Comparative Analysis of Undersampling, Oversampling, and SMOTE Techniques for Addressing Class Imbalance in Phishing Website Detection

Kamal Omari[1], Chaimae Taoussi[2], Ayoub Oukhatar[3]
Computer Science Department-Polydisciplinary Faculty of Ouarzazate, University Ibn Zohr, Ouarzazate, Morocco[1]
Laboratory of Process Engineering Computer Science and Mathematics,
University Sultan Moulay Slimane, Beni Mellal, Morocco[2]
Computer Science Department-Higher School of Technology Ouarzazate, University Ibn Zohr, Ouarzazate, Morocco[3]

***Abstract*—Since this is one of the most challenging tasks in cyber security, many of them are affected by class imbalance when it comes to the performance of machine learning. This paper evaluates various strategies using a number of resampling-based approaches: ROS, RUS, and SMOTE-based methods in conjunction with XGBoost classifier techniques to solve such an imbalanced dataset. Key performance measures include precision, F1 score, recall, precision, ROC-AUC, and geometric mean score. Among the methods, the highest was found with regard to the SMOTE-NC-XGB with precision equal to 98.0% and a recall of 98.5%, thus ensuring an effective trade-off between sensitivity and specificity. Although the stand-alone XGB model performs really well, adding resampling techniques makes its efficiency much higher, especially in cases of evident imbalance between classes. These results also revealed that resampling techniques are really helpful to enhance detection performance; hence, the SMOTE-NC-XGB is found out as the best among all of these. It will be of great contribution for future works in order to enhance the development of phishing detection systems and investigate other new hybrid resampling methods.**

Keywords*—Phishing website detection; class imbalance; XGBoost; SMOTE-NC

I. Introduction

While the cyberworld has expanded infinitely, phishing assaults have evolved into increasingly sophisticated attacks that target individuals and entities by assuming the identity of legitimate websites for the purpose of retrieving sensitive information [1]. Detection of these malicious sites is an important challenge in cybersecurity which is further complicated by the gross class imbalance contained in phishing collections. In these datasets, legitimate websites far exceed phishing websites, and machine learning models cannot detect phishing attacks effectively—a key process of mitigating cyber attacks [2].

Accuracy and reliability of machine learning models heavily depend on the quality and consistency of training data. Proper data preparation—cleaning, processing, validation, and transformation of raw data—is essential to building good models [3]. Class imbalance handling is one of the most important tasks in this work. Class imbalance, as a condition of underrepresented minority class, is common in applications such as fraud detection, health, and phishing website detection [4, 5]. In phishing detection, underrepresentation of phishing sites in datasets generally results in model bias towards the majority class, increasing the risk of false negatives, where phishing sites are classified as normal [6].

Machine learning models are negatively affected by imbalanced datasets with skewed class distributions, as this causes the models to be biased towards the majority class. This diminishes their generalization capability and hinders their applicability in real-world situations [7]. Traditional classification techniques are biased towards the majority class, which worsens these problems and highlights the necessity of specialized methods to counteract class imbalance.

In order to address this problem, researchers have put forward various resampling methods, i.e., undersampling, oversampling, and the Synthetic Minority Over-sampling Technique (SMOTE) [8]. All these methods assist in rebalancing data by reducing the majority class size or enhancing the minority class size and, therefore, the model becomes more capable of distinguishing between authentic sites and phishing websites. Here, we will attempt to combine these resampling techniques with the XGBoost classifier, a widely recognized algorithm for its high performance and scalability [9]. Despite its demonstrated effectiveness, inadequate detailed comparative studies in the literature have investigated the impacts of using different resampling techniques when coupled with XGBoost for detecting phishing websites.

This research aims to bridge this gap by providing a comparative analysis of undersampling, oversampling, and SMOTE methods under the XGBoost algorithm for phishing website detection optimization. From real datasets, we compare how these resampling methods affect the performance of the XGBoost classifier in handling class imbalance. The results not only show the relative merits and drawbacks of every resampling method but also give specific guidelines for researchers and practitioners interested in constructing more robust phishing detection systems.

