An Exploratory Study of Classification Strategies for Imbalanced Phishing Email Dataset Using Naive Bayes and Logistic Regression

Asja Bašović

Abstract - In this study, the performance of two classic machine learning algorithms, Naïve Bayes and Logistic Regression, is evaluated on a highly imbalanced email phishing dataset. The goal is not to achieve the highest accuracy and build the best-performing model, but to analyze how each of these algorithms behaves under class imbalance and what strategies would help improve their ability to classify the minority class correctly, specifically regarding email phishing. The priority was to increase recall while monitoring for precision and F-score, which is the usual approach when dealing with class imbalance. Strategies applied and evaluated are class weighting, Synthetic Minority Oversampling Technique (SMOTE), undersampling, and feature engineering. Each approach was implemented systematically in order to understand the impact on each model's classification performance. In addition, a hybrid approach was implemented and examined on the Logistic Regression model, which consisted of class weighting and SMOTE techniques applied at the same time.

Index Terms - Classification, Email phishing, Imbalanced dataset, Logistic Regression, Naïve Bayes

INTRODUCTION

Phishing is a form of fraud in which the attacker represents as genuine and attacks using communication channels, primarily emails [1]. Email phishing attacks are attacks that deceive recipients into revealing sensitive information or downloading malicious software, leading to identity theft, security risks, and loss of information [2]. Usually, attackers aim to exploit the trust and familiarity people have in major known brands or institutions to make users interact with malicious links or attachments in phishing emails [3]. Despite the dangers of phishing emails, people often do not know what to look for to detect such an email [4]. To help bridge this gap and detect phishing emails more efficiently, machine learning approaches have emerged as promising solutions, analyzing content, URLs, and words in web pages to identify malicious emails [2]. However, practical implementation faces significant challenges due to severe class imbalance in real-world datasets where phishing emails are outnumbered. The main concern with imbalanced classes is their ability to significantly decrease the performance of many standard learning models [5]. In such imbalanced datasets, classifiers tend to favor the majority class (legitimate emails), yielding deceptively high accuracy while missing most phishing messages which is a critical failure given the cost of false negatives [6]. Hence, accuracy often is not a reliable prediction for the minority class. Naive Bayes and Logistic Regression are commonly applied algorithms for email classification tasks. Naive Bayes is a probabilistic classifier that assumes feature independence and that all features contribute to the outcome equally [7]. Despite the often unrealistic assumptions, the algorithm remains effective in phishing email detection, where the model classifies emails by computing the posterior probability of each class [7], [8]. Logistic Regression is a statistical technique for binary classification tasks, such as phishing email or spam email detection, for which various email features are analyzed to determine the likelihood that an email is malicious [7], [9]. Each feature has a weight assigned by the model, which collectively contributes to the final probability score. If the score is above a certain threshold, the email is classified as phishing [7]. However, their effectiveness on severely imbalanced phishing datasets remains unexplored, particularly regarding responses to preprocessing strategies. Multiple techniques address class imbalance, including oversampling, undersampling, hybrid approaches, and feature engineering. SMOTE (Synthetic Minority Over-sampling Technique) is a widely used and popular approach that synthetically generates new minority class samples along the feature-space vector between existing instances and their nearest neighbors [10], [11]. Undersampling, however, removes majority samples until the proportion of the two classes is balanced and helps reduce training time [12]. In theory, it could help models identify the minority class better. Similarly, feature engineering can be very impactful in enhancing the performance of text classification models. Incorporating polynomial interaction terms enables models to capture complex, non-linear relationships between features, which can significantly improve predictive accuracy [13]. Feature selection is equally critical with effective feature selection techniques aim to retain the most informative features while discarding irrelevant or redundant ones, thereby improving model accuracy and efficiency [14]. Lastly, class weight balancing in Logistic Regression adjusts the model's loss function to penalize misclassifications of the minority class more heavily, improving the detection of fraudulent transactions [19]. This study addresses the limitations of imbalanced classes through systematic comparative analysis of Naive Bayes and Logistic Regression on phishing data. This paper aims to investigate the effect of class weighting, SMOTE, undersampling, hybrid approaches, and feature engineering techniques (polynomial interactions) on the classification of the minority class in an imbalanced email phishing dataset. The aim is to provide a systematic evaluation of two widely-used algorithms on realistic imbalanced phishing data, a comprehensive analysis of preprocessing strategies and their algorithm-specific effects, and practical insights for cybersecurity practitioners.

RELATED WORKS

Class imbalance is recognized as a fundamental challenge in cybersecurity machine learning. Chawla et al. (2002) highlighted how traditional accuracy metrics can be misleading in imbalanced scenarios, since a model can achieve high overall accuracy by simply predicting the majority class every time [10]. Naive Bayes has long been a baseline model for email classification tasks due to its probabilistic framework and computational efficiency. It continues to be used in imbalanced settings, including phishing and spam detection, though its performance can be limited when the data distribution is skewed. In a recent study comparing classifiers on phishing-related datasets, Altwaijry et al. reported that "the naïve Bayes classifier achieved an accuracy of 87.63% for the Spam Corpus and 79.53% for the Spambase datasets" [15], highlighting the model's applicability and performance limitations in real-world email data. Logistic Regression has been similarly adopted in phishing and spam classification, where its probabilistic outputs and interpretability make it a practical choice. Gontla et al. have implemented various classification models, including Logistic Regression, to detect detect phishing websites [16]. The Logistic Regression model achieved 91.73% accuracy rate. Similarly, Eftimie et al. explored the psychological aspects of social engineering by analyzing personality traits in the context of spear-phishing attacks with the help of logistic regression analyses performed at each phase of the phishing attack [17]. Several studies have employed the Synthetic Minority Oversampling Technique (SMOTE) to address the challenge of class imbalance in classification tasks. For instance, Uyun and Sulistyowati [18] integrated SMOTE with bootstrapping to improve the classification of river water quality statuses across four classes. The study adopted random oversampling within the SMOTE procedure to synthetically increase the representation of minority classes, followed by bootstrapping, resulting in significantly improved classification accuracy from 83.3% to 98.8%. Moreover, class imbalance is a common challenge in predictive modeling, particularly when minority class events are rare but critical to identify. One effective strategy employed in recent studies is the use of class weighting in logistic regression. Botchey et al. [19] investigated mobile money fraud detection and addressed the issue of skewed class distributions by applying class weighting in logistic regression using the class_weight='balanced' option from scikit-learn. Their results showed improved classification performance compared to standard logistic regression and SMOTE-based methods. Similarly, Wang et al. [20] employed class-weighted logistic regression to predict the risk of chronic obstructive pulmonary disease (COPD) from imbalanced health survey data. Their study demonstrated that logistic regression with class weighting achieved the best trade-off between precision and recall among several machine learning models, outperforming even SMOTE-enhanced ensemble models. Feature engineering has emerged as a critical strategy for enhancing model performance, particularly when addressing class imbalance issues in medical datasets. Thakur et al. [21] investigated diabetes prediction during the COVID-19 pandemic and applied advanced feature engineering techniques-including the generation of interaction and polynomial features to better capture complex nonlinear relationships in the data (e.g. creating interaction terms like Age × BMI). The augmentation of feature space through nonlinear transformations helped effectively address the challenges posed by data imbalance.

DATASET

For this research, Email Phishing Dataset [22] was used, which contains cleaned and feature-engineered email data specifically designed for phishing detection tasks. The dataset, published by Cratchley on Kaggle in April 2025 under the Apache 2.0 license, provides preprocessed and cleaned email samples with engineered features that facilitate machine learning-based phishing detection research. The Apache 2.0 license permits unrestricted use, modification, and distribution of the dataset for both commercial and non-commercial purposes. The Email Phishing Dataset comprises nine engineered features extracted from email content, designed to facilitate automated phishing detection without exposing sensitive information. The dataset contains exclusively numerical features derived from textual analysis with 524,846 samples, with no raw email content or headers included to ensure privacy protection. The feature set encompasses both lexical and structural characteristics of email communications. Lexical features include the total word count (num_words), unique word diversity (num_unique_words), and frequency of common stopwords (num_stopwords) based on a standardized English stopword list. Content quality is assessed through spelling accuracy (num_spelling_errors) using the pyspellchecker library applied to filtered tokens. Behavioral indicators are captured through the presence of urgent language patterns (num_urgent_keywords), which quantifies occurrences of keywords such as "urgent," "verify," and "update" commonly associated with phishing attempts. Structural features focus on embedded elements within emails, including hyperlink frequency (num_links), domain diversity (num_unique_domains), and embedded email address counts (num_email_addresses). The target variable (label) provides binary classification labels, where 0 indicates legitimate emails and 1 denotes phishing attempts. All analysis was done using Python and relevant libraries.

EXPLORATORY DATA ANALYSIS (EDA)

An Exploratory Data Analysis (EDA) was conducted as an initial analysis to understand the dataset's structure and characteristics. The analysis showed that the dataset had no missing values and that all columns held numerical data. Distribution plots (see Figure 1) further revealed that most features are highly skewed, with most data points concentrated around zero. Most notably, the dataset is highly imbalanced, with approximately 1.3% of samples labeled as phishing (517897 labeled as not phishing emails and 6949 as phishing emails) as shown on Figure 2. This analysis confirmed the need for methods that would help manage the imbalance and feature skew. We generated a Pearson correlation matrix to analyze relationships between features (see Fig. 3). High correlations were observed among num_words, num_stopwords, and num_spelling_errors (≥ 0.85), which may indicate redundancy that can negatively affect models like Logistic Regression. Moderate correlation was found (0.62) between num_links and num_unique_domains, while, overall, very low correlation was noted with the target variable (label), suggesting no individual feature alone is strongly predictive. All correlations can be found on Figure 3. This is typical in real-world data, where complex combinations of features often matter more than individual ones. While exact multicollinearity was not detected, the presence of correlated groups may affect the stability of Logistic Regression and violate the independence assumption of Naive Bayes.

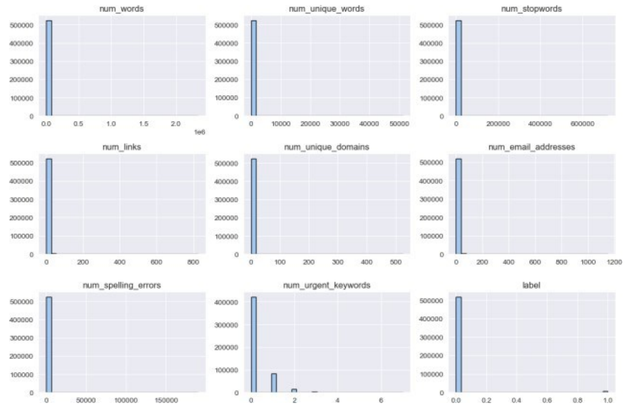


FIGURE 1. DISTRIBUTION PLOTS SHOWING SKEWNESS OF THE DATA.

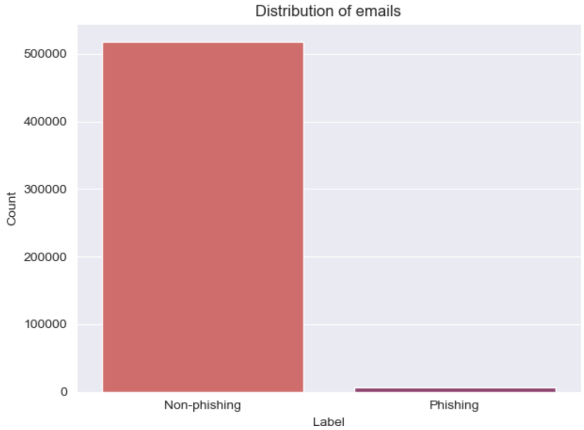


FIGURE 2. COUNTPLOT SHOWING THE AMOUNT OF PHISING VS. NON-PHISHING EMAILS IN THE DATASET USED.

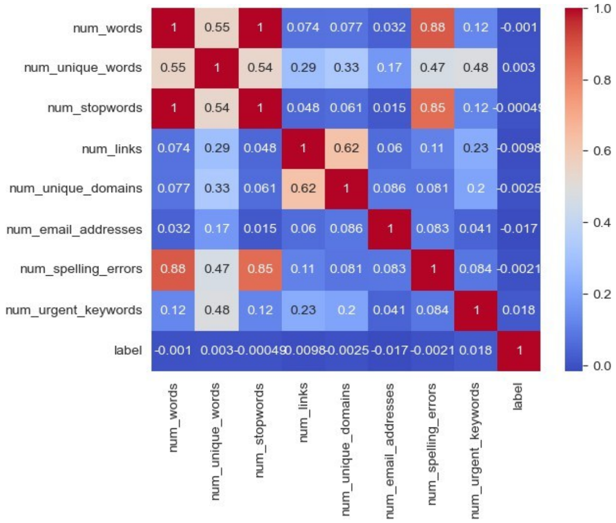


FIGURE 3. CORRELATION MATRIX OF FEATURES IN THE DATASET

Methodology

I. Models

This study evaluates two classical machine learning algorithms: Naive Bayes and Logistic Regression. For Naive Bayes, the Complement Naive Bayes (ComplementNB) variant was used due to its higher robustness for imbalanced data and lack of assumption of Gaussian-distributed features. This choice is motivated by the skewed nature of the dataset, which violates the assumptions of GaussianNB. Logistic Regression is used as a linear classifier that estimates class probabilities via the logistic function. Due to its reliance on feature coefficients, it is sensitive to multicollinearity and requires careful feature scaling.

II. Preprocessing

To prepare the data for each model, the dataset was split into training and testing sets using a 70-30 stratified split, maintaining the original class distribution in both sets. The 70% training portion provides adequate data to learn from the rare phishing class while ensuring an unbiased 30% evaluation set. The decision to use a stratified split was to make sure the split does not randomly split the data in a way that excludes the minority class, which would make the model biased and unreliable. Scaling was performed to ensure uniformity in magnitude and to support algorithmic stability but only for the Logistic Regression model as the Naive Bayes model is generally less sensitive to feature scaling. Both StandardScaler and RobustScaler were evaluated. The StandardScaler transforms features to have zero mean and unit variance, making it suitable for linear models like Logistic Regression. However, due to the dataset's skewed distributions, RobustScaler was also considered. The RobustScaler uses the median and interquartile range.

III. Handling imbalance

The dataset exhibited extreme class imbalance, with phishing emails comprising about 1.3% of all samples. To address this synthetic oversampling and undersampling. For Logistic Regression, the Synthetic Minority Over-sampling Technique (SMOTE) was used to generate synthetic phishing samples in the training set, which balanced the class distribution. SMOTE increases minority class representation. Additionally, the impact of class weighting was checked by setting class_weight='balanced', which penalizes misclassification of the minority class during model training. This dual approach aimed to improve recall for the phishing class without excessively compromising precision. For Naive Bayes, SMOTE was initially avoided due to its sensitivity to synthetic data that may not preserve the original feature distributions, