IAES International Journal of Artificial Intelligence (IJ-AI)
Vol. 15, No. 3, June 2026, pp. 2935~2945
ISSN: 2252-8938, DOI: 10.11591/ijai.v15.i3.pp2935-2945

Enhancing phishing website detection: a comparative study of SMOTETomek-XGB and SMOTEENN-XGB

Kamal Omari[1], Ayoub Oukhatar[2]
[1]Laboratory of Computer Systems and Vision (LabSIV), Department of Computer Science, Polydisciplinary Faculty of Ouarzazate, University Ibn Zohr, Ouarzazate, Morocco
[2]Department of Computer Science, Higher School of Technology Ouarzazate, University Ibn Zohr, Ouarzazate, Morocco

Article Info

Article history:

Received Sep 8, 2024
Revised Mar 6, 2026
Accepted Apr 22, 2026

Keywords:

Class imbalance
Phishing website detection
SMOTEENN techniques
SMOTETomek techniques
XGBoost classifier

ABSTRACT

In the evolving landscape of cybersecurity, phishing websites continue to be a persistent threat, challenging detection methods due to the significant class imbalance between phishing and legitimate websites. This study evaluates the effectiveness of two advanced hybrid-resampling techniques SMOTETomek and SMOTEENN integrated with the extreme gradient boosting (XGBoost) classifier to enhance phishing website detection. SMOTETomek combines the synthetic minority over-sampling technique (SMOTE) with Tomek links, creating synthetic examples and eliminating overlapping instances to address dataset imbalance. SMOTEENN, on the other hand, merges SMOTE with edited nearest neighbors (ENN) to improve class balance through synthetic sample generation and noise reduction. The comparative analysis reveals that both methods significantly enhance classification performance, SMOTETomek-XGB consistently outperforms SMOTEENN-XGB across key evaluation metrics, including accuracy, F1-score, recall, and receiver operating characteristic - area under the curve (ROC-AUC), underscoring its superior effectiveness in distinguishing phishing sites from legitimate ones. This study offers practical insights into the application of advanced resampling methods for improving machine learning model performance in cybersecurity.

This is an open access article under the CC BY-SA license.

Corresponding Author:

Kamal Omari
Laboratory of Computer Systems and Vision (LabSIV), Department of Computer Science
Polydisciplinary Faculty of Ouarzazate, University Ibn Zohr
Ouarzazate, Morocco
Email: k.omari@uiz.ac.ma

1. INTRODUCTION

Phishing attacks pose a significant and growing threat in cybersecurity, with cybercriminals employing increasingly sophisticated tactics to deceive individuals into revealing sensitive information. These phishing websites mimic legitimate online platforms, putting both individuals and organizations at risk by tricking users into divulging personal details or credentials [1]. Detecting these fraudulent websites is crucial for cybersecurity experts, yet traditional detection methods often fall short due to the pronounced class imbalance in phishing datasets [2].

This imbalance arises because phishing websites are much less common than legitimate ones, leading to a skewed class distribution that hampers the effectiveness of classification algorithms [3]. Standard machine learning classifiers tend to be biased toward the majority class, often resulting in poor detection of phishing websites [2]. This issue highlights the urgent need for advanced techniques that can address this imbalance and improve detection accuracy.

To tackle these challenges, researchers have developed various resampling techniques aimed at adjusting class distribution and enhancing model performance. Among these, two notable hybrid-resampling methods are SMOTETomek and SMOTEENN, both of which have shown promise in balancing class distributions and improving classification results across different applications [4]. SMOTETomek combines the synthetic minority over-sampling technique (SMOTE) with Tomek links. SMOTE generates synthetic samples for the minority class to address imbalance, while Tomek links remove ambiguous instances near class boundaries, refining the separation between classes [5]. This dual approach not only strengthens the minority class representation but also refines the dataset, potentially enhancing classifier performance.

Conversely, SMOTEENN integrates SMOTE with edited nearest neighbors (ENN). While SMOTE generates synthetic instances of the minority class, ENN reduces noise by removing instances misclassified by their nearest neighbors [5]. This method seeks to balance oversampling with data cleaning, offering an effective strategy for dealing with imbalanced data.

Despite the potential of these techniques, their effectiveness in detecting phishing websites remains unclear. Comparing different under- and oversampling methods can provide practical insights for data scientists and cybersecurity experts, bridging the gap between theoretical research and real-world applications. This study addresses this gap by conducting a comprehensive comparative analysis of SMOTETomek-XGB and SMOTEENN-XGB for detecting phishing websites. Extreme gradient boosting (XGBoost) was selected, a leading gradient boosting algorithm, for its proven ability to handle imbalanced datasets and its exceptional performance across various machine-learning tasks [6]. By comparing the performance of SMOTETomek and SMOTEENN with XGBoost, this research aims to identify the most effective resampling technique for phishing detection and provide actionable insights for future applications in cybersecurity.

The rest of this paper is organized as follows. Section 2 reviews existing literature on phishing website detection and resampling techniques. Section 3 details the methodologies used, including explanations of SMOTETomek, SMOTEENN, the XGBoost classifier, and experimental design. Section 4 presents the proposed framework, including class imbalance handling and the evaluation metrics used to assess model performance. Section 5 presents the results of comparative analysis, highlighting performance metrics for both SMOTETomek-XGB and SMOTEENN-XGB. Finally, section 6 discusses the implications of these findings for phishing website detection and concludes with recommendations for future research.

2. LITERATURE REVIEW

2.1. Phishing website detection

Phishing attacks are a significant threat in cybersecurity, exploiting users' trust by tricking them into revealing sensitive information on fraudulent websites designed to mimic legitimate ones. Detecting these deceptive sites is challenging due to the advanced techniques employed by attackers and the inherent class imbalance in phishing datasets. Traditionally, phishing detection relied on rule-based methods, which identified known phishing tactics or specific features like domain name discrepancies and visual similarities. However, as attackers have refined their strategies, these static rule-based approaches have become less effective [7].

In response to these evolving threats, recent efforts have shifted towards machine learning-based methods for phishing detection. Decision tree (DT), logistic regression (LR), k-nearest neighbors (KNN), random forest (RF), support vector machine (SVM), naive Bayes (NB), and gradient boosting are all examples of algorithms. have been employed to classify websites as phishing or legitimate based on various features [8]. Despite the advancements in these methods, the issue of class imbalance remains a critical challenge. Phishing websites generally represent only a small fraction of the data, causing classifiers to perform poorly on these infrequent instances [9].

2.2. Resampling techniques for imbalanced data

To address the challenge of class imbalance, several resampling techniques have been developed. Among these, SMOTE [10] combined with Tomek links [11], known as SMOTETomek [12], and SMOTE paired with ENN [5], forming SMOTEENN [13], are particularly noteworthy. Chawla *et al.* [10] came up with SMOTE, which makes fake samples for the minority class by filling in the gaps between real examples. This approach increases the representation of the minority class and addresses class imbalance. However, while SMOTE effectively boosts minority class instances, it can also introduce noise and overlap between classes, which may affect classifier performance.

Tomek links, proposed by Tomek [11], helps clean the dataset by removing ambiguous instances near the decision boundary. When combined with SMOTE, the SMOTETomek technique [12] not only augments the minority class but also refines the decision boundary by eliminating these ambiguous instances,

resulting in clearer class separation. On the other hand, SMOTEENN [13] combines SMOTE with ENN [5], a technique that reduces noise by removing instances misclassified by their nearest neighbors [14]. This hybrid approach balances the dataset while also cleaning it, which can enhance the performance of classification models.

Both SMOTEENN and SMOTETomek have been widely used to fix class imbalances in many fields, showing that they work well to improve model performance. In medicine, SMOTEENN has been widely used for early detection tasks, like predicting when septic shock will start, diagnosing missed abortion, and predicting chronic conditions like Parkinson's disease and chronic heart failure. This has led to better diagnostic accuracy [15], [16]. In the financial sector, SMOTEENN has also been successfully used to predict fraud with better results [17], [18]. These studies together show that the technique can balance datasets and make predictive models more reliable.

In addition, SMOTETomek has been extensively implemented in several fields, were imbalanced data poses challenges to model accuracy. For instance, in medicine, SMOTETomek has been used to develop predictive models for illnesses like diabetes and high blood pressure, as well as in cancer studies to predict the development of prostate and cervical cancer cases based on imbalanced data sets [19], [20]. Similarly, in computer science, SMOTETomek has proved to be useful in recommender systems and software bug prediction, as well as in handling highly unbalanced data sets in personality recognition [12]. These examples illustrate the considerable advantages that can be obtained from using the SMOTETomek algorithm, which is why this technique may be considered useful for dealing with the problem of class imbalances in numerous cases. In conclusion, although machine learning algorithms can provide more sophisticated methods of phishing detection, the problem of class imbalances requires special attention and resampling.

3. METHOD

This section presents the research methodology for improving phishing website detection by integrating advanced resampling techniques with the XGBoost algorithm. Specifically, we describe the dataset used in our experiments and detail the preprocessing steps applied to address class imbalance through hybrid resampling methods such as SMOTETomek and SMOTEENN. Furthermore, we outline the experimental framework designed to compare the performance of SMOTETomek-XGB and SMOTEENN-XGB, including evaluation metrics and validation procedures to ensure a robust and reliable analysis.

3.1. Dataset description

This study used a publicly available benchmark phishing website dataset from the UCI Machine Learning Repository [21]. The dataset comprises 11,055 samples with 31 attributes, including 30 features and one class label. The class imbalance ratio is calculated using (1).

$$Imbalance\ Ratio = \frac{Size\ of\ Majority\ Class}{Size\ of\ Minority\ Class} \quad (1)$$

The dataset categorized into two classes: “phishing website” and “legitimate website.” Specifically, there are 6,157 instances of phishing websites and 4,898 instances of legitimate websites. Specifically, there are 6,157 instances of phishing websites and 4,898 instances of legitimate websites. The imbalance ratio, calculated as the ratio of the number of legitimate websites to phishing websites, is 1.2570.

The selected dataset introduced several innovative features, including the experimental introduction of novel rules for specific well-established parameters [21]. A total of 30 parameters were considered in this analysis, which are listed in Table 1. The descriptive statistics summarized above such as count, mean, standard deviation, minimum, and maximum offer crucial insights into the central tendency, dispersion, and overall distribution of each attribute within the dataset. This analysis provides an in-depth understanding of the dataset's characteristics, which is essential for developing robust models and drawing reliable conclusions.

3.2. Advanced data-balancing techniques for phishing website detection

This study explored two advanced data-balancing techniques to address class imbalance in phishing website detection. The techniques used are i) SMOTEENN hybrid sampling method [13] and ii) SMOTETomek hybrid sampling method [12]. These techniques are designed to enhance the performance of machine learning models by creating a more balanced dataset, which is crucial for improving classification accuracy.