The rest of this paper is organized as follows: Section II describes the importance of class imbalance in phishing website detection and how resampling methods can be utilized to alleviate it. Section III compares the performance of the XGBoost classifier when combined with these resampling methods, comparing their impacts on key performance metrics. Section IV outlines the experimental design, data set, preprocessing, and application of the XGBoost algorithm. Section V presents the results and discusses the impact of resampling methods on model performance. Finally, Section VI concludes with a summary of the major findings and suggesting potential areas for further study.

II. Literature Review

Preparation of data is an important phase of the data mining process, typically consuming 70–80% of the total time invested in data science activities [10]. The "garbage in, garbage out" (GIGO) principle emphasizes the extreme importance of high-quality input data in obtaining accurate and useful output. Good data preparation is more than data cleansing to include missing value handling, duplicate elimination, and outlier handling, all of which are required to derive actionable insights from raw and usually unstructured data [11].

In detecting phishing sites, datasets show complex patterns when it comes to characteristics such as URL patterns, hosting characteristics, and user activities. These patterns can be effective inputs for predictive models to identify phishing attacks [12]. However, these patterns have to be unearthed via repeated testing and knowledge of domains [13]. Therefore, data preparation is an important task in cybersecurity that transforms raw data into formats where it can be acted upon, being sensitive to machine learning algorithms.

Among the biggest problems for detecting phishing is the extreme class imbalance, where genuine websites heavily outnumber phishing websites. The issue tends to bias model performance and has a propensity to result in bias towards the majority class and higher false negative likelihoods. This research seeks to mitigate this through experimentally testing resampling methods including undersampling, oversampling, and an in-house approach called SMOTE-NC (Synthetic Minority Over-sampling Technique for Nominal and Continuous variables). These methods are intended to introduce balance to the dataset so that the model can better identify phishing sites accurately and eliminate the adverse effect of class imbalance.

A. Undersampling Techniques to Handle Class Imbalance Undersampling reduces the majority class in such a way that the dataset is balanced and machine learning algorithms can learn from minority class samples without bias. For instance, downsampling 12,000 majority class samples to balance 2,000 minority class samples creates a balanced 1:1 ratio. Although this technique makes the dataset easy to train, it must selectively choose majority class samples to maintain the integrity and predictive power of the dataset (Fig. 1).

A. Undersampling Techniques for Addressing Class Imbalance

Undersampling decreases the majority class to a level where the dataset is balanced, allowing the machine learning models to learn from the instances in the minority class without bias. For instance, decreasing 12,000 majority class samples to equal 2,000 minority class samples creates a 1:1 balanced dataset. Although this technique minimizes the complexity of the dataset during training, it needs to sample the majority class samples aggressively so that it does not compromise the integrity and forecastability of the dataset (Fig. 1).

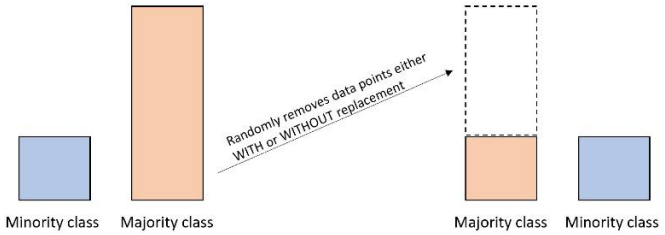

Fig. 1. Random undersampling process [14].

1) Random Undersampling (RUS): Random Undersampling (RUS) is the most basic and widely used undersampling technique. It does this by randomly selecting a subset of samples from the majority class to build a balanced dataset. This reduces the majority class bias, thereby improving the sensitivity of the model to phishing websites [15].

RUS has been used in a wide range of applications that encompass anomaly detection, fraud detection, and cyber security. When used in the detection of phishing websites, it reduces false negatives and enhances model performance by stabilizing class representation. Its simplicity comes at a price, though: the potential loss of informative information from the discarded majority class samples, which can limit the model's generalization capability in real-world scenarios.

B. Oversampling Techniques for Addressing Class Imbalance

Oversampling compensates for class imbalance by increasing the size of the minority class in the data set. The current minority class samples are replicated or new synthetic samples are created. Since the data set is balanced, oversampling eliminates the majority class bias and enhances the performance of the model in identifying phishing websites (Fig. 2).

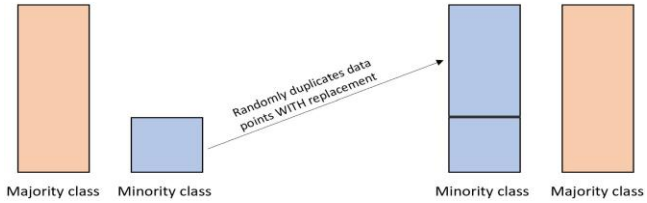

Fig. 2. Random oversampling process [14].

1) Random Oversampling (ROS): Random Oversampling (ROS) evens out the dataset by duplicating instances from the minority class to be identical to the majority class instances. As effective as this process treats class imbalance, it would also lead to overfitting based on the duplication of the same instances, hence reduced robustness and ability to generalize to new data [16].

In order to counteract these issues, variants like distributional random oversampling, where the synthetic samples are created from the statistical distribution of features, and stratified random oversampling, where generation of varied samples is guaranteed, have been suggested. Though having its own drawbacks, ROS has been found to be highly beneficial for phishing website detection by enhancing the representation of the minority class and model performance during training.

C. SMOTE-NC for Handling Class Imbalance

SMOTE-NC is an extension of the original SMOTE algorithm for the treatment of datasets with mixed data types, i.e., categorical and numerical variables. SMOTE-NC creates the synthetic samples in such a manner that it preserves the inherent features and interrelation among the data and effectively addresses class imbalance without compromising data integrity (Fig. 3).

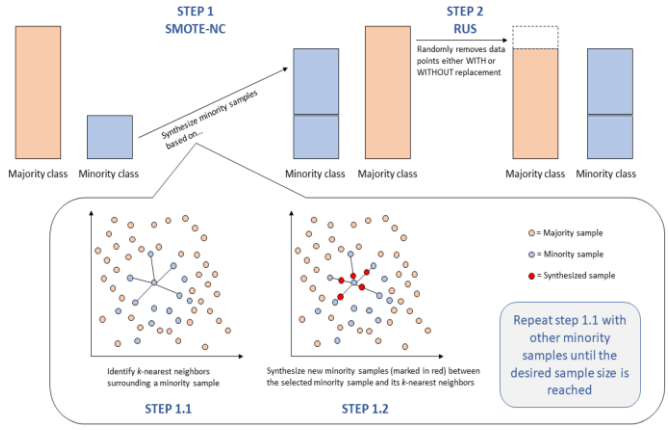

Fig. 3. SMOTE-NC process [14].

SMOTE-NC algorithm generates samples by interpolating between the minority class instances and their k-nearest neighbors in the numerical features. In categorical features, it finds the most frequent category among the neighbors so that the synthetic samples preserve the inherent categorical data distributions. This approach preserves the integrity of the mixed data types while improving the model performance, particularly in the detection of phishing websites [17].

However, SMOTE-NC is not limited [18]:

- Problems with high-dimensional data: Distance measures are meaningless in high-dimensional space.
- Introduction of noise: Synthetic samples may deviate from the true minority class distribution.
- Fixed k-neighbors: The k-neighbors parameter is applied uniformly, without considering the local variation of the data.

Despite the limitations outlined, SMOTE-NC has proven to be a valuable tool when used in phishing website detection, particularly when operating in datasets that involve categorical features such as domain-pattern-based or URL feature-based classifications. Through the preservation of integrity of the categorical variables and handling the class imbalance problem, SMOTE-NC operates to enhance the model towards effective phishing website detection.

Overall, undersampling, oversampling, and hybrid approaches each possess their own strength in class imbalance handling for phishing website detection. Selection of an effective resampling approach must be guided by the characteristics of the dataset and the requirements of the task at hand. Effective data preprocessing, particularly when working with imbalanced datasets, is critical to developing stable and actionable machine learning models with the ability to counteract the dynamic nature of phishing attacks.

III. Methods

This chapter introduces the research framework that seeks to enhance phishing website detection using a combination of state-of-the-art resampling methods and the XGBoost classifier. This chapter begins with introducing a step-by-step description of the dataset, as well as a heatmap visualization of feature correlation, which gives valuable information regarding variable correlation. The article then goes on to outline various class imbalance learning techniques, i.e., Random Oversampling (ROS), Random Undersampling (RUS), and SMOTE-NC, and outlines their settings and respective contributions to the resampling process. The XGBoost classifier is then outlined in detail, including its setup, tuning, and why it is suitable for the phishing detection task. This integrated paradigm facilitates systematic study of resampling's methods and their combined impact when used along with XGBoost, thereby preventing the consequences of class imbalance in phishing site detection.

A. Dataset Description

The dataset used in this work is taken from the UCI Machine Learning Repository [19]. It contains 11,055 records with 30 website attributes along with a class label indicating websites as phishing (1) and legitimate (-1). Table I summarizes the makeup of the dataset.

TABLE I. Dataset Description

Total number of instances	Total of Features	Class Variable Phishing website	Class Variable Legitimate website
11055	30	6157	4898

1) Visualizing the dataset: Heatmap of feature correlations: To understand relationships between features, a heatmap visualization was generated, identifying patterns and correlations crucial for feature engineering and selection [20] (Fig. 4).

In the heat map, each cell is colored according to a correlation value that goes from dark to light, indicating respectively a strong positive correlation and a strong negative one. The closer the correlation value is to +1, the stronger the positive relationship. The closer the value is to -1, the stronger the negative relationship. A correlation close to 0 means there is little or no linear relationship between the compared features. This visualization has given a better idea of interfeature dependencies; hence, crucial toward understanding the feature relevancy in respect of phishing website detection.

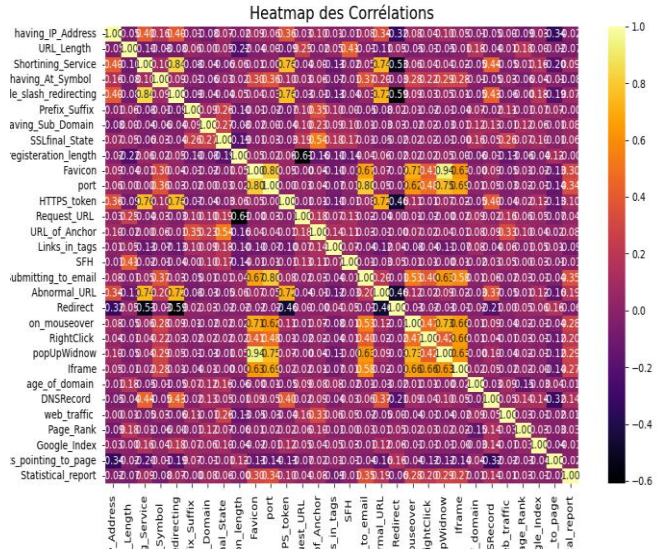

Fig. 4. The heatmap of dataset features.

B. Imbalance Learning Techniques

In this paper, ROS, RUS, and a hybrid approach with SMOTE-NC are used for performance comparisons of different resampling methods. SMOTE-NC with five nearest neighbors uses a resampling ratio of 0.8 to generate synthetic samples, while ROS and RUS apply random sampling processes to adjust the class distribution. These resampling techniques were implemented using the imbalanced-learn Python package [21], ensuring the reproducibility and consistency of the results.

1) Random Oversampling-XGBoost (ROS-XGB): ROS is a technique of oversampling the minority class samples to make the dataset balanced [22]. The ROS-XGB method, which combines the XGBoost classifier with random oversampling, enhances the model performance in phishing website detection by overcoming the problem of class imbalance. The hyperparameters of ROS-XGB are tuned for better performance. The details are shown in Table II. The parameters include the learning rate, maximum depth, number of estimators, and subsampling rate, which are tuned carefully for better accuracy and robustness of the model.

TABLE II. ROS-XGB PARAMETERS

Parameter	Value	Description
learning_rate	0.5	Resampling the minority class to balance the dataset.
max_depth	7	Controls tree depth for model complexity.
n_estimators	100	Number of boosting rounds.
subsample	1.0	Fraction of samples used for fitting each tree.

ROS-XGB improves recall and overall performance by mitigating class imbalance, making it effective at detecting phishing websites while maintaining high precision.

2) Random Undersampling-XGBoost (RUS-XGB): Random Undersampling (RUS) reduces the size of the majority class to that of the minority class, ensuring a balanced dataset [23]. When combined with the XGBoost classifier, the RUS-XGB approach enhances phishing website detection by giving equal importance to both classes. The hyperparameters for RUS-XGB are shown in Table III, which presents the specific configurations used to optimize model performance.

TABLE III. RUS-XGB PARAMETERS

Parameter	Value	Description
learning_rate	0.5	Resampling the minority class to balance the dataset.
max_depth	8	Tree depth to control model complexity.
n_estimators	300	Number of boosting rounds.
subsample	0.8	Fraction of samples used for fitting each tree.

RUS-XGB helps address class imbalance by reducing the influence of the majority class, thus improving recall and overall detection accuracy.

3) SMOTE-NC-XGBoost (SMOTE-NC-XGB): SMOTE-NC [24] is the Synthetic Minority Over-sampling Technique for Nominal and Continuous features that will generate synthetic samples for the minority class without affecting the structure of categorical features. XGB integrated with SMOTE-NC, denoted as SMOTE-NC-XGB, addresses class imbalance by generating more balanced data. The detailed hyperparameter tuning for SMOTE-NC-XGB on the number of nearest neighbors and resampling ratio is listed in Table IV.

TABLE IV. SMOTE-NC-XGB PARAMETERS

Parameter	Value	Description
learning_rate	0.1	Resampling the minority class to balance the dataset.
max_depth	7	Tree depth to control model complexity.
n_estimators	300	Number of boosting rounds.
subsample	0.8	Fraction of samples used for fitting each tree.

SMOTE-NC-XGB generates synthetic samples while maintaining the integrity of both categorical and continuous features, boosting recall, precision, and overall model performance in phishing website detection.

C. The XGBoost Classifier

XGBoost was selected because of its superior performance in classification tasks, especially for complex and imbalanced datasets, which is often the case in phishing website detection. Being a gradient boosting algorithm, XGBoost creates an ensemble of decision trees, improving predictive accuracy by iteratively adding models in a sequence [25]. It is particularly suited for phishing detection, where the challenge lies in identifying a minority class of phishing websites within a larger majority class of legitimate websites. XGBoost has, by default, mechanisms that handle class imbalances by adjusting the class weights, thus making the model focus more on the minority class. Moreover, in XGBoost, there is a possibility of extensive hyperparameter tuning, including but not limited to learning rate and tree depth,

which can be optimized for a particular dataset [26]. Its parallel processing ability combined with regularization techniques enhances the efficiency of the algorithm and helps avoid overfitting, therefore enhancing the robustness of the model [27]. These features together make XGBoost a very effective tool in combating class imbalance for the improvement of the performance of a phishing website detector.

IV. Proposed Framework

The presented work investigates some of the state-of-the-art class imbalance mitigation strategies in the phishing website detection problem by performing performance evaluation with respect to three different resampling techniques: Random Oversampling, Random Undersampling, and SMOTE-NC, when used together with the XGBoost classifier. Its main purpose is to look at the results that can be expected regarding major model performance metrics with such resampling techniques.

The proposed framework describes, in detail, the handling of class imbalance and the measurement of performance of the XGBoost classifier. Fig. 5 depicts the step-by-step data acquisition from the UCI database, analysis of distribution of classes for checking possible imbalances. Then, resampling techniques such as SMOTE-NC, ROS, and RUS are further used to balance the dataset.

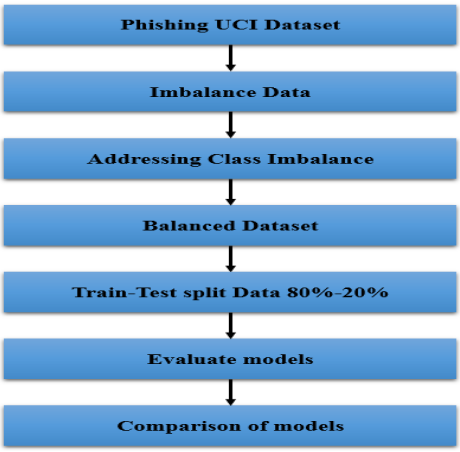

Fig. 5. Proposed framework.

After balancing, the dataset will then be divided into training and testing subsets. Here, the training subset is used in optimizing and training the XGBoost model while the test subset will be used in determining the classification accuracy of the model. Thereafter, after the training is done, the model classifies a new instance as either a phishing or a legitimate website. The key performance metrics used to measure the performance of the approach include accuracy, F1 score, recall, precision, ROC-AUC, and geometric mean score. These metrics also serve to underline the varying impact different resampling techniques have on overall model performance.

A. Class Distribution Analysis Before and After Resampling

Table V presents the class distributions of the original dataset and the balanced datasets after applying the resampling techniques.

TABLE V. The Class Distribution

Class	Legitimate	Phishing
Original Samples	4898	6157
Distribution (%)	44.30%	55.70%
ROS Samples	4926	4926
Distribution (%)	50.00%	50.00%
RUS Samples	3918	3918
Distribution (%)	50.00%	50.00%
SMOTE-NC Samples	4926	4926
Distribution (%)	50.00%	50.00%

This table is representing the balance of the imbalanced dataset in which phishing samples had a minority participation compared to the legitimate dataset. It is balanced using ROS, RUS, and SMOTE-NC. These all techniques are implemented to address the class imbalance such that the performance of the classifier XGBoost would improve in phishing web site detection so as to make a fair deal for both the classes: minority as well as the majority.

B. Evaluation Metrics

Evaluation metrics play an important role in machine learning for the measurement of model performance. It provides a quantitative measure of the performance of the model using various metrics: accuracy, precision, recall, F1 score, ROC-AUC, and Geometric Mean score. These metrics help the researchers in choosing the best approach for a particular task and hence ensure that the chosen model will effectively address the unique challenges of the problem.

1) Accuracy: Accuracy is the measure of the percentage of correctly predicted instances. Although useful, it can be misleading for imbalanced data sets as high accuracy might overemphasize the performance on the majority class.

$$Accuracy = \frac{Number\ of\ correctly\ predicted\ instances}{Total\ number\ of\ instances}$$

2) F1 Score: F1-score is the harmonic mean of precision and recall, effectively balancing the trade-off between false positives and false negatives.

$$F1Score = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

3) Recall: Recall measures the proportion of actual positives correctly identified, making it crucial when minimizing false negatives is a priority.

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives}$$

4) Precision: Precision quantifies the proportion of true positives among all predicted positives, and is particularly important when minimizing false positives is critical.

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives}$$

5) ROC-AUC: ROC-AUC measures the model's ability to distinguish between classes. A higher value indicates better performance in differentiating between the positive and negative classes.

$$ROC - AUC = \int_0^1 TPR(FPR)d(FPR)$$

6) Geometric Mean Score (G-Mean): The geometric mean score equilibrates performance on both classes and is hence useful in the case of imbalanced datasets. It is calculated as the square root of the product of recall and specificity, offering a more balanced evaluation when class distribution is skewed.

$$\text{G} - \text{Mean} = \sqrt{Recall \times Specificity}$$

V. Findings and Analysis

This paper investigates the performance of three resampling techniques, namely ROS, RUS, and SMOTE-NC, integrated with the XGBoost classifier for phishing website detection. The key objective is to investigate how these techniques affect class distribution, improve predictive accuracy, and optimize key evaluation metrics.

Table VI provides the performance evaluation of the discussed models with respect to six performance metrics: accuracy, F1 score, recall, precision, ROC-AUC, and the Geometric Mean score.

TABLE VI. Evaluation Results in (%)

Classifier	Accuracy	F1 score	Recall	Precision	ROC-AUC	The Geometric Mean score
XGB	0.977	0.979	0.983	0.976	0.998	0.976
ROS-XGB	0.976	0.978	0.979	0.977	0.998	0.975
RUS-XGB	0.974	0.976	0.974	0.979	0.998	0.974
SMOTE-NC-XGB	0.980	0.982	0.985	0.979	0.998	0.979

SMOTE-NC-XGB gives the best general performance among the tested models, with the highest accuracy of 0.980, F1 score of 0.982, recall of 0.985, and Geometric Mean score of 0.979. It means that the SMOTE-NC-XGB model provides very well-balanced sensitivity and specificity while handling class imbalance. The XGB baseline also performs quite well, with high recall of 0.983 and a high ROC-AUC of 0.998. However, resampling techniques enormously increased the generalization of the model to imbalanced datasets.

- ROS-XGB slightly improved precision, at 0.977, from the baseline XGB, though in most metrics it performed the same as the SMOTE-NC-XGB model.
- RUS-XGB has better precision, 0.979, but it decreases the recall to 0.974 due to reduced false positives.
- SMOTE-NC-XGB outperforms all the other methods since it strikes the best balance among all metrics; thus, it is the most suitable approach for phishing website detection.

These findings really pinpoint the very crucial role that resampling techniques play in improving model performance, especially when dealing with imbalanced datasets. The results clearly indicate that SMOTE-NC-XGB emerges as the optimal method for phishing detection in this study.

VI. Conclusion

This paper discusses the performance comparison of undersampling, oversampling, and SMOTE-based techniques combined with the XGBoost classifier for class imbalance in phishing website detection. The experimental results illustrate that resampling methods greatly enhance the capability of a model in dealing with imbalanced datasets and improving the key evaluation metrics, including accuracy, F1 score, recall, precision, ROC-AUC, and Geometric Mean score.

Among the models tested, SMOTE-NC-XGB has always been at the top for all metrics with an accuracy of 98.0% and a recall of 98.5%, values which are rather high, therefore, it is able to balance sensitivity and specificity, which is the very key problem in phishing website detection; either false positives or false negatives can cause huge losses.

While the XGB baseline is also good to go, at 99.8% ROC-AUC, the introduction of resampling techniques into its training gives the model extra strength, particularly under strong class imbalance. Both Random Oversampling (ROS) and Random Undersampling (RUS) turn in competitive performances; the former slightly improves precision, while the latter is able to strongly reduce false positives.

In a nutshell, the results of this work highlight the key contribution of resampling techniques in enhancing the performance of models on imbalanced datasets. Among the considered methods, SMOTE-NC-XGB is the most balanced and reliable, hence presenting a promising solution for the enhancement of phishing website detection systems. Future work may consider hybrid resampling methods or more advanced methods to further optimize detection performance.

References

[1] M. S. Kheruddin, M. A. E. M. Zuber, and M. M. M. Radzai, "Phishing Attacks: Unraveling Tactics, Threats, and Defenses in the Cybersecurity Landscape," TechRxiv, Jan. 15, 2024. DOI: 10.22541/au.170534654.48067877/v1.

[2] I. Araf, A. Idri, and I. Chairi, "Cost-sensitive learning for imbalanced medical data: A review," Artificial Intelligence Review, vol. 57, no. 4, p. 80, 2024. DOI: 10.1007/s10462-023-10652-8.

[3] A. Aldoseri, K. N. Al-Khalifa, and A. M. Hamouda, "Re-thinking data strategy and integration for artificial intelligence: Concepts, opportunities, and challenges," Applied Sciences, vol. 13, no. 12, p. 7082, 2023. DOI: 10.3390/app13127082.

[4] B. Krawczyk, "Learning from imbalanced data: open challenges and future directions," Progress in Artificial Intelligence, vol. 5, no. 4, pp. 221–232, 2016. DOI: 10.1007/s13748-016-0094-0.

[5] M. Lokanan, "Exploring Resampling Techniques in Credit Card Default Prediction," Research Square, Preprint, 15 Mar. 2024. DOI: 10.21203/rs.3.rs-4087259/v1.

[6] Q. Wei and R. L. Dunbrack Jr., "The Role of Balanced Training and Testing Data Sets for Binary Classifiers in Bioinformatics," PLoS ONE, vol. 8, no. 7, p. e67863, Jul. 2013. DOI: 10.1371/journal.pone.0067863.

[7] M. Galar, A. Fernández, E. Barrenechea, and F. Herrera, "EUSBoost: Enhancing ensembles for highly imbalanced data-sets by evolutionary undersampling," Pattern Recognition, vol. 46, no. 12, pp. 3460-3471, 2013. DOI: 10.1016/j.patcog.2013.05.006.

[8] M. Shelke, P. R. Deshmukh, and V. K. Shandilya, "A review on imbalanced data handling using undersampling and oversampling technique," International Journal of Recent Trends in Engineering, 2017. DOI: 10.3883/IJRTER.2017.3168.0UWXM.

[9] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, Aug. 2016, pp. 785-794. Association for Computing Machinery, doi: 10.1145/2939672.2939785.

[10] L. Yu, S. Wang, and K. K. Lai, "An integrated data preparation scheme for neural network data analysis," IEEE Transactions on Knowledge and Data Engineering, vol. 18, no. 2, pp. 217–230, 2006, doi: 10.1109/TKDE.2006.22.

[11] S. Angra and S. Ahuja, "Machine learning and its applications: A review," in 2017 International Conference on Big Data Analytics and Computational Intelligence (ICBDAC), 2017, pp. 57–60, doi: 10.1109/ICBDACI.2017.8070809.

[12] M. J. Nigrini, "The patterns of the numbers used in occupational fraud schemes," *Managerial Auditing Journal*, vol. 34, no. 5, pp. 606–626, 2019, doi: 10.1108/MAJ-11-2017-1717.

[13] V. Mirchevska, M. Luštrek, and M. Gams, "Combining domain knowledge and machine learning for robust fall detection," *Expert Systems*, vol. 31, no. 2, pp. 163–175, 2014, doi: 10.1111/exsy.12019.

[14] T. Wongvorachan, S. He, and O. Bulut, "A comparison of undersampling, oversampling, and SMOTE methods for dealing with imbalanced classification in educational data mining," *Information*, vol. 14, no. 1, p. 54, 2023, doi: 10.3390/info14010054.

[15] G. Menardi and N. Torelli, "Training and assessing classification rules with imbalanced data," *Data Mining and Knowledge Discovery*, vol. 28, no. 1, pp. 92–122, 2014.

[16] S. J. Dattagupta, "A performance comparison of oversampling methods for data generation in imbalanced learning tasks," Ph.D. thesis, Universidade Nova de Lisboa, Lisbon, Portugal, 2018.

[17] M. Mukherjee and M. Khushi, "SMOTE-ENC: A Novel SMOTE-Based Method to Generate Synthetic Data for Nominal and Continuous Features," *Applied System Innovation*, vol. 4, no. 1, p. 18, 2021, doi: 10.3390/asi4010018.

[18] W. W. Islahulhaq and I. D. Ratih, "Classification of Non-Performing Financing Using Logistic Regression and Synthetic Minority Oversampling Technique-Nominal Continuous (SMOTE-NC)," *International Journal of Advanced Soft Computing and Its Applications*, vol. 13, pp. 116–128, 2021.

[19] UCI Machine Learning Repository, "Phishing Websites Data Set," Available online: https://archive.ics.uci.edu/ml/datasets/phishing+websites.

[20] K. Omari, "Comparative study of machine learning algorithms for phishing website detection," *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 9, 2023.

[21] Lema, G., "Imbalanced-learn: A Python toolbox to tackle the curse of imbalanced datasets in machine learning," *Journal of Machine Learning Research*, vol. 18, pp. 1–5, 2017.

[22] More, A., "Survey of resampling techniques for improving classification performance in unbalanced datasets," *arXiv preprint*, 2016. doi: 10.48550/ARXIV.1608.06048.

[23] R. Zuech, J. Hancock, and T. M. Khoshgoftaar, "Detecting Web Attacks Using Random Undersampling and Ensemble Learners," *Journal of Big Data*, vol. 8, no. 1, p. 75, 2021. doi: 10.1186/s40537-021-00460-8.

[24] Chawla, N.V., Bowyer, K.W., Hall, L.O., and Kegelmeyer, W.P., "SMOTE: Synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002.

[25] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Francisco, CA, USA, Aug. 2016, pp. 785–794. Association for Computing Machinery, doi: 10.1145/2939672.2939785.

[26] T. Kavzoglu and A. Teke, "Advanced hyperparameter optimization for improved spatial prediction of shallow landslides using extreme gradient boosting (XGBoost)," *Bulletin of Engineering Geology and the Environment*, vol. 81, Article 201, 2022. doi: 10.1007/s10064-022-02708-w.

[27] S. S. Dhaliwal, A. A. Nahid, and R. Abbas, "Effective Intrusion Detection System Using XGBoost," *Information*, vol. 9, no. 7, p. 149, 2018. doi: 10.3390/info9070149.