distribution, where there is a small number of instances in one class (minority class) and a vast majority of instances in the other class (majority class). The predictive performance of supervised models is hampered when encountering a class imbalance problem described above deteriorates as the model becomes biased towards the majority class. Consequently, advanced methods were created to handle the class imbalance problem, they can be divided into three groups: oversampling methods, under-sampling methods, and hybrid methods combining the two.

Most of the research tackling the class imbalance problem in the context of corporate bankruptcy prediction primarily focuses on oversampling methods, specifically, the most popular oversampling method Synthetic Minority Over-sampling Technique (SMOTE), as it is evident that its ability to generate minority synthetic samples has a substantial positive effect on the performance of standalone and ensemble models (Chawla et al., 2002). Although there is no industry standard under-sampling technique used in research, under-sampling methods can obtain the same number of class samples without a decrease in training time and performance (Vellamcheti & Singh, 2020). A study using the Taiwanese bankruptcy dataset comparing the performance of 5 under-sampling methods found Gaussian Naïve Bayes combined with ENN has the best performance (Wang & Liu, 2021). Similarly, a study comparing the performance of standalone and ensemble learners using four oversampling and two under-sampling methods, found RF and DT to perform best with under-sampling methods, and Logistic Regression and DTBagging to perform best with oversampling, with DTBagging performing the best overall (Sisodia & Verma, 2018). A similar study utilizing the same dataset as Sisodia & Verma, plus an additional new dataset, found Random Forest (RF) with RUS method had the highest AUC on the datasets, however, RF was slightly outperformed by CS-XGB on all other parameters across both datasets (Yotsawat et al., 2023).

Table 1: Related Works Overview

Work	Sampling methods: Oversampling	Sampling methods: Under-sampling	Feature selection: Filter	Feature selection: Wrapped	Prediction models: Ensemble	Prediction models: Standalone
Yotsawat et al. (2023)	SMOTE	RUS	None		AB, XGB, RF, CS-XGB, B-DT, B-NN	
Vellamcheti & Singh (2020)	SMOTE	RUS	None		GMB, CB, RF, AB	DT, MLP, SVM, LDA, LR, C4.5,
Le et al. (2018)	SMOTE	ENN, IHT	None		GMB, CB, RF	DT, MLP
Sisodia & Verma (2018)	SMOTE, ROS B-SMOTE, SL-SMOTE,	RUS	None		RF, AB, DT- BG	C4, LR, SVM
Wang & Liu (2021)		TL, ENN, RENN, OSS, NCR	None		XGB, CB, RF	DT, LR, MLP, SVM, NB, k-NN, NN,LDA
Premalatha et al. (2023)	SMOTE		PC, MI, VT		AB, XGB, RF	DT
Aly et al. (2022)	SMOTE		CFS,	SFS, RFE	CB	
Faris et al. (2020)	SMOTE		CFS, IG,CAE, ReliefF, CorrAE		RF BSTN, BGNN, AB	DT, MLP, NB, k-NN, ANN
Paraschiv et al. (2023)	None		PC	FFS		LR, DNN
Tsai (2010)	None		T-test, CM	PCA, SR		MLP
Lin et al. (2018)	None		IG	GA		LR, NB, NN, C4.5, SVM, k-NN
Zoričák et al. (2020)	None		FS, ReliefF	RFE,		SVM, IF, LSAD, OCSVM
Hung & Binh (2021)	None		None		BSTN, BGNN	LR, SVM, DT, k-NN
Wang et al. (2014)	None		None		BSTN, BGNN,	LRA, NB, DT, ANN, SVM

AB – AdaBoost, BGN - Bagging, BST - Boosting, B-DT – Bagging Decision Tree, B-NN – Bagging Neural Network, XGB – XGBoost, CS-XGB – Cost Sensitive XGBoost, GMB – GMBoost, CB – Cboost, LR – Logistic Regression, SVM Support Vector Machine, DT – Decision Tree, LDA – Linear Discriminant Analysis, RF – Random Forest, MLP – Multi Layer Perceptron, C4.5 – Decision Tree, LRA - Logistic Regression Analysis, NB – Naïve Bayes, ANN – Artificial Neural Network, DNN – Deep Neural Network, k-NN – k nearest neighbors, IF – Isolation Forest, LSAD - Local Subspace Anomaly Detection, OCSVM - One-Class Support Vector Machine, RUS – Random Under-sampling, B-SMOTE – Borderline-SMOTE, SL-SMOTE - Safe-Level-SMOTE, ROS – Random Oversampling

Table 2: Overview of hybrid resampling techniques

Work	Hybrid resampling methods	Feature selection		Prediction models	
		Filter	Wrapped	Ensemble	Standalone
Le et al. (2018)	Borderline-SMOTE, ADASYN, SMOTE-ENN, SMOTE-Tomek	None		RF	ReLU, MLP, SVM, DT
Le (2021)	SMOTE-ENN, ADASYN, Borderline-SMOTE, SMOTE-Tomek	None		CB, XGB, RF	

Similarly, other studies have implemented models belonging to the family of gradient boosting algorithms, such as AdaBoost, XGBoost, and CatBoost. These models have shown the best results in their respective studies due to their respective ability to focus on correcting errors made by weak learner-based on weighted training instances, rate feature contribution to the models' predictions, and ability to handle categorical features automatically (Faris et al., 2020; Le, 2021; Le et al., 2018). Similarly to the research of Yotsawat et al., the study of Vellamcheti and Singh trained 13 ensemble and standalone baseline models on the same datasets with the same under-sampling and oversampling methods. The experimental results found their custom-tuned MLP model to be the best-performing model measured on AUC. Although AdaBoost and RF performed marginally worse on the selected performance metrics, they were trained only with default hyperparameters (Vellamcheti & Singh, 2020).

Highly relevant to this research, a study focused on specific hybrid resampling methods with ensemble learners, results found RF with SMOTE-ENN to be the best-performing classifier in the imbalanced learning framework that analyzed two oversampling techniques and three hybrid resampling techniques (Le, 2021). Similarly, a study comparing the performance of ensemble and standalone models using hybrid resampling found RF with SMOTE-ENN and DT with SMOTE-Tomek to be the best performing across their categories, additionally, hybrid techniques provided better results on AUC than the traditional resampling techniques (Le et al., 2018).

2.3 Feature selection (FS)

Table 1 presents related works related to FS in the context of corporate bankruptcy prediction. In most studies, researchers evaluate the performance of filter and wrapper-based FS methods in combination with oversampling methods, or with no resampling methods at all. Moreover, the following studies focused on improving the performance of either standalone or ensemble models with FS methods.

An increasing area of interest for researchers is the impact of FS methods on bankruptcy prediction performance. FS methods aim to find the best subset of features within the dataset using wrapper-based or filter-based feature selection techniques and assess how these selected features contribute to improving the overall performance of prediction models, while mitigating the risk of overfitting and reducing the running time of a learner (Lin et al., 2018). For example, a study evaluating the effect of feature selection methods on the performance of MLP found that both filter and wrapper-based methods enhance the performance of the learner, with filter-based methods, specifically the T-test having the best performance out of all methods (Tsai, 2010). Similarly, Faris et al., evaluating the impact of filter-based and feature selection methods on model performance, found that both methods improve performance, with wrapper-based RFE method proving the best performing, despite being computationally expensive (Faris et al., 2020). Another study found RFE to provide the best impact on performance after LASSO, which is an embedded FS method, and these types of methods do not fall into the scope of this research as embedded methods achieve a balance between the efficiency of filters and effectiveness of wrappers, and this research focuses primarily on evaluating and optimizing the performance of filter and wrapper feature selection methods for their application in the specific context of this study. Significant to this research, a study using different types of models trained on Taiwanese bankruptcy data treated with SMOTE, combined and evaluated by adding 3 filter-based feature selection methods, found RF and DT models with Mutual Information (MI) achieve the highest performance (Premalatha et al., 2023).

2.4 Selected Models

Based on the comprehensive literature review provided, the justification for the selection of DT and RF models for corporate bankruptcy prediction is supported by their ability in handling imbalanced data (Sisodia & Verma, 2018; Wang & Liu, 2021; Yotsawat et al., 2023), successful integration with FS methods to deliver an improvement in model performance (Premalatha et al., 2023), adaptability with hybrid resampling methods (Le, 2021; Le et al., 2018), and competitive performance with more complex models. While gradient boosting models like AdaBoost, XGBoost, and CatBoost have shown excellent results, RF and DT are not far behind in performance, especially considering that they achieve this performance with default hyperparameters (Vellamcheti & Singh, 2020; Faris et al., 2020; Le, 2021).

Moreover, DT is known for its simplicity and interpretability, which is a valuable aspect in financial modeling where understanding the decision process and the contributing factors is crucial. RF, while being more complex than DT, still retains a level of interpretability and adds the benefit of improved accuracy through ensemble learning.

2.5 Research Gap

The literature review has highlighted a noticeable gap in current research, specifically the interaction of hybrid resampling techniques and FS methods on predictive performance. The following studies evaluated resampling with FS methods and found it brought an improvement to predictive performance (Premalatha et al., 2023; Paraschiv et al., 2023). However, at the moment of writing, there is no study investigating the synergies between the hybrid resampling techniques and FS methods.

3. METHODOLOGY & EXPERIMENTAL SETUP

This section outlines the methodology adopted to address the research questions, encompassing an overview of the dataset and the methods applied for data analysis, cleaning, and preprocessing. It further elaborates on the algorithms and techniques used for prediction and discusses the evaluation process in detail. The list of software and packages used in this study is presented in Appendix A.

The methodological framework is illustrated in Figure 1. Section 3.1 details the exploratory data analysis, along with the data cleaning and preprocessing steps. For the training of classifiers (DT, RF) and for conducting and evaluating experiments, the dataset was divided into training and test sets using a stratified split to ensure a correct representation of classes in both subsets.

Initial baseline evaluations were established using default, untuned algorithms on data without employing hybrid resampling methods or feature selection techniques. Section 3.2 describes the implementation of hybrid resampling methods.

Section 3.3 examines the effects of integrating feature selection techniques into the chosen models in conjunction with resampling methods. Sections 3.4, 3.5, and 3.6 delve into the algorithms employed, the experimental procedures, and the evaluation metrics used for interpreting these experiments, respectively.

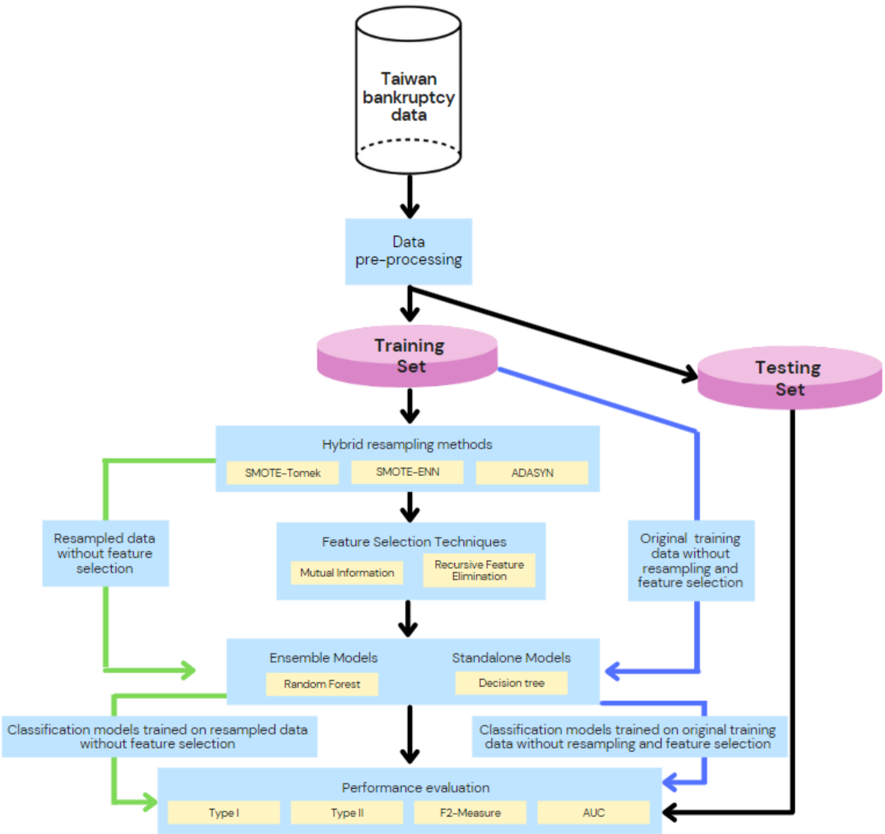


Figure 1: Overview of the methodology

3.1 Data pre-processing

In the context of this analysis, a systematic exploration of data features through the application of Exploratory Data Analysis serves as a foundational approach for comprehending and visualizing the Taiwanese bankruptcy dataset. EDA techniques implemented in this study such as distribution analysis, box plots, correlation matrices, and heatmaps.

Overview of the dataset: The dataset used in this study comprises financial and performance indicators collected from the Taiwan Economic Journal for the years 1999 to 2009. The dataset contains 96 features with 6819 instances, where each feature type is an integer. The dataset encompasses a diverse set of features that reflect various aspects of financial health, operational efficiency, profitability, and growth of companies. The features within the dataset can be categorized into 6 different types of financial metrics: financial ratios, book value and share information, employee and asset efficiency ratios, sales and cash flow ratios, and other miscellaneous indicators that don't fall into distinctly into other categories. For a comprehensive categorized breakdown of features within the dataset, see Appendix B.

Missing values, constant, duplicate, and negative values: Conducting a thorough check for missing and negative values is a fundamental step in data preprocessing to ensure data completeness, prevent biases, and enhance model performance. The Taiwanese dataset contains no missing values, duplicate values, or negative values.

Negative values are problematic in the context of financial and performance indicators. Certain financial metrics such as Current Ratio, Acid Test, Return on Total Assets, Return on Equity, etc., are expected to be positive, similarly, sales and cash flow ratios such as Sales Per Share, Net Sales, and Cash Flow-related metrics like Cash Flow Per Share are expected to be positive. Negative values for these metrics might indicate issues with accounting or financial reporting.

Checking and handling constant values in each column is important because columns with constant values can negatively impact the performance of machine learning algorithms and contribute little to the predictive power of a model. Therefore, they are removed as a feature from the dataset and the further analysis, specifically, column "Net Income Flag" is dropped.

Box Plots and Distributions: Figure 2 illustrates the class distribution of the target feature "Bankrupt?" revealing a highly imbalanced class distribution, specifically, there are 6599 instances in the non-bankrupt class and, only, 220 instances in the bankrupt class. Highly imbalanced datasets impact the performance of machine learning models by making them more biased towards the majority class. The purpose of this study is to ensure that the selected machine learning models effectively capture patterns in both minority and majority classes, thereby improving their utility in bankruptcy prediction.

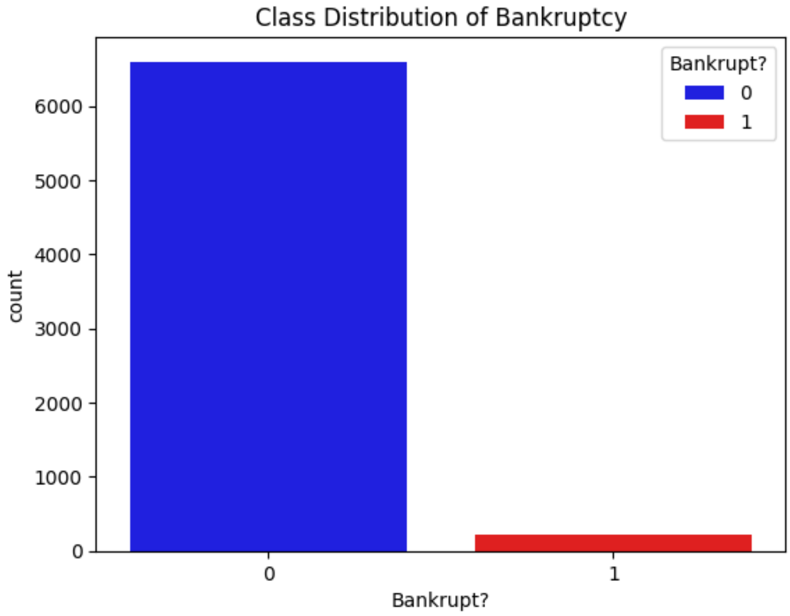


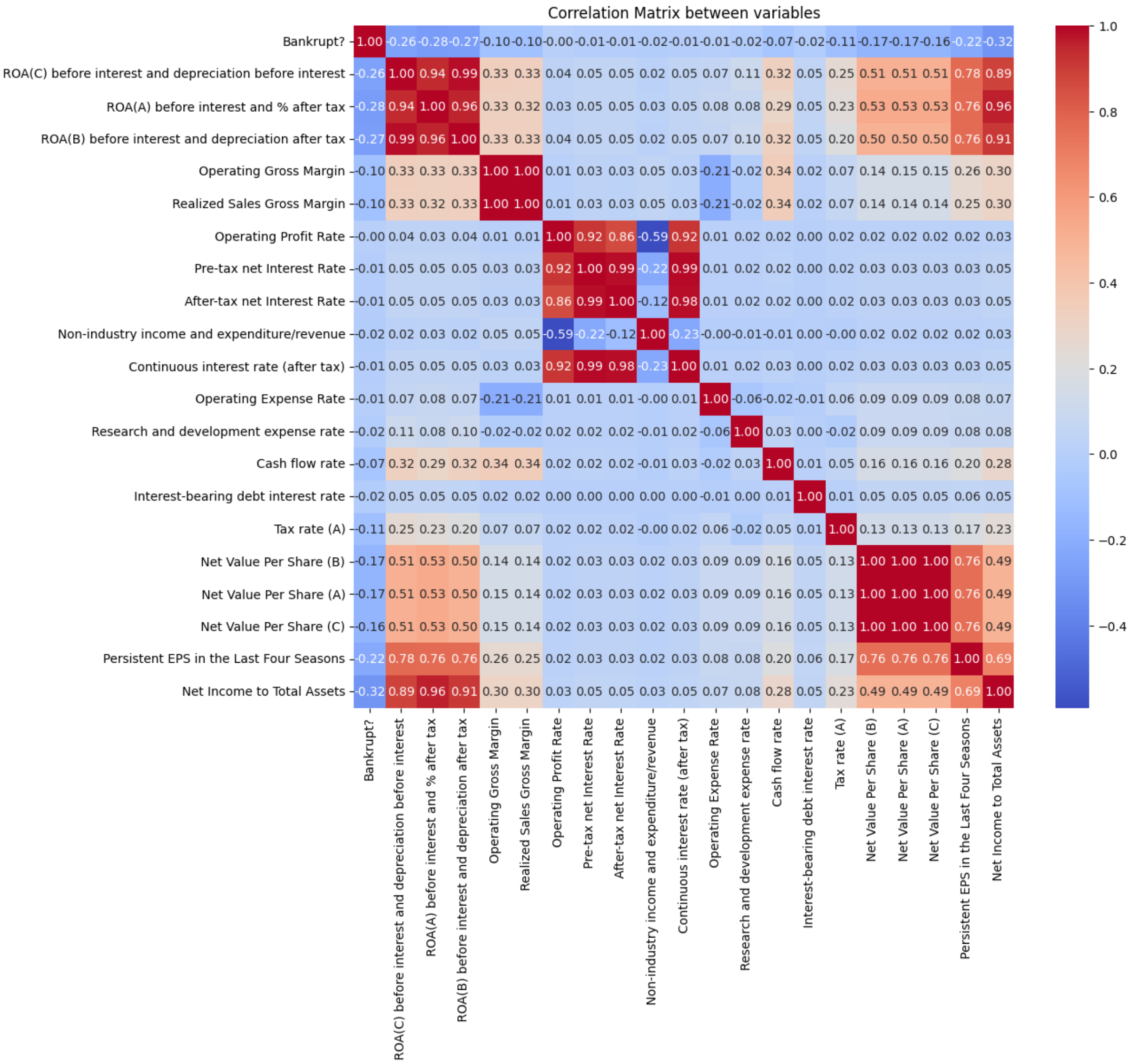
Figure 2: Class Distribution for target feature

Based on the data analysis, it is apparent that skewness is present in specific variables. However, DT is not sensitive to the scale of features or the distribution of individual variables. They make binary decisions at each node based on feature thresholds, and the order or scale of the features does not impact their performance (Premalatha et al., 2023).

Correlations: The correlation matrix in Figure 3 reveals that features were not highly correlated with the target feature ("Bankrupt?"). The feature with the strongest correlation to the target feature is "Net Income to Total Assets" with a correlation coefficient of 0.32, which illustrates that their relationship is not extremely strong and is considered mildly moderate.

Figure 3 shows features that exhibited strong correlations with each other, such as "ROA(C) before interest and depreciation before interest" with "ROA(C) before interest and % after-tax" with a correlation of 0.94. Despite some strong correlations between features, most have a weak correlation with each other, additionally, the correlation of all features with the target feature remained mildly moderate. Therefore, feature screening was not implemented despite the presence of some strong correlations due to the priority of retaining all available information, correlated features may better capture the complexities of the underlying data patterns.

Figure 3: Correlation Matrix



3.2 Hybrid Resampling Methods

3.2.1 Class Imbalance Problem

Bankruptcy prediction, like many other real-world datasets, exhibits the class imbalance problem, which refers to any dataset in which one class has most of the instances compared to the other class. They are referred to as the majority class and the minority class, respectively. In this study, the target feature ‘Bankrupt?’ presents itself with an extreme class imbalance, as 6599 companies are labeled non-bankrupt, and only 220 are labeled as bankrupt. This problem can negatively impact the performance of the selected classification algorithms.

To visualize the class imbalance problem, the balancing is presented below in equation 1, as a measure of class imbalance:

$$br_X = |X_{min}|/|X_{maj}| \tag{1}$$

The smaller the balancing ratio br_X, the more imbalanced the dataset is. Using the Taiwanese bankruptcy dataset, the bankrupt class is denoted as $|X_{min}|$ with 220 instances and the non-bankrupt class as $|X_{maj}|$ with 6599 data instances. The balancing ratio is approximately 0.033, which illustrates a very extreme class imbalance.

With a low br_X, machine learning models might become biased towards the majority class and underperform on the minority class. To address this problem, resampling techniques are used to create a new dataset X_{res}, where the balancing ratio $br_{X_{res}}$ is greater than the original balancing ratio br_X. These techniques oversample the minority class, under-sample the majority class, or combine them in a hybrid approach. The class imbalance problem is addressed by applying one of these techniques to the dataset before training the models.

To measure the degree of improvement of the hybrid resampling method on model performance, the following resampling techniques are implemented to address the class imbalance: SMOTE-ENN, SMOTE-Tomek, and ADASYN. An experimental study implementing the hybrid resampling methods found a substantial increase in predictive performance measured by AUC for all models, with Random Forest combined with SMOTE-ENN performing the best (Le, 2021).

3.2.2 SMOTE-ENN

SMOTE-ENN is a hybrid resampling technique that combines Synthetic Minority Oversampling Technique (SMOTE) with Edited Nearest Neighbors (ENN). Firstly, SMOTE is applied to generate synthetic samples for the minority class, followed by ENN which removed overlapping instances from the majority class.

SMOTE creates new synthetic samples for the minority class by connecting each minority class instance to one of its nearest neighbors and interpolating along the line segment. This interpolation depends on the similarity between instances of the minority feature space to create the synthetic samples. For each instance $x_i \in X_{min}$ the minority class, the *k*-nearest neighbors K_{x_i} are calculated based on Euclidean distance of x_i, using the *K* input set by the user. To create a synthetic instance for x_i, SMOTE randomly selects two samples: x_i' from the set x_i and x_i' from the minority class X_{min}. Equation 2 shows the feature vector of the newly created synthetic instance (Chawla, 2002):

$$x_{new} = x_i + (x_i' - x_i) * \delta \tag{2}$$

where x_i' is an element in X_i, such that $x_i' \in X_{min}$. The parameter δ controls the interpolation factor, determining the extent of movement along the line segment, and influences the similarity between the synthetic instance and its neighbors in the minority feature space.

Edited Nearest Neighbor (ENN) is an under-sampling technique that determines the *k*-nearest neighbors K_{x_i} for each instance and is calculated based on Euclidean distance of x_i, using the *K* input set by the user. If the neighbors belong to a different class from the class of the instance, then both the instance and the *k*-nearest neighbors are removed. The purpose of ENN is to eliminate instances where the majority class is surrounded by minority class instances, thereby reducing the likelihood of machine learning models exhibiting bias towards the majority class during training.

3.2.3 SMOTE-Tomek

This method combines the SMOTE and Tomek Links (TL). TL is a modification of Condensed Nearest Neighbors created by Tomek (Tomek, 1976), the integration of TL with SMOTE gave rise to the formulation of SMOTE-Tomek Links method. Firstly, using SMOTE this method generates synthetic samples for the minority class to balance the dataset, followed by TL which removes instances from the majority class that has the shortest Euclidian distance with the minority class instances.

A pair of data instances d(x_i,x_j) represents the Euclidian distance between x_i and x_j, where $x_i \in X_{min}$ and $x_j \in X_{maj}$, respectively. For this given pair of data instances, it can be considered a TL exclusively under the condition that no other data instance x_k exists such that the distance from x_i to x_k is less than the distance from x_i to x_j, or the distance from x_j to x_k is less than the distance from x_i to x_j. This condition is represented in equation 3:

$$\mathrm{d}(x_i, x_k) < \mathrm{d}(x_i, x_j) \; or \; \mathrm{d}(x_j, x_k) < \mathrm{d}(x_i, x_j) \tag{3}$$

3.2.4 ADASYN

ADASYN introduces a systematic adaptive technique, introducing variability in the number of synthetic instances created based on the distribution of the instances. Equation 4 shows how the number of generated synthetic instances is determined:

$$G = (|X_{maj}| - |X_{min}|) * \beta \tag{4}$$

β is the ratio of minority data instances to majority data instances desired after implementation, $|X_{maj}|$ are the total instances in the majority class, and $|X_{min}|$ is the total number of instances in the minority class. Followed by determining the *k*-nearest neighbors K_{x_i} for each minority class instance $x_i \in X_{min}$ and computing the r_i value, shown in the figure below:

$$r_i = \# majority/k \quad (5)$$

The r_i quantifies the proportion of majority class instances among the neighbors of the sample denoted K_{x_i} and determines the r_i between majority class instances in the k nearest neighbors and k for each sample $x_i \in X_{min}$. The r_i value offers insight into the class distribution in the local vicinity of that data point. For instance, r_i = 0.7 would indicate that 70% of the *k*-nearest neighbors K_{x_i} for each minority class instance x_i belongs to the majority class. Normalization shown in the equation below is applied to the r_i values for the sum of all the r_i values to equal to 1. $Nmin$ is the total number of minority class instances, and the equation normalizes the r_i values across all minority instances.

$$\hat{r_i} = \frac{r_i}{\sum_{j=1}^{Nmin} r_j} \quad (6)$$

Lastly, the equation below demonstrates how to calculate the number of synthetic examples in the minority class for each instance per neighborhood:

$$G_i = G\hat{r_i} \quad (7)$$

As mentioned above, a high r_i value indicates a high presence of majority class instances in the neighborhood, therefore, for those neighborhoods more synthetic minority class instances will be added (He et al., 2008).

3.3 Feature Selection

3.3.1 Feature Selection in the Context of Bankruptcy Prediction

Drawing from the established literature in the field that focuses on applying feature selection methods using imbalanced datasets in the context of bankruptcy prediction, the following filter-based and wrapper-based approaches are selected for this study: Mutual Information and Recursive Feature Elimination, respectively.

Through the choices made, one representative method from each type, this study aims to explore and compare the effectiveness of filter-based and wrapper-based approaches in enhancing the predictive performance of bankruptcy prediction models when confronted with imbalanced datasets, with Mutual Information and Recursive Feature Elimination chosen as representative methods for filter-based and wrapper-based approaches, respectively. This investigation seeks to provide insights into the comparative efficacy of these approaches, shedding light on their impact on feature selection and their overall influence on the performance of bankruptcy prediction models in imbalanced scenarios.

3.3.2 Mutual Information (MI)

MI quantifies the degree of association or dependence between two variables. The equation below shows how to calculate MI between two discreet random variables:

$$I(X;Y) = \sum_{x \in X} \sum_{y \in Y} P(x,y) \log\left(\frac{P(x,y)}{P(x)P(y)}\right) \tag{8}$$

where P(x, y) is the joint probability mass function of X and Y, and P(x) and P(y) are the marginal probability mass functions of X and Y, respectively. If I(X; Y)=0, it illustrates that the variables are independent, while larger values indicate higher dependence. MI is a useful criterion to determine the relevance of a given feature against the target variable.

3.3.3 Recursive Feature Elimination (RFE)

RFE iteratively removes the least important features from the dataset until the desired number of features is attained. The criterion for feature selection is based on the performance of the model, rather than evaluating all possible subsets of features.

3.4 Algorithms

The algorithms used in this study were Random Forest (RF) and Decision Tree (DT). This subsection explains the nature of the selected algorithms, along with their relevant hyperparameters.

3.4.1 Decision Tree

DT is a standalone model that independently structures and makes decisions based on the features present in the dataset. This supervised learning approach generates a tree structure model for either classification or regression problems.

The tree hierarchy starts with a root node, initiating a question about an input data feature, leading to branches representing the potential outcomes, further leading to new questions called child nodes. The recursive process repeats until a stopping criterion is met, concluding with a lead node responsible for the final prediction.

The training data forms the root node and undergoes division into smaller subsets, incorporating decision nodes and leaf nodes. The challenge lies in choosing the attribute for the root node at each level. The two methods of attribute selection that are employed are Gini Index and Information Gain.

3.4.2 Random Forest

Random forest is an ensemble algorithm that is based on stacking multiple decision trees to combine their predictions to improve overall performance and robustness. RF utilizes the technique of "bagging" to randomly select, with replacement, various samples from the dataset for each individual DT in the ensemble. In an RF, each tree within the ensemble is trained independently with its own randomly sampled dataset. Therefore, each tree contributes its vote to the final classification for a given sample (Bankrupt/Non-bankrupt).

3.4.3 Hyperparameter Tuning

The architecture of a machine learning model incorporates parameters acquired from training and hyperparameters set by the model builder. A machine learning model lacking tuned configurations fails to achieve optimal performance as a classifier. Therefore, it is recommended to experiment with various hyperparameters, delineated within a designated search space.

Various methods are available to systematically explore and test different configurations within a designated search space. GridSearch tests predefined combinations of hyperparameter values within a specified grid, resulting in a longer completion time. On the other hand, Random Search randomly samples hyperparameter values from the predefined search space, allowing for a faster completion time, but may lead to a suboptimal set of hyperparameter values.

This study aims to to rigorously explore various hyperparameter combinations to achieve optimal model performance, therefore, GridSearch is chosen for its exhaustive search within a specified grid.

The dataset is split into training and test sets, using stratified sampling which ensures that the proportion of the different classes in the target feature is maintained in the testing and test sets. The tuning process employs a 5-fold stratified cross-validation to find the optimal set of hyperparameters to be evaluated using the test set.

Fine-tuning DT: The process involves adjusting critical hyperparameters such as max_depth which controls the depth of the decision tree and their complexity. Setting the max_depth values to high may lead to overfitting, where the model becomes overly complex and captures noise in the data. On the other hand, values too low may result in underfitting, where the model cannot deal with complexity and misses important patterns.

Additional parameters such as min_samples_split and min_samples_leaf also play a role in controlling the tree's growth. These parameters govern the minimum number of samples required to split an internal node and the minimum number of samples required to be a leaf node, respectively.

As shown in Table 3, the selected search space for DTs is typically selected based on logical values and is informed by literature recommendations whenever possible.

Table 3: Hyperparameter Space for Decision Tree

Hyperparameter	Values
max_depth	5 to 30 (step 5)
max_features	sqrt, log2,
criterion	gini, entropy
min_samples_split	3 to 15 (step 3)
min_samples_leaf	3 to 15 (step 3)

Fine-tuning RF: As RF is based on stacking multiple independent DTs, the hyperparameters are like those of a DT. Additionally to the previously mentioned hyperparameters in DT, "n_estimators" and "bootstrap" are included. These hyperparameters are responsible for the number of trees in the forest, the splitting criterion used to construct the trees, and whether or not to bootstrap the data when building the trees, respectively.

Table 4: Hyperparameter Space for Random Forest

Hyperparameter	Values
max_depth	5 to 30 (step 5)
n_estimators	100 to 500 (step 100)
max_features	sqrt, log2,
criterion	gini, entropy
min_samples_split	3 to 15 (step 3)
min_samples_leaf	3 to 15 (step 3)
Bootstrap	True, False

3.4.4 Number of Features as a Hyperparameter

MI is often used to rank features based on their relevance to the target variable. The number of features selected is determined by selecting the top *k* features with the highest MI scores.

Similarly, to MI, RFE can also be employed to select several features, however, in a different way. RFE works recursively by removing the least important features, fitting the model again, and repeating until the stated number of features is reached.

The parameter *k*, representing the top number of features varies from 5 to 30 with a step size of 5 in this experiment. This approach is often part of a hyperparameter tuning process, where different values of *k* are tested to find the optimal number of features for supervised learning models and provide a way to evaluate their impact on predictive performance. Therefore, it was possible to select the number of top *k* features as a hyperparameter for both feature selection methods and add them to the other hyperparameters relevant to the supervised learning models, shown in the table below:

Table 5: Number of possible features

Parameter	Values
feature selection *k*	10 to 40 (step 10)

The supervised learning models will train on the specified number of the top k features, aiming to determine the best number of features for optimal predictive performance.

3.5 Experimental Set-up

This subsection provides a comprehensive review of the experimental methodology undertaken to address the sub-research questions, and finally, the main research question.

3.5.1 Model with imbalanced data

Firstly, the selected classifiers, DT and RF, are trained on the stratified training set to evaluate their performance without hybrid resampling methods and feature selection. The performance of these models will serve as a benchmark to assess the effectiveness of hybrid resampling methods and feature selection techniques in improving predictive performance. Measures such as Test F2 score, Test AUC score, Test Type I Error, and Test Type II Error, provide a foundational reference point. They enable a comprehensive evaluation of the identified models, offering insights into their strengths and weaknesses.

3.5.2 Models with resampled data

Secondly, to address the class imbalance in the stratified training set prior to model training, hybrid resampling techniques are employed, specifically, SMOTE-ENN, SMOTE-Tomek, and ADASYN. As mentioned in the literature review, these methods combine multiple resampling techniques to improve the performance of machine learning models on imbalanced datasets. The performance of DT and RF models is discussed in the context of the selected evaluation measures.

3.5.3 Models with resampled data and feature selection

Lastly, we use two feature selection methods, namely, MI and RFE, individually on the selected models, using each resampling method to find the optimal subset of features that maximizes predictive performance.

3.6 Performance evaluation metrics

After using 5-fold cross-validation, this study evaluates the performance of the prediction models. The performance of baseline DT and RF is evaluated without resampling and feature selection methods, then with resampling, and finally with both resampling and feature selection.

Referencing the related works in bankruptcy prediction in the context of imbalanced datasets guides our understanding of the necessary evaluation metrics in classification. Firstly, the use of standard metrics such as accuracy, precision, recall, and F1-score can be misleading in imbalanced domains as they favor the majority class and rarely correctly predict the minority class. Secondly, to address the imbalance problem, researchers must seek reliable evaluation metrics that measure the classifier's performance between classes.

Therefore, the primary evaluation metrics selected for this study is the F2-measure, and alongside Type I errors, Type II errors, and AUC, it is used to assess the performance of the selected models. The performance measurements selected for this experiment, are supported by the following studies in the domain of bankruptcy prediction (Le, 2018; Faris et al. 2020; Aly et al. 2022). A comprehensive explanation of selected evaluation metrics, including their formulas, is provided in Appendix C.

4. RESULTS

The outcomes of the experiments involving the selected classifiers are detailed in this section. The optimal sets of hyperparameters for each classifier at each stage of the experiment are presented in Appendix D.

4.1 Predictive performance of the selected models, without hybrid resampling methods and feature selection techniques.

Figure 4 illustrates the outcomes of the models' predictive performance trained only on the imbalanced dataset, establishing a baseline for assessing the effectiveness of future model enhancements. The DT model outperforms RF in terms of F2 score, AUC, and Type II error. The RF model achieved a Type I Error of 0.00 indicating that the model did not classify any instances of the negative class as positive incorrectly, moreover, RF's high score on Type II error of 0.84 means the model misclassified 84% of the actual positive instances. In terms of the AUC score, DT model's AUC score of 0.70 presents a good level of performance in distinguishing between the two classes. While not perfect, this score indicates that the DT model has a reasonably high probability of correctly differentiating between the positive and negative classes. The RF model's AUC score of 0.57 is closer to 0.5, which is the score of a random classifier (one that makes predictions by chance). This indicates that the RF model has a limited ability to distinguish between the two classes. It performs only slightly better than random guessing.

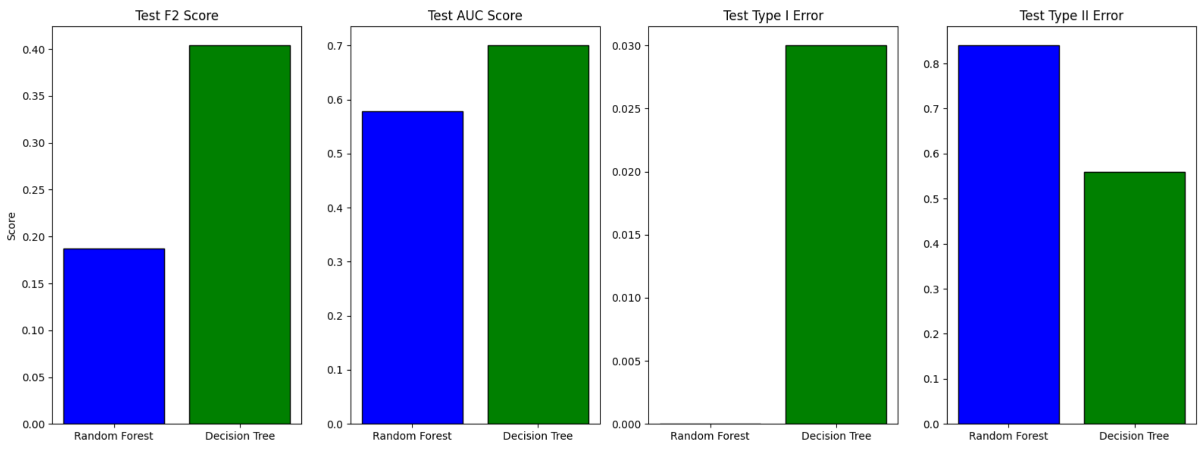


Figure 4: Baseline RF and DT

4.2 Predictive performance of the models with hybrid resampling methods, without feature selection techniques.

As presented in Figure 5, DT with SMOTE-ENN achieved an F2 score of 0.50 among resampling methods, while DT with SMOTE-Tomek and ADASYN achieved the same F2 score of 0.40. Similarly, in terms of the AUC score, DT with SMOTE-ENN achieved the highest score of 0.80, while DT with SMOTE-Tomek and ADASYN scored 0.70. Regarding the error analysis, for Type I error, DT with SMOTE-Tomek achieved the best score of 0.05, followed by ADASYN with a score of 0.06, and lastly by SMOTE-ENN with a score of 0.07. For Type II errors, DT with SMOTE-ENN with a score of 0.32 marginally outperformed ADASYN with a score of 0.34, SMOTE-Tomek performed the worst with a score of 0.52.

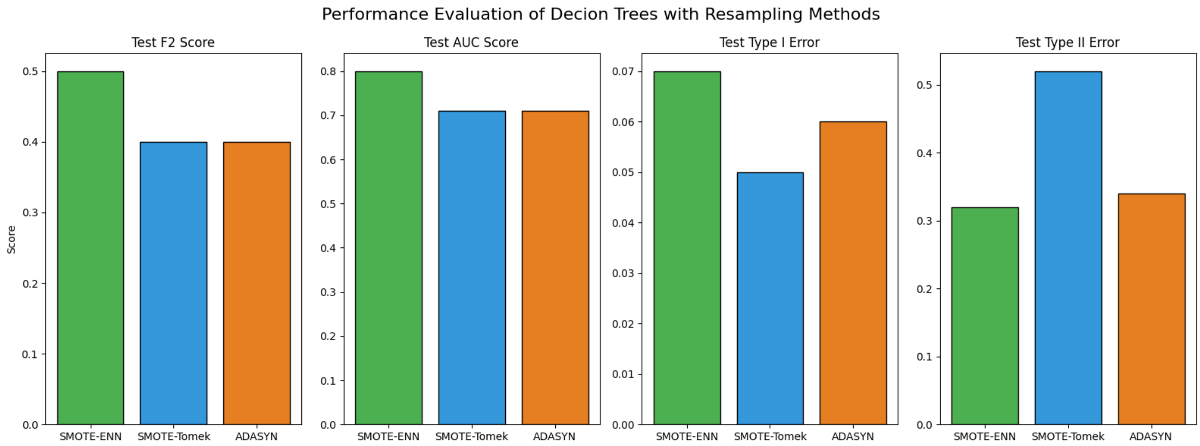


Figure 5: DT performance with hybrid resampling methods

As shown in Figure 6, RF with SMOTE-Tomek has the highest F2 score with 0.64 and the lowest F2 score for RF is shared by SMOTE-ENN and ADASYN with a score of 0.60. The highest AUC score of 0.84 is achieved by RF with SMOTE-ENN and SMOTE-Tomek, RF with ADASYN performed marginally worse with a score of 0.83. In terms of error analysis for the RF model, the lowest Type I error score of 0.03 is achieved by ADASYN, followed by SMOTE-Tomek with 0.04, and lastly SMOTE-ENN with a score of 0.05. The lowest Type II error of 0.23 is attained by SMOTE-ENN and SMOTE-Tomek, followed by ADASYN with a score of 0.29.

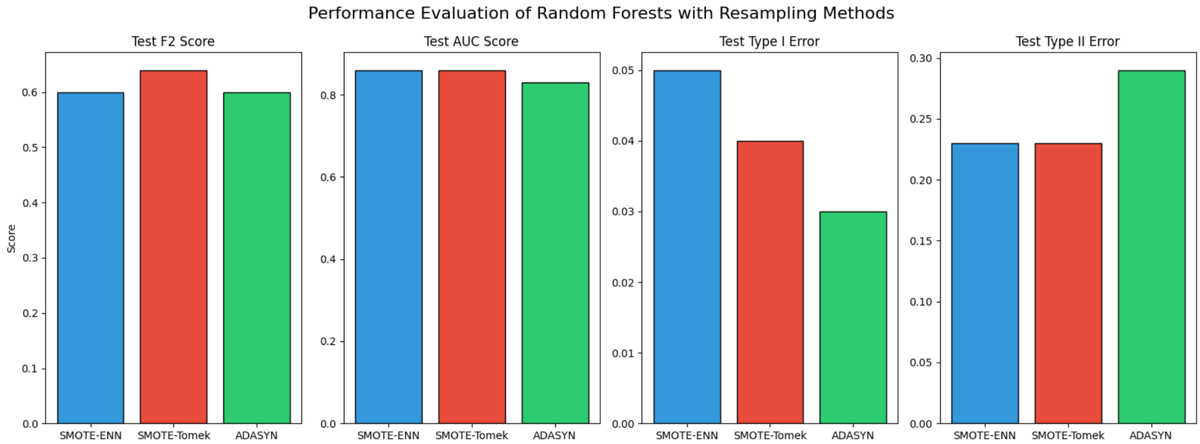


Figure 6: RF performance with hybrid resampling methods

Based on these results, for the DT model, SMOTE-ENN is found to be the most effective method overall, considering its top performance in both F2 and AUC scores. Although it has a slightly higher Type I error than SMOTE-Tomek, its significantly lower Type II error and higher AUC and F2 scores make it the preferable method for DT.

For the RF model, resampling method selection presents more complexity, whereas, SMOTE-Tomek has the highest F2 score, both SMOTE-ENN and SMOTE-Tomek share the highest AUC score and the lowest Type II error. ADASYN, though slightly lagging in AUC and F2 scores, has the lowest Type I error. Considering the overall balance of metrics, SMOTE-Tomek and SMOTE-ENN seem to be equally effective for the RF model, with a slight edge to SMOTE-Tomek due to its higher F2 score.

4.3 Predictive performance of the models with hybrid resampling methods and with feature selection techniques.

For hybrid resampling and FS methods, the F2 score for RF models with MI exhibit very similar scores, 0.63 for SMOTE-ENN, 0.62 for ADASYN and 0.61 for SMOTE-Tomek. Similarly, the highest F2 score for RF models with RFE is 0.64 with ADASYN, with SMOTE-ENN and SMOTE-Tomek sharing the score of 0.61. DT with SMOTE-Tomek and RFE scored the lowest F2 score of 0.44, moreover, the lowest scoring models for F2 use the RFE method. The performance for DT using MI performs better than RFE, specifically, ADASYN with MI score 0.54, with the other models marginally behind. For DTs using RFE, the F2 score is the lowest with SMOTE-Tomek at 0.44, followed by SMOTE-ENN at 0.48 and ADASYN at 0.49. Models utilizing ADASYN, tend to achieve higher F2 scores compared to the others.

RFs generally exhibit higher AUC scores for both MI and RFE with all the scores above 0.84, the highest being SMOTE-ENN with MI with a score of 0.88. RFE methods achieve lower Type I errors for both models. Models with SMOTE-ENN tend to have higher Type I errors for both models and FS methods, similarly SMOTE-Tomek and ADASYN achieve the same Type I error. RFE with DT has the highest Type II error with 0.43, followed by the other two resampling methods, RFE with DT underperforms the most on this parameter. MI with RF has the lowest Test Type II error.

Table 6: Performance of DT and RF with hybrid resampling and FS

Standalone Model	F2 score	AUC score	Type I Error	Type II Error
DT with SMOTE-ENN and MI	0.50	0.83	0.09	0.25
DT with SMOTE-Tomek and MI	0.53	0.84	0.08	0.23
DT with ADASYN and MI	0.54	0.85	0.08	0.20
DT with SMOTE-ENN and RFE	0.48	0.79	0.07	0.34
DT with SMOTE-Tomek and RFE	0.44	0.75	0.06	0.43
DT with ADASYN and RFE	0.49	0.78	0.05	0.38
RF with SMOTE-ENN and MI	0.63	0.88	0.06	0.18
RF with SMOTE-Tomek and MI	0.61	0.85	0.04	0.25
RF with ADASYN and MI	0.62	0.85	0.04	0.25
RF with SMOTE-ENN and RFE	0.61	0.86	0.05	0.23
RF with SMOTE-Tomek and RFE	0.61	0.84	0.04	0.27
RF with ADASYN and RFE	0.64	0.86	0.04	0.25

Drawing from these results, as presented in Table 6, the study concludes that the Decision Tree model employing ADASYN and Mutual Information (MI) emerges as the most effective, achieving superior results across multiple metrics, including the highest F2 score of 0.54, an AUC score of 0.85, and the lowest Type II error at 0.20, alongside a competitive Type I error of 0.08. Similarly, in the Random Forest models, the combination of SMOTE-ENN with MI stands out as the top performer, demonstrating the highest F2 score of 0.63 and an AUC score of 0.88, complemented by a low Type I error of 0.06 and the lowest Type II error of 0.18, thereby underscoring its robustness and efficiency in predictive accuracy.

5. Discussion

5.1 Results Discussion

The study's findings on the effectiveness of specific combinations of resampling methods and feature selection techniques directly contribute to the field of bankruptcy prediction. This is especially relevant given the literature's identification of class imbalance as a persistent challenge in this domain (Wang & Liu, 2021; Kumar, 2007).

The results of this study align with the literature that emphasizes the effectiveness of hybrid resampling techniques in dealing with imbalanced datasets, particularly in bankruptcy prediction. For DT models, SMOTE-ENN outperforms other resampling methods on the F2, AUC and Type II error scores, for the Type I error the score is slightly higher than SMOTE-Tomek and ADASYN, however, the difference between the lowest and highest score is 0.02. These results contradict the finding of a similar study, which found DT with SMOTE-Tomek to outperform other hybrid resampling techniques (Le et al., 2018). Similarly, studies found RF models to outperform other hybrid resampling when utilized with SMOTE-ENN (Le et al., 2018; Le, 2021), which is not supported by this study, in the context of the overall performance metrics, both SMOTE-Tomek and SMOTE-ENN demonstrate comparable effectiveness for the RF model. However, SMOTE-Tomek has a slight advantage, primarily because of its superior F2 score. RF models with resampling outperform DT models across all measures, showcasing the effectiveness of ensemble methods and resampling in combination. These results are consistent with the findings from other studies that evaluated a variety of ensemble and standalone models and found RF models with resampling outperforming DT models (Sisodia & Verma, 2018; Vellamcheti & Singh, 2020).

The implementation of FS techniques combined with hybrid resampling methods is successful in improving model performance. The findings of this study corroborate previous research, emphasizing the importance of FS methods in improving model performance (Lin et al., 2018; Faris et al., 2020). The ADASYN with MI method in DT models and SMOTE-ENN with MI in RF models emerge as the most effective combinations, showcasing superior performance across multiple metrics, including F2 scores, AUC scores, and Type II errors, these results align with the findings of Premalatha et al. (2023). This study contributes significantly to the scientific field by demonstrating that while wrapper-based methods (RFE) lead to lower Type I errors, filter-based methods (MI) generally offer better overall performance in key metrics like F2 score, AUC score, and Type II errors, a valuable contribution to the domain of financial risk prediction.

In summary, the study not only reinforces existing theories but also offers practical, empirically tested strategies for improving bankruptcy prediction models, an advancement with significant implications for financial risk assessment and economic stability. Moreover, the finding can be beneficial to any domain utilizing real-world imbalanced datasets.

5.2 Limitations

This study has certain limitations that must be recognized.

Firstly, within the scope of this research, only financial and accounting information was evaluated, however, there are other factors influencing corporate bankruptcies, such as crowd psychology, effects of speculations, and government policies.

Secondly, given the time constraints, only the minimum number of features could be selected from 10 to 40 features to evaluate the models' performance, which might not include the full number of necessary relevant variables that could find the best possible predictive accuracy of the classifiers.

Thirdly, to evaluate the applicability of these findings, another dataset should be used for evaluation for the validation of its generalizability and robustness across datasets, and potentially, other domains.

5.3 Relevance

Enhancing the accuracy of bankruptcy prediction models has significant implications for economic stability and financial decision-making. This study's findings offer potential strategies for financial institutions and regulatory bodies to improve their predictive models, leading to more informed risk assessment and decision-making processes, thereby safeguarding economic interests.

5.4 Future Work

The future work is to apply the proposed methodology to various other datasets to ensure the reliability of results. Moreover, other FS methods can be used to evaluate their impact on performance. An interesting area for research is the application of cost-sensitive learning, considering the potential influence of varying misclassification costs on the overall effectiveness of the proposed approach. Also, it is of value to compare the performance of specific under or over-sampling methods with hybrid resampling methods combined with FS to evaluate respective contributions in addressing class imbalance and enhancing predictive capabilities. Moreover, future research could explore the application of these techniques in other domains facing class imbalance problems. Additionally, further investigation into the scalability of these methods in larger datasets or their application in different financial contexts could provide broader insights.

Most importantly, future research could also focus on further reducing Type I and Type II errors and exploring the impact of these resampling and feature selection techniques in dynamic, real-time prediction environments.

6. CONCLUSION

This section provides the answers to the main research question of this study.

6.1 To what extent does the implementation of hybrid resampling techniques with feature selection methods enhance the predictive performance of supervised models on corporate insolvency when dealing with imbalanced datasets?

For baseline classifier performance, the results demonstrate that both DT and RF exhibit a significant deficiency in predicting corporate bankruptcies, with high Type II error rates (0.84 for RF and 0.56 for DT). Due to the nature of the imbalanced data, both models show low Type I errors, with RF having no false positive. DT outperforms RF in terms of F2 score, AUC score, and Type II error.

6.2 What is the impact of implementing hybrid resampling techniques on the predictive performance of Decision Tree and Random Forest models?

Hybrid resampling techniques, including SMOTE-Tomek, SMOTE-ENN, and ADASYN, were applied to address the class imbalance, resulting in improved performance for both models across all measures. For DT model, SMOTE-ENN was found to be the most effective resampling method for the DT model, considering its high F2 and AUC scores, although it had a slightly higher Type I error than SMOTE-Tomek. In the case of the RF model, selecting the most effective resampling method is more complex. While SMOTE-Tomek had the highest F2 score, both SMOTE-ENN and SMOTE-Tomek achieved the highest AUC score and the lowest Type II error. ADASYN, on the other hand, had the lowest Type I error. RF models with resampling consistently outperform DT models across all measures, aligning with previous studies.

6.3 How does the integration of feature selection methods after implementing hybrid resampling techniques impact the predictive performance of bankruptcy prediction models?

Mutual Information (MI) and Recursive Feature Elimination (RFE) were applied to resampled data before training. AUC score of DT models with resampling benefits from FS methods, particularly MI. The addition of FS methods generally results in a slight performance detriment to Type I error, however, in certain cases that can be tolerated for better Type II error scores. FS methods contribute to a decrease in Type II errors for both DT and RF models, except for RFE which has a marginally negative effect on the score. MI achieves the lowest Type II error scores for both RF and DT models across resampling methods. RFE generally doesn't achieve a positive improvement in Type II error, except in specific cases.

To conclude, hybrid resampling techniques combined with FS methods exhibit a generally positive impact on predictive performance of both models. While these approaches contribute to improved classifier performance, particularly in mitigating Type II errors, the effectiveness varies across resampling and feature selection methods. The most effective FS method is the filter-based MI, which delivers the best results for both models across all measures. Moreover, the study highlights the superior performance of RF over DT in the context of corporate bankruptcy predictions. The findings contribute valuable insights into the interplay between resampling and feature selection in bankruptcy prediction models.

7. References

Altman, E. (1968). Financial ratios, discriminant analysis and the prediction of corporate bankruptcy. *Journal of Finance*, 589-609.

Aly, S., Alfonse, M., & Salem, A. (2022). Intelligent model for enhancing the bankruptcy prediction with imbalanced data using oversampling and CatBoost. *International Journal of Intelligent Computing and Information Sciences*, *22*(3), 92-108. https://doi.org/10.21608/ijicis.2022.105654.1138

Beaver, W. (1966). Financial ratios as predictors of failure. *Journal of Accounting Research*, 71-111.

Bennin, K., Keung, J., Phannachitta, P., Monden, A., & Mensah, S. M. (2018). Diversity-based oversampling approach to alleviate the class imbalance issue in software defect prediction. *IEEE Transactions on Software Engineering*, 534-550.

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, *16*, 321-357.

Faris, H., Abukhurma, R., Almanaseer, W., Saadeh, M., Mora, A. M., Castillo, P. A., & Aljarah, I. (2020). Improving financial bankruptcy prediction in a highly imbalanced class distribution using oversampling and ensemble learning: A case from the Spanish market. *Progress in Artificial Intelligence*, *9*(1), 31-53. https://doi.org/10.1007/s13748-019-00197-9

Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., Kern, R., Picus, M., Hoyer, S., van Kerkwijk, M. H., Brett, M., Haldane, A., del Río, J. F., Wiebe, M., Peterson, P., . . . Oliphant, T. E. (2020). *Array programming with NumPy. Nature, 585(7825),* 357–362. https: //doi.org/10.1038/s41586-020-2649-2

He, H., Bai, Y., Garcia, E. A., & Li, S. (2008). ADASYN: Adaptive synthetic sampling approach for imbalanced learning. In *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)* (pp. 1322-1328). Hong Kong. https://doi.org/10.1109/IJCNN.2008.4633969

Herndon, N., & Caragea, D. (2016). A study of domain adaptation classifiers derived from logistic regression for the task of splice site prediction. *IEEE Transactions on Nanobioscience*, 75-83.

Hung, D., & Binh, T. V. (2021). Data mining for bankruptcy prediction: An experiment in Vietnam. 175-184. https://doi.org/10.15439/2021KM30

Hunter, J. D. (2007). Matplotlib: A 2d graphics environment. *Computing in Science & Engineering, 9(3)*, 90–95. https://doi.org/10.1109/MCSE. 2007.55

Kubat, M. (2000). Addressing the curse of imbalanced training sets: One-sided selection. In *Fourteenth International Conference on Machine Learning* (pp. 186-197).

Kumar, P. R. (2007). Bankruptcy prediction in banks and firms via statistical and intelligent techniques. *European Journal of Operational Research*, 5-28.

Le, T., Son, L. H., Vo, M. T., Lee, M. Y., & Baik, S. W. (2018). A cluster-based boosting algorithm for bankruptcy prediction in a highly imbalanced dataset. *Symmetry, 10*(7), 256.

Le, T. (2021). A comprehensive survey of imbalanced learning methods for bankruptcy prediction. *IET Communications, 16*(5), 433-441. https://doi.org/10.1049/cmu2.1226

Lemaitre, G., Nogueira, F., & Aridas, C. K. (2017). Imbalanced-learn: A Python toolbox to tackle the curse of imbalanced datasets in machine learning. *Journal of Machine Learning Research, 18*(17), 1-5. Retrieved from http://jmlr.org/papers/v18/16-365

Lin, W. C., Tsai, C. F., Hu, Y. H., & Jhang, J. S. (2018). Clustering-based undersampling in class-imbalanced data. *Information Sciences, 409*, 17-26.

Luo, J., & Xiao, Q. (2017). A novel approach for predicting microRNA-disease associations by unbalanced bi-random walk on heterogeneous network. *Journal of Biomedical Informatics*, 194-203.

Ohlson, J. (1980). Financial ratios and the probabilistic prediction of bankruptcy. *Journal of Accounting Research, 18*, 109-131.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research, 12*, 2825-2830.

Paraschiv, F., Schmid, M., & Wahlstrøm, R. R. (2023). Bankruptcy prediction of privately held SMEs using feature selection methods. SSRN 3911490. Retrieved from https://ssrn.com/abstract=3911490

Premalatha, G., Priyanka, R., & Chaitya, K. (2023). Feature selection for predicting bankruptcy: Comparative analysis. In *2023 Fifth International Conference on Electrical, Computer and Communication Technologies (ICECCT)* (pp. 1-5).

Sisodia, D. S., & Verma, U. (2018). The impact of data re-sampling on learning performance of class imbalanced bankruptcy prediction models. *International Journal on Electrical Engineering and Informatics*, *10*(2), 442-458.

Tomek, I. (1976). Two modifications of CNN. *IEEE Transactions on Systems, Man, and Cybernetics*, *6*(11), 769-772.

Tsai, C.-F. (2010). Feature selection in bankruptcy prediction: A comparative study. *Expert Systems with Applications*, *37*(5), 3508-3517.

Tsai, C.-F., Hsu, Y.-F., & Yen, D. C. (2014). A comparative study of classifier ensembles for bankruptcy prediction. *Applied Soft Computing*, *24*, 977-984. https://doi.org/10.1016/j.asoc.2014.08.047

Vellamcheti, S., & Singh, P. (2020). Class imbalance deep learning for bankruptcy prediction. In *First International Conference on Power, Control and Computing Technologies* (pp. 421-425). https://doi.org/10.1109/ICPC2T48082.2020.9071460

Waskom, M. L. (2021). Seaborn: Statistical data visualization. *Journal of Open Source Software*, *6(60)*, 3021. https://doi.org/10.21105/joss.03021

Wang, G., Ma, J., & Yang, S. (2014). An improved boosting based on feature selection for corporate bankruptcy prediction. *Expert Systems with Applications*, *41*(5), 2353-2361. https://doi.org/10.1016/j.eswa.2013.09.033

Wang, Y., & Liu, Y. (2021). Undersampling bankruptcy prediction: Taiwan bankruptcy data. *PLOS ONE*, *16*(7), e0254030. https://doi.org/10.1371/journal.pone.0254030

Yotsawat, W., Phodong, K., Promrat, T., & Wattuya, P. (2023). Bankruptcy prediction model using cost-sensitive extreme gradient boosting in the context of imbalanced datasets. *International Journal of Electrical and Computer Engineering (IJECE)*, *13*(4), 4683-4691.

Zakaryazad, A., & Duman, E. (2016). A profit-driven Artificial Neural Network (ANN) with applications to fraud detection and direct marketing. *Neurocomputing*, 121–131.

Zoričák, M., Gnip, P., Drotár, P., & Gazda, V. (2020). Bankruptcy prediction for small- and medium-sized companies using severely imbalanced datasets. *Economic Modelling, 84*, 165-176.

8. Appendix

Appendix A

The data cleaning, preprocessing, and modeling were performed using Python (Version 3.9.13). The following packages and libraries were utilized:

- Scikit-learn (sklearn) for various machine learning functionalities including metrics calculation (fbeta_score, roc_auc_score, confusion_matrix, make_scorer), classifiers (DecisionTreeClassifier, RandomForestClassifier), and model selection tools (GridSearchCV, train_test_split) (Pedregosa et al., 2011).

- Imbalanced-learn (imblearn) for addressing class imbalance issues using techniques like Pipeline, SMOTEENN, SMOTETomek, and ADASYN (Lemaitre et al., 2017).

- Pandas (pd) for data manipulation and analysis (development team, 2020).

- Matplotlib (plt) and Seaborn (sns) for data visualization (Hunter, 2007; Waskom, 2020).

- NumPy (np) for numerical computations (Harris et al., 2020).

- Scikit-learn's SelectKBest and mutual_info_classif for feature selection, along with Recursive Feature Elimination (RFE) (Pedregosa et al., 2011).

Appendix B

Feature Name	Categorization
X1 Cost of Interest-bearing Debt	Financial Ratios
X2 Cash Reinvestment Ratio	Financial Ratios
X3 Current Ratio	Financial Ratios
X4 Acid Test	Financial Ratios
X5 Interest Expenses/Total Revenue	Financial Ratios
X6 Total Liability/Equity Ratio	Financial Ratios
X7 Liability/Total Assets	Financial Ratios
X8 Interest-bearing Debt/Equity	Financial Ratios
X9 Contingent Liability/Equity	Financial Ratios
X10 Operating Income/Capital	Financial Ratios
X11 Pretax Income/Capital	Financial Ratios
X12 Working Capital to Total Assets	Financial Ratios
X13 Quick Assets/Total assets	Financial Ratios
X14 Current Assets/Total Assets	Financial Ratios
X15 Cash/Total Assets	Financial Ratios
X16 Quick Assets/Current Liability	Financial Ratios
X17 Cash/Current Liability	Financial Ratios
X18 Current Liability to Assets	Financial Ratios
X19 Operating Funds to Liability	Financial Ratios
X20 Inventory/Working Capital	Financial Ratios
X21 Inventory/Current Liability	Financial Ratios
X22 Current Liabilities/Liability	Financial Ratios
X23 Working Capital/Equity	Financial Ratios
X24 Current Liabilities/Equity	Financial Ratios
X25 Long-term Liability to Current Assets	Financial Ratios
X26 Current Liability to Current Assets	Financial Ratios
X27 One if Total Liability exceeds Total Assets	Financial Ratios
X28 Equity to Liability	Financial Ratios
X29 Equity/Total Assets	Financial Ratios
X30 (Long-term Liability+Equity)/Fixed Assets	Financial Ratios
X31 Fixed Assets to Assets	Financial Ratios
X32 Current Liability to Liability	Financial Ratios
X33 Current Liability to Equity	Financial Ratios
X34 Equity to Long-term Liability	Financial Ratios
X35 Liability to Equity	Financial Ratios
X36 Degree of Financial Leverage	Financial Ratios
X37 Interest Coverage Ratio	Financial Ratios
X38 Operating Expenses/Net Sales	Financial Ratios
X39 (Research and Development Expenses)/Net Sales	Financial Ratios
X40 Effective Tax Rate	Financial Ratios
X41 Book Value Per Share(B)	Book Value and Share Information
X42 Book Value Per Share(A)	Book Value and Share Information
X43 Book Value Per Share(C)	Book Value and Share Information
X44 Cash Flow Per Share	Book Value and Share Information

X45 Sales Per Share	Book Value and Share Information
X46 Operating Income Per Share	Book Value and Share Information
X47 Sales Per Employee	Employee and Asset Efficiency Ratios
X48 Operation Income Per Employee	Employee and Asset Efficiency Ratios
X49 Fixed Assets Per Employee	Employee and Asset Efficiency Ratios
X50 total assets to GNP price	Employee and Asset Efficiency Ratios
X51 Return On Total Assets(C)	Employee and Asset Efficiency Ratios
X52 Return On Total Assets(A)	Employee and Asset Efficiency Ratios
X53 Return On Total Assets(B)	Employee and Asset Efficiency Ratios
X54 Gross Profit /Net Sales	Profitability and Performance Ratios
X55 Realized Gross Profit/Net Sales	Profitability and Performance Ratios
X56 Operating Income /Net Sales	Profitability and Performance Ratios
X57 Pre-Tax Income/Net Sales	Profitability and Performance Ratios
X58 Net Income/Net Sales	Profitability and Performance Ratios
X59 Net Non-operating Income Ratio	Profitability and Performance Ratios
X60 Net Income-Exclude Disposal Gain or Loss/Net Sales	Profitability and Performance Ratios
X61 EPS-Net Income	Profitability and Performance Ratios
X62 Pretax Income Per Share	Profitability and Performance Ratios
X63 Retained Earnings to Total Assets	Profitability and Performance Ratios
X64 Total Income to Total Expenses	Profitability and Performance Ratios
X65 Total Expenses to Assets	Profitability and Performance Ratios
X66 Net Income to Total Assets	Profitability and Performance Ratios
X67 Gross Profit to Sales	Profitability and Performance Ratios
X68 Net Income to Stockholder's Equity	Profitability and Performance Ratios
X69 One if Net Income is Negative for the Last Two Years; Zero Otherwise	Other Indicator
X70 (Inventory +Accounts Receivables) /Equity	Other Indicator
X71 Total Asset Turnover	Sales and Cash Flow Ratios
X72 Accounts Receivable Turnover	Sales and Cash Flow Ratios
X73 Days Receivable Outstanding	Sales and Cash Flow Ratios
X74 Inventory Turnover	Sales and Cash Flow Ratios
X75 Fixed Asset Turnover	Sales and Cash Flow Ratios
X76 Equity Turnover	Sales and Cash Flow Ratios
X77 Current Assets to Sales	Sales and Cash Flow Ratios
X78 Quick Assets to Sales	Sales and Cash Flow Ratios
X79 Working Capital to Sales	Sales and Cash Flow Ratios
X80 Cash to Sales	Sales and Cash Flow Ratios
X81 Cash Flow to Sales	Sales and Cash Flow Ratios
X82 No-credit Interval	Sales and Cash Flow Ratios
X83 Cash Flow from Operating/Current Liabilities	Sales and Cash Flow Ratios
X84 Cash Flow to Total Assets	Cash Flow and Growth Ratios

X85 Cash Flow to Liability	Cash Flow and Growth Ratios
X86 CFO to Assets	Cash Flow and Growth Ratios
X87 Cash Flow to Equity	Cash Flow and Growth Ratios
X88 Realized Gross Profit Growth Rate	Cash Flow and Growth Ratios
X89 Operating Income Growth	Cash Flow and Growth Ratios
X90 Net Income Growth	Cash Flow and Growth Ratios
X91 Continuing Operating Income after Tax Growth	Cash Flow and Growth Ratios
X92 Net Income-Excluding Disposal Gain or Loss Growth	Cash Flow and Growth Ratios
X93 Total Asset Growth	Cash Flow and Growth Ratios
X94 Total Equity Growth	Cash Flow and Growth Ratios
X95 Return on Total Asset Growth	Cash Flow and Growth Ratios

Appendix C

Confusion matrix, presented in table 6, is the most used metric to evaluate the performance of machine learning models in binary classification problems. The matrix is a clear representation of the classification outcomes. Based on this table, a variety of evaluation measures that are used in this study can be calculated:

Table 7: Confusion matrix

	Predicted Bankrupt	Predicted Non-bankrupt
Actual bankrupt	True positive (TP)	False negative (FN)
Actual non-bankrupt	False positive (FP)	True negative (TN)

These rates offer insights into the performance as it related to bankrupt and non-bankrupt classification outcomes. Moreover, the two-by-two confusion matrix is used as a binary evaluation metric to calculate the total number of correctly classified divided by the total number of instances. This metric is called Accuracy, and its formula is shows in equation 9:

$$Accuracy = \frac{TP + TN}{total\ number\ of\ outcomes} \tag{9}$$

Even though accuracy is used as a standard evaluation metric, for imbalanced datasets accuracy of minority class decreases as balancing ratio br_X decreases, indicating a more imbalanced distribution, therefore accuracy becomes heavily biased towards the majority class. Therefore, understanding the specific type of misclassification taking place is essential for a comprehensive evaluation of the selected models' performance and its real-world implications.

False Positive Rate (FPR), representing the Type I error determines the rate of false positives, instances where the model erroneously classified a company as bankrupt that is not facing such a situation. Calculation for FPR is given below in equation 10:

$$Type\ I\ error\ (FPR) = 1 - TNR = 1 - Specificity \tag{10}$$

False Negative Rate (FNR), representing Type II error, reveals the rate of false negatives, this rate calculates the misclassifications to identify companies that are genuinely bankrupt. The formula for FNR is given in the equation below:

$$Type\ II\ error\ (FNR) = 1 - TPR = 1 - Sensitivity \tag{11}$$

Minimizing false negatives is crucial in bankruptcy prediction, therefore, F2-measure is chosen as a metric to strike a balance between precision and recall, with an emphasis on reducing false negatives. The Fβ-Measure is an extension of the F1 measure that introduces sets β to 2 to emphasize recall and minimize false negatives, which makes this metric the F2 measure, given in the equation below:

$$F\beta = \frac{(1 + \beta^2) \times \text{Precision} \times \text{Recall}}{\beta^2 \times \text{Precision} + \text{Recall}} \tag{12}$$

Precision is the ratio of true positives (TP) to the sum of true positives and false positives (FP). Recall is the ratio of true positives to the sum of true positives and false negatives (FN). Equations for both are given below:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{13}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{14}$$

The F2-measure is employed as the primary scoring metric for assessing the performance of the selected supervised learning models, particularly to address the importance of minimizing the false negatives in the bankruptcy prediction context.

Moreover, AUC is a fundamental measurement of the predictive model performance that is effective with datasets exhibiting an extreme class imbalance. It provides a concise summary of the model performance in distinguishing between positive and negative instances across various decision thresholds. AUC is derived from the Receiver Operating Characteristic (ROC) curve, which is a representation of the tradeoff between the TPR and FPR across various threshold values.

Appendix D

Model	Optimal Hyperparameters
Decision Tree Base Model	'criterion': 'entropy', 'max_depth': 15, 'max_features': 'log2', 'min_samples_leaf': 3, 'min_samples_split': 3
Random Forest Base Model	'criterion': 'gini', 'max_depth': 10, 'max_features': 'sqrt', 'min_samples_leaf': 3, 'min_samples_split': 3, 'n_estimators': 50, 'bootstrap': 'False'
Decision Tree + SMOTE-EEN	'criterion': 'entropy', 'max_depth': 10, 'max_features': 'sqrt', 'min_samples_leaf': 3, 'min_samples_split': 3
Decision Tree + SMOTE-Tomek	'criterion': 'entropy', 'max_depth': 15, 'max_features': 'sqrt', 'min_samples_leaf': 3, 'min_samples_split': 3
Decision Tree + ADASYN	'criterion': 'entropy', 'max_depth': 10, 'max_features': 'log2', 'min_samples_leaf': 3, 'min_samples_split': 3
Random Forest + SMOTE-EEN	'criterion': 'entropy', 'max_depth': 20, 'max_features': 'sqrt', 'min_samples_leaf': 3, 'min_samples_split': 3, 'n_estimators': 200, 'bootstrap': 'True'
Random Forest + SMOTE-Tomek	'criterion': 'entropy', 'max_depth': 25, 'max_features': 'sqrt', 'min_samples_leaf': 3, 'min_samples_split': 3, 'n_estimators': 200, 'bootstrap': 'False'
Random Forest + ADASYN	'criterion': 'entropy', 'max_depth': 25, 'max_features': 'sqrt', 'min_samples_leaf': 3, 'min_samples_split': 3, 'n_estimators': 500, 'bootstrap': 'True'
Decision Tree + SMOTE-EEN DT with MI	'criterion': 'entropy', 'max_depth': 10, ' max_features': 'sqrt', 'min_samples_leaf': 3, 'min_samples_split': 3, 'selectkbest__k': 40
Decision Tree + SMOTE-EEN with RFE	'criterion': 'entropy', 'max_depth': 10, 'max_features': 'sqrt', 'min_samples_leaf': 3, 'min_samples_split': 3, 'n_features_to_select': 40
Decision Tree + SMOTE-Tomek with MI	'criterion': 'gini', 'max_depth': 10, 'max_features': 'sqrt', 'min_samples_leaf': 3, 'min_samples_split': 3, 'selectkbest__k': 40

Decision Tree + ADASYN with MI	'criterion': 'entropy', 'max_depth': 10, ' max_features': 'sqrt', ' min_samples_leaf': 3, 'min_samples_split': 9, 'selectkbest__k': 40
Decision Tree + ADASYN with RFE	'criterion': 'entropy', 'max_depth': 15, 'max_features': 'log2', 'min_samples_leaf': 3, 'min_samples_split': 3, 'n_features_to_select': 40
Random Forest + SMOTE-ENN with MI	'criterion': 'entropy', 'max_depth': 15, 'max_features': 'sqrt', 'min_samples_leaf': 3, 'min_samples_split': 3, 'n_estimators': 500, 'bootstrap': 'False', 'selectkbest__k': 40
Random Forest + SMOTE-ENN with RFE	'criterion': 'entropy', 'max_depth': 20, 'max_features': 'sqrt', 'min_samples_leaf': 3, 'min_samples_split': 3, 'n_estimators': 500, 'bootstrap': 'False', 'n_features_to_select': 40
Random Forest + SMOTE-Tomek with MI	'criterion': 'entropy', 'max_depth': 20, 'max_features': 'sqrt', 'min_samples_leaf': 3, 'min_samples_split': 3, 'n_estimators': 300, 'bootstrap': 'True', 'selectkbest__k': 40
Random Forest + SMOTE-Tomek with RFE	'criterion': 'entropy', 'max_depth': 15, 'max_features': 'sqrt', 'min_samples_leaf': 3, 'min_samples_split': 3, 'n_estimators': 200, 'bootstrap': 'False', 'n_features_to_select': 40
Random Forest + ADASYN with MI	'criterion': 'entropy', 'max_depth': 20, 'max_features': 'sqrt', 'min_samples_leaf': 3, 'min_samples_split': 3, 'n_estimators': 400, 'bootstrap': 'False', 'selector__k': 40
Random Forest + ADASYN with RFE	'criterion': 'entropy', 'max_depth': 25, 'max_features': 'sqrt', 'min_samples_leaf': 3, 'min_samples_split': 3, 'n_estimators': 200, 'bootstrap': 'False', 'n_features_to_select': 40

Random Forest + SMOTE-ENN with RFE	'criterion': 'entropy', 'max_depth': 20, 'max_features': 'sqrt', 'min_samples_leaf': 3, 'min_samples_split': 3, 'n_estimators': 500, 'bootstrap': 'False' 'n_features_to_select': 40