3.2.1. SMOTEENN hybrid sampling

SMOTEENN hybrid sampling [13] is a technique that combines two methods SMOTE [10] and ENN [5] under-sampling to tackle class imbalance effectively. SMOTE synthesizes minority examples by

choosing one minority example and creating more examples on the line joining it to its k-nearest minority examples. Although this technique makes sure there is a balance in the dataset by adding examples of the minority class, SMOTE might consider that any point joining any two examples of the minority class is also an example of the minority class.

To address this, ENN steps in to clean the dataset. It evaluates each sample based on its nearest neighbors and removes samples that are out of place, meaning those whose class labels don't match the majority class labels of their neighbors. This process helps to eliminate noise, especially when new synthetic samples from SMOTE might overlap with the majority class space as shown in Figure 1. By first applying SMOTE to balance the dataset and then using ENN to clean it up, SMOTEENN ensures that the dataset is well-balanced and minimizes the risk of introducing noise.

Table 1. Phishing website data attributes

Category	Data attributes	Mean	Std	Count	Min	25%	50%	75%	Max
URL-based	URL_Length	-0.633198	0.766095	11055	-1.0	-1.0	-1.0	-1.0	1.0
URL-based	having_IP_Address	0.313795	0.949534	11055	-1.0	-1.0	1.0	1.0	1.0
URL-based	Shortening_Service	0.738761	0.673998	11055	-1.0	1.0	1.0	1.0	1.0
URL-based	having_At_Symbol	0.700588	0.713598	11055	-1.0	1.0	1.0	1.0	1.0
URL-based	double_slash_redirecting	0.741474	0.671011	11055	-1.0	1.0	1.0	1.0	1.0
URL-based	Prefix_Suffix	-0.734962	0.678139	11055	-1.0	-1.0	-1.0	-1.0	1.0
Domain-based	Domain_registration_length	-0.336771	0.941629	11055	-1.0	-1.0	-1.0	1.0	1.0
Domain-based	age_of_domain	0.061239	0.998168	11055	-1.0	-1.0	1.0	1.0	1.0
Domain-based	DNSRecord	0.287291	0.827733	11055	-1.0	-1.0	1.0	1.0	1.0
Domain-based	Page_Rank	-0.483673	0.875289	11055	-1.0	-1.0	-1.0	1.0	1.0
Domain-based	Google_Index	0.721574	0.692369	11055	-1.0	1.0	1.0	1.0	1.0
Content-based	SSLfinal_State	0.250927	0.911892	11055	-1.0	-1.0	1.0	1.0	1.0
Content-based	Favicon	0.628584	0.777777	11055	-1.0	1.0	1.0	1.0	1.0
Content-based	Request_URL	0.186793	0.982444	11055	-1.0	-1.0	1.0	1.0	1.0
Content-based	URL_of_Anchor	-0.076526	0.715138	11055	-1.0	-1.0	0.0	0.0	1.0
Content-based	Links_in_tags	-0.118137	0.763973	11055	-1.0	-1.0	0.0	0.0	1.0
Content-based	SFH	-0.595749	0.759143	11055	-1.0	-1.0	-1.0	-1.0	1.0
Content-based	Submitting_to_email	0.635640	0.772021	11055	-1.0	1.0	1.0	1.0	1.0
Content-based	Abnormal_URL	0.705292	0.708949	11055	-1.0	1.0	1.0	1.0	1.0
Behavioral	Redirect	0.115694	0.319872	11055	0.0	0.0	0.0	0.0	1.0
Behavioral	on_mouseover	0.762099	0.647490	11055	-1.0	1.0	1.0	1.0	1.0
Behavioral	RightClick	0.913885	0.405991	11055	-1.0	1.0	1.0	1.0	1.0
Behavioral	popUpWindow	0.613388	0.789818	11055	-1.0	1.0	1.0	1.0	1.0
Behavioral	Iframe	0.816915	0.576784	11055	-1.0	1.0	1.0	1.0	1.0
Traffic & popularity	web_traffic	0.287291	0.827733	11055	-1.0	0.0	1.0	1.0	1.0
Traffic & popularity	Links_pointing_to_page	0.344007	0.569944	11055	-1.0	0.0	0.0	1.0	1.0
Traffic & popularity	Statistical_report	0.719584	0.694437	11055	-1.0	1.0	1.0	1.0	1.0
Traffic & popularity	HTTPS_token	0.675079	0.737779	11055	-1.0	1.0	1.0	1.0	1.0
Traffic & popularity	port	0.728268	0.685324	11055	-1.0	1.0	1.0	1.0	1.0

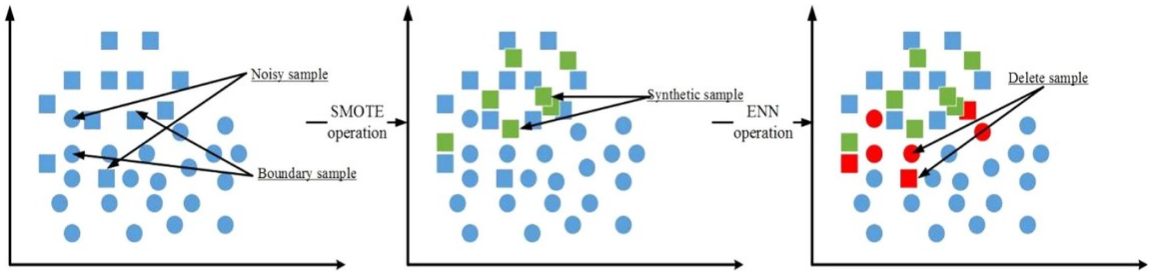

Figure 1. Schematic diagram of the SMOTEENN method

Table 2 outlines the parameters used in the SMOTEENN hybrid sampling method. The "sampling_strategy" is set to "auto", ensuring that the minority class is adequately resampled to achieve dataset balance. A "random_state" of 42 is used to maintain reproducibility by controlling the random number generation process. The "smote" parameter corresponds to the SMOTE object, configured with 5 nearest neighbors "k_neighbors =5" to guide the creation of synthetic samples for the minority class. The "enn" parameter relates to the ENN method, which is set to evaluate 1 nearest neighbor "n_neighbors =1" and apply the 'mode' selection strategy "kind_sel = "mode" to efficiently eliminate noisy instances. The "n_jobs" parameter is specified as "None", indicating the use of a single CPU core, though this can be adjusted based on the computational resources available (Table 2).

Table 2. SMOTEENN parameters

Parameter	Value	Description
sampling_strategy	'auto'	Resample the minority class to balance the dataset.
random_state	42	Seed used by the random number generator for reproducibility.
smote	SMOTE object (k_neighbors =5)	SMOTE object with specified parameters.
enn	ENN object (n_neighbors =1, kind_sel ='mode')	ENN object with specified parameters.
n_jobs	None	Number of CPU cores used during processing (None means using one core).

3.2.2. SMOTETomek hybrid sampling

SMOTETomek hybrid sampling is a technique that combines SMOTE and Tomek links to tackle class imbalance and boost classification performance. Here's how it works Tomek links is an under-sampling method that finds and removes pairs of samples that are close to each other but belong to different classes. This helps clean up the area between classes, reducing overlap and noise.

SMOTE, on the other hand, helps by creating artificial data points for the minority class. The technique involves choosing a minority class point and forming new points along the line segment joining that point to its neighboring points. This helps create a balanced data set and ensures proper distribution, which is extremely important in boosting machine learning model performance.

After applying SMOTE to increase the number of minority class samples, Tomek links further refines the dataset by removing the problematic pairs identified earlier. This step helps clear up any remaining noise and sharpens the decision boundary between classes as shown in Figure 2. By combining these techniques, SMOTETomek hybrid sampling effectively addresses class imbalance, reduces noise, and refines decision boundaries, significantly enhancing the performance of machine learning models, such as those used for detecting phishing websites.

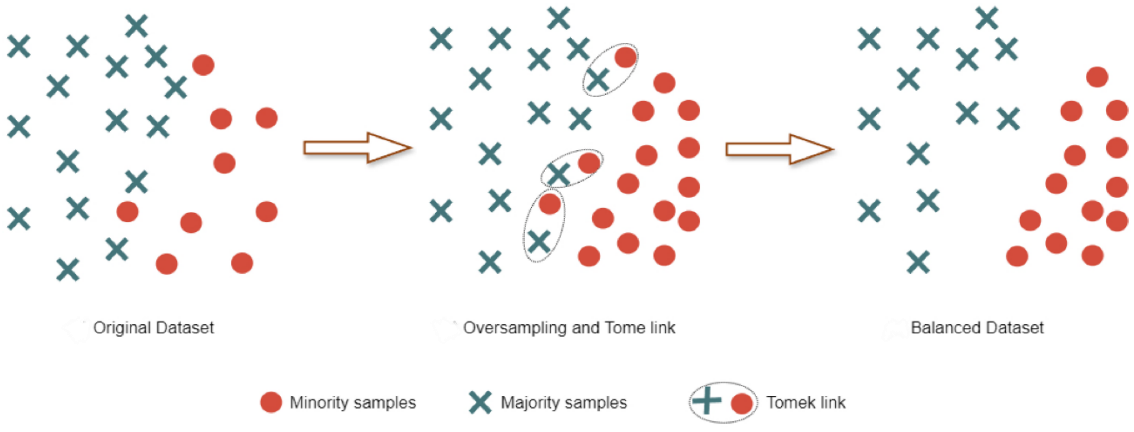

Figure 2. Schematic diagram of the SMOTE-Tomek method

The key difference between Tables 2 and 3 lies in the substitution of the ENN method with the Tomek links method. While the core parameters such as "sampling_strategy", "random_state", and "smote" remain consistent across both approaches, the introduction of the "tomek" parameter in Table 3 marks a key

adjustment. This parameter leverages Tomek links to identify and eliminate pairs of samples that are nearest neighbors but belong to different classes. This refinement process is crucial, as it effectively sharpens class boundaries, thereby reducing overlap and enhancing the overall precision of the classifier. By incorporating Tomek links, the SMOTETomek method offers a more refined approach to addressing class imbalance. The hyperparameters used for configuring the XGBoost classifier within the SMOTETomek and SMOTEENN frameworks are outlined. These settings were carefully chosen to balance model complexity, training efficiency, and predictive accuracy.

Table 3. SMOTETomek parameters

Parameter	Value	Description
sampling_strategy	'auto'	Resample the minority class to balance the dataset.
random_state	42	Seed used by the random number generator for reproducibility.
smote	SMOTE object (k_neighbors =5)	SMOTE object with specified parameters.
tomek	Tomek links object	Tomek Links method to identify and remove sample pairs that are nearest neighbors but belong to different classes.
n_jobs	None	Number of CPU cores used during processing (None means using one core).

3.2.3. XGBoost classifier

The XGBoost classifier was chosen for this study due to its exceptional performance and appropriateness for the task. XGBoost is particularly effective in managing complex and non-linear data through its gradient boosting framework, which builds an ensemble of trees to enhance model accuracy [22]. This characteristic is especially beneficial for phishing website detection, where the challenge of distinguishing between the minority class (phishing sites) and the majority class (legitimate sites) is exacerbated by class imbalance [23]. XGBoost's built-in class weight adjustment mechanisms are designed to address these imbalances, thereby improving the classifier's ability to identify phishing sites accurately. Furthermore, XGBoost provides extensive hyperparameter tuning options, such as adjustments to the learning rate and tree depth, which facilitate optimization for specific datasets [24]. Its efficiency is also enhanced by parallelization and regularization techniques, which contribute to faster processing and greater model robustness [25]. These features, together with robust community support and comprehensive documentation, make XGBoost a highly suitable choice for enhancing phishing website detection.

Careful tuning of these hyperparameters is essential for maximizing the classifier's performance in both frameworks, ensuring that the dataset remains well-balanced and of high quality as shown in Table 4. By carefully tuning these hyperparameters, we make the most of the SMOTETomek method for data resampling and the powerful capabilities of the XGBoost algorithm to accurately detect phishing domains, even when dealing with significant class imbalances in the dataset. In summary, the selected hyperparameters for both SMOTETomek-XGB and SMOTEENN-XGB are designed to balance model complexity, training time, and performance. The goal is to develop robust and effective classifiers capable of addressing the challenges posed by imbalanced datasets.

Table 4. XGBoost classifier parameters

Parameter	Value	Description
n_estimators	100	The number of boosting rounds. This parameter controls the number of trees in the ensemble.
max_depth	7	The maximum depth of each tree. Deeper trees can model more complex patterns but may lead to overfitting.
learning_rate	0.6	The step size shrinkage used in each boosting step to prevent overfitting. A higher learning rate speeds up learning but may require careful tuning.
random_state	42	The seed used by the random number generator for reproducibility. Ensures that the results can be replicated.
objective	'binary'	Loss function for binary classification tasks.
booster	'gbtree'	Specifies the learner type as 'gbtree', indicating that the model will be an ensemble of DT.

4. PROPOSED FRAMEWORK

This study introduces two advanced approaches designed to address class imbalance in phishing website detection datasets: SMOTEENN-XGB and SMOTETomek-XGB. Both methods integrate sophisticated resampling techniques with a powerful classification algorithm to enhance detection performance as shown in Figure 3. The figure illustrates the overall workflow for addressing class imbalance in a dataset

and evaluating the XGBoost classifier's performance. It begins with sourcing the dataset from the UCI database, followed by the identification of imbalanced class distributions. To rectify this, hybrid sampling techniques such as SMOTEENN and SMOTETomek are applied, resulting in a class-balanced dataset.

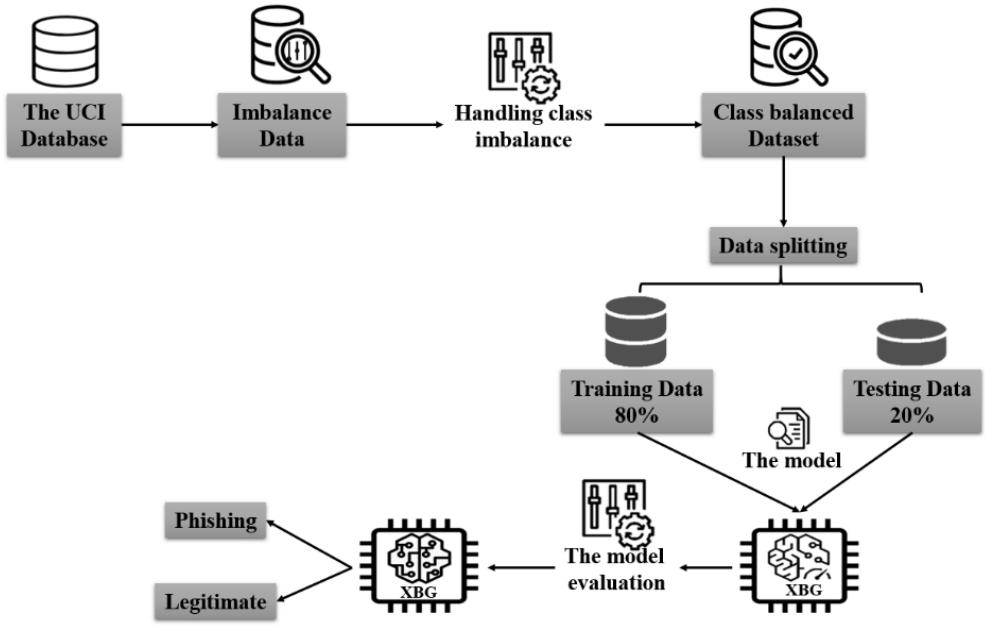

Figure 3. Proposed framework

This dataset is split into training and testing sets, with 80% of the data used for training and 20% reserved for testing, following widely accepted recommendations in machine learning literature. The training data, now balanced, is used to train the XGBoost classifier, while the testing data is reserved for model evaluation. Post-training, the XGBoost model is applied to classify new data into phishing or legitimate categories. The model's effectiveness is then assessed using metrics such as accuracy, F1-score, recall, precision, receiver operating characteristic-area under the curve (ROC-AUC), and geometric mean score (G-mean), demonstrating the impact of hybrid sampling techniques on improving model performance.

4.1. Comparison of class distributions before and after apply SMOTEENN and SMOTETomek techniques

Table 5 illustrates the class distribution of the original dataset as well as the datasets obtained after applying the proposed SMOTEENN and SMOTETomek methods. Table 5 presents a comparison of sample distributions before and after applying two hybrid-sampling techniques: SMOTEENN and SMOTETomek. The "Original samples" column shows the initial count and percentage distribution of the legitimate and phishing classes, with legitimate samples at 44.301% and phishing samples at 55.698%. After applying SMOTEENN, the distribution of legitimate samples increases to 50.736% and phishing samples decrease to 49.263%, resulting in a more balanced dataset. Similarly, applying SMOTETomek results in an equal distribution of 50.00% for both legitimate and phishing samples, further balancing the dataset. This comparison highlights how each sampling technique adjusts the class distribution to address class imbalance.

Table 5. The class distribution

Class	Legitimate	Phishing
Original samples	4,898	6,157
Distribution (%)	44.301	55.698
SMOTEENN samples	4,926	4,783
Distribution (%)	50.736	49.263
SMOTETomek samples	4,950	4,950
Distribution (%)	50.00	50.00

4.2. Evaluation metrics

Evaluation metrics play a vital role in assessing the performance of machine learning models. They provide quantitative measures such as accuracy, precision, recall, F1-score, ROC-AUC, and the G-mean, which are essential for comparing different models. These metrics help researchers determine which approach is best suited for their specific tasks, ensuring that the chosen model effectively handles the challenges of the problem at hand.

i) Accuracy: accuracy measures the overall correctness of a model, calculated as the ratio of correctly predicted instances to the total instances in the dataset. While it is intuitive and straightforward, accuracy may not be ideal for imbalanced datasets as it can be misleading when one class dominates, as defined in (2).

$$Accuracy = \frac{Number\ of\ correctly\ predicted\ instances}{Total\ number\ of\ instances} \quad (2)$$

ii) F1-score: the F1-score is a balanced metric that considers both precision and recall. It is especially useful for imbalanced datasets, as it calculates the harmonic mean of precision and recall, providing a single value that balances the trade-off between false positives and false negatives, as shown in (3).

$$F1 - score = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (3)$$

iii) Recall: recall, also known as sensitivity or true positive rate, measures the proportion of actual positive instances correctly identified by the model. It is crucial when minimizing false negatives is essential, focusing on correctly identifying as many positive instances as possible, as defined in (4).

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives} \quad (4)$$

iv) Precision: precision represents the proportion of true positive predictions among all positive predictions made by the model. It is vital when the cost of false positives is high, aiming to reduce the number of incorrectly classified positive instances, as shown in (5).

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives} \quad (5)$$

v) ROC-AUC: the ROC-AUC measures the model's ability to distinguish between classes [26]. It plots the true positive rate (recall) against the false positive rate, providing a single value that summarizes the model's performance across all classification thresholds. A higher ROC-AUC indicates better model performance.

vi) G-mean: the G-mean is a metric that balances the performance of a model across both classes, especially in imbalanced datasets. It is calculated as the square root of the product of the true positive rate (recall) and the true negative rate (specificity). G-mean ensures that the model performs well on both the minority and majority classes, providing a balanced evaluation [27].

In summary, a combination of multiple evaluation metrics provides a comprehensive and reliable assessment of model performance. Specifically, accuracy measures overall correctness, while the F1-score balances precision and recall to handle imbalanced datasets effectively. In addition, recall emphasizes minimizing false negatives, precision focuses on reducing false positives, ROC-AUC evaluates the model's ability to distinguish between classes, and the G-mean ensures balanced performance across all classes.

5. RESULTS AND DISCUSSION

This study compares the performance of the XGB model, which doesn't address class imbalance, to two advanced methods specifically designed to handle this issue in phishing website detection: SMOTEENN and SMOTETomek. These methods combine sophisticated resampling techniques with the XGB algorithm to improve detection performance, creating a more balanced class distribution and potentially making the model more effective. The results obtained from the experimentation using the scikit-learn tool highlight a comparison between three classifiers XGB, SMOTEENN-XGB, and SMOTETomek-XGB, as shown in Table 6. Among them, SMOTETomek-XGB marginally outperforms the others in key metrics, particularly in accuracy (0.978), F1-score (0.981), recall (0.982), and the G-mean (0.978), positioning it as the most balanced and effective approach for addressing class imbalance in phishing website detection.

Although XGB does not specifically target class imbalance, it performs remarkably well, achieving a high ROC-AUC (0.998) and strong overall accuracy (0.977), demonstrating its robustness. SMOTEENN-XGB excels in precision (0.982), effectively minimizing false positives, but exhibits a minor trade-off in recall (0.972) and the G-mean (0.975) when compared to SMOTETomek-XGB. This analysis suggests that while all three methods are highly effective, SMOTETomek-XGB provides the best balance between sensitivity and specificity, making it the most suitable choice for this application.

Table 6. Evaluation results in (%)

Classifier	Accuracy	F1-score	Recall	Precision	ROC-AUC	G-mean
XGB	0.977	0.980	0.979	0.980	0.998	0.977
SMOTEENN-XGB	0.974	0.977	0.972	0.982	0.997	0.975
SMOTETomek-XGB	0.978	0.981	0.982	0.979	0.998	0.978

After a thorough examination of our research findings, we will proceed to conduct a comparative analysis with other relevant studies in the field (Table 7). In comparison to the referenced studies, our classifiers, particularly SMOTETomek-XGB and XGB, exhibit superior performance, with SMOTETomek-XGB achieving an accuracy of 97.8% and XGB reaching 97.7%. These results slightly surpass the best accuracies reported by Alsariera *et al.* [28] (97.4%) and Alnemari and Alshammari [29] (97.3%) using advanced ensemble methods and RF. Notably, the models not only match or exceed these accuracies but also demonstrate excellence in other key metrics, such as F1-score, recall, precision, ROC-AUC, and G-mean. This indicates that the hybrid sampling techniques combined with XGB in our study provide a more balanced and robust approach to phishing website detection compared to the methods employed in the referenced research.

Table 7. Evaluation of existing phishing domain detection models

Authors	Dataset	Algorithm	Accuracy (%)
Ubing *et al.* [30]	UCI	Ensemble bagging, boosting, and stacking	95.4
Alsariera *et al.* [28]	UCI	ForestPA-PWDM, Bagged-ForestPA-PWDM, and Adab-ForestPA-PWDM	96.26, 96.5, and 97.4
Lakshmi *et al.* [31]	UCI	DNN + Adam	96.00
Alnemari and Alshammari [29]	UCI	RF	97.3
Omari [8]	UCI	Gradient boost	97.2

6. CONCLUSION

This study has underscored the critical importance of addressing class imbalance in phishing website detection through the application of advanced resampling techniques. By comparing three approaches XGBoost, SMOTEENN-XGB, and SMOTETomek-XGB this research highlights the effectiveness of integrating hybrid sampling methods with a robust classifier like XGBoost. The results indicate that while all methods perform exceptionally well, SMOTETomek-XGB offers the most balanced performance across key metrics, providing an optimal trade-off between sensitivity and specificity. This makes it particularly well-suited for phishing detection tasks, where both false positives and false negatives can have significant consequences. The findings emphasize that careful consideration of class imbalance strategies can significantly enhance the performance of machine learning models in cybersecurity applications, ultimately contributing to more reliable and accurate detection systems. Future research could explore the integration of these techniques with other classifiers or in different domains where class imbalance is prevalent, thereby broadening the applicability of these findings.

FUNDING INFORMATION

Authors state that no funding was received for this research. This work was conducted independently without financial support from any funding agency, institution, or organization. Consequently, no grant numbers or external sponsorships are associated with this study.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Kamal Omari	✓	✓				✓	✓	✓	✓			✓	✓	
Ayoub Oukhatar	✓		✓	✓	✓					✓	✓			

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state that they have no conflict of interest. The authors declare that they have no known financial, personal, or professional relationships that could have influenced the work reported in this paper.

DATA AVAILABILITY

The data that support the findings of this study are openly available in UC Irvine Machine Learning Repository at https://archive.ics.uci.edu/ml/datasets/phishing+websites, reference [22].

REFERENCES

[1] APWG, "Phishing activity trends report 1st quarter (2024): unifying the global response to cybercrime," *docs.apwg.org*. 2024. Accessed: Jul. 25, 2024. [Online]. Available: https://docs.apwg.org/reports/apwg_trends_report_q1_2024.pdf

[2] K. Nagaraj, B. Bhattacharjee, A. Sridhar, and S. GS, "Detection of phishing websites using a novel twofold ensemble model," *Journal of Systems and Information Technology*, vol. 20, no. 3, pp. 321–357, Nov. 2018, doi: 10.1108/JSIT-09-2017-0074.

[3] M. Buda, A. Maki, and M. A. Mazurowski, "A systematic study of the class imbalance problem in convolutional neural networks," *Neural Networks*, vol. 106, pp. 249–259, Oct. 2018, doi: 10.1016/j.neunet.2018.07.011.

[4] M. Khushi *et al.*, "A comparative performance analysis of data resampling methods on imbalance medical data," *IEEE Access*, vol. 9, pp. 109960–109975, 2021, doi: 10.1109/ACCESS.2021.3102399.

[5] D. Elreedy, A. F. Atiya, and F. Kamalov, "A theoretical distribution analysis of synthetic minority oversampling technique (SMOTE) for imbalanced learning," *Machine Learning*, vol. 113, no. 7, pp. 4903–4923, Jul. 2024, doi: 10.1007/s10994-022-06296-4.

[6] C. Bentéjac, A. Csörgő, and G. M. -Muñoz, "A comparative analysis of gradient boosting algorithms," *Artificial Intelligence Review*, vol. 54, no. 3, pp. 1937–1967, Mar. 2021, doi: 10.1007/s10462-020-09896-5.

[7] N. Abdelhamid, F. Thabtah, and H. A. -Jaber, "Phishing detection: a recent intelligent machine learning comparison based on models content and features," in *2017 IEEE International Conference on Intelligence and Security Informatics (ISI)*, Jul. 2017, pp. 72–77, doi: 10.1109/ISI.2017.8004877.

[8] 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, doi: 10.14569/IJACSA.2023.0140945.

[9] K. Subashini and V. Narmatha, "Detecting phishing websites using recent techniques: a systematic literature review," *ITM Web of Conferences*, vol. 57, Nov. 2023, doi: 10.1051/itmconf/20235701008.

[10] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, Jun. 2002, doi: 10.1613/jair.953.

[11] E. F. Swana, W. Doorsamy, and P. Bokoro, "Tomek link and SMOTE approaches for machine fault classification with an imbalanced dataset," *Sensors*, vol. 22, no. 9, Apr. 2022, doi: 10.3390/s22093246.

[12] Z. Wang, C. Wu, K. Zheng, X. Niu, and X. Wang, "SMOTETomek-based resampling for personality recognition," *IEEE Access*, vol. 7, pp. 129678–129689, 2019, doi: 10.1109/ACCESS.2019.2940061.

[13] M. Lamari *et al.*, "SMOTE–ENN-based data sampling and improved dynamic ensemble selection for imbalanced medical data classification," in *Advances on Smart and Soft Computing*, Singapore: Springer, 2021, pp. 37–49, doi: 10.1007/978-981-15-6048-4_4.

[14] M. E. Lokanan, "Predicting mobile money transaction fraud using machine learning algorithms," *Applied AI Letters*, vol. 4, no. 2, Apr. 2023, doi: 10.1002/ail2.85.

[15] Z. Xu, D. Shen, T. Nie, and Y. Kou, "A hybrid sampling algorithm combining M-SMOTE and ENN based on Random forest for medical imbalanced data," *Journal of Biomedical Informatics*, vol. 107, Jul. 2020, doi: 10.1016/j.jbi.2020.103465.

[16] F. Yang, K. Wang, L. Sun, M. Zhai, J. Song, and H. Wang, "A hybrid sampling algorithm combining synthetic minority over-sampling technique and edited nearest neighbor for missed abortion diagnosis," *BMC Medical Informatics and Decision Making*, vol. 22, no. 1, Dec. 2022, doi: 10.1186/s12911-022-02075-2.

[17] M. Isangediok and K. Gajamannage, "Fraud detection using optimized machine learning tools under imbalance classes," in *2022 IEEE International Conference on Big Data (Big Data)*, Dec. 2022, pp. 4275–4284, doi: 10.1109/BigData55660.2022.10020723.

[18] I. D. Mienye and Y. Sun, "A deep learning ensemble with data resampling for credit card fraud detection," *IEEE Access*, vol. 11, pp. 30628–30638, 2023, doi: 10.1109/ACCESS.2023.3262020.

[19] N. L. Fitriyani, M. Syafrudin, G. Alfian, and J. Rhee, "Development of disease prediction model based on ensemble learning approach for diabetes and hypertension," *IEEE Access*, vol. 7, pp. 144777–144789, 2019, doi: 10.1109/ACCESS.2019.2945129.

[20] L. Boratto, S. Carta, W. Iguider, F. Mulas, and P. Pilloni, "Fair performance-based user recommendation in eCoaching systems," *User Modeling and User-Adapted Interaction*, vol. 32, no. 5, pp. 839–881, Nov. 2022, doi: 10.1007/s11257-022-09339-6.

[21] R. Mohammad and L. McCluskey, "Phishing websites," *UCI Machine Learning Repository*, 2012, doi: 10.24432/C51W2X.

[22] T. Chen and C. Guestrin, "XGBoost: a scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Aug. 2016, pp. 785–794, doi: 10.1145/2939672.2939785.

[23] M. Bahaghighat, M. Ghasemi, and F. Ozen, "A high-accuracy phishing website detection method based on machine learning," *Journal of Information Security and Applications*, vol. 77, Sep. 2023, doi: 10.1016/j.jisa.2023.103553.
[24] 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, no. 5, May 2022, doi: 10.1007/s10064-022-02708-w.
[25] S. S. Dhaliwal, A.-A. Nahid, and R. Abbas, "Effective intrusion detection system using XGBoost," *Information*, vol. 9, no. 7, Jun. 2018, doi: 10.3390/info9070149.
[26] G. Varoquaux and O. Colliot, "Evaluating machine learning models and their diagnostic value," in *Machine Learning for Brain Disorders*, New York: Humana, 2023, pp. 601–630, doi: 10.1007/978-1-0716-3195-9_20.
[27] R. M. Vogel, "The geometric mean?," *Communications in Statistics - Theory and Methods*, vol. 51, no. 1, pp. 82–94, Jan. 2022, doi: 10.1080/03610926.2020.1743313.
[28] Y. A. Alsariera, A. V. Elijah, and A. O. Balogun, "Phishing website detection: forest by penalizing attributes algorithm and its enhanced variations," *Arabian Journal for Science and Engineering*, vol. 45, no. 12, pp. 10459–10470, Dec. 2020, doi: 10.1007/s13369-020-04802-1.
[29] S. Alnemari and M. Alshammari, "Detecting phishing domains using machine learning," *Applied Sciences*, vol. 13, no. 8, Apr. 2023, doi: 10.3390/app13084649.
[30] A. A. Ubing, S. Kamilia, A. Abdullah, N. Jhanjhi, and M. Supramaniam, "Phishing website detection: an improved accuracy through feature selection and ensemble learning," *International Journal of Advanced Computer Science and Applications*, vol. 10, no. 1, pp. 252–257, 2019, doi: 10.14569/IJACSA.2019.0100133.
[31] L. Lakshmi, M. P. Reddy, C. Santhaiah, and U. J. Reddy, "Smart phishing detection in web pages using supervised deep learning classification and optimization technique ADAM," *Wireless Personal Communications*, vol. 118, no. 4, pp. 3549–3564, Jun. 2021, doi: 10.1007/s11277-021-08196-7.

BIOGRAPHIES OF AUTHORS

Kamal Omari holds a Ph.D. in Computer Science. Currently, he serves as an associate professor at the Multidisciplinary Faculty of Ouarzazate, Ibn Zohr University, Agadir, Morocco. He can be contacted at email: k.omari@uiz.ac.ma.

Ayoub Oukhatar is a professor in Higher School of Technology of Ouarzazate, Ibn Zohr University, Agadir, Morocco. He has a Ph.D. in Computer Sciences and Networks from Faculty of Sciences of Meknes, Moulay Ismail University, Morocco, and he has an engineer degree in Telecommunications and Networks System from National School of Applied Sciences of Tetuan in 2014, Abdelmalek Essadi University. His research interests include wireless nano sensor networks, distributed system, smart grid network, blockchain technology, and internet of things. He can be contacted at email: a.oukhatar@uiz.ac.ma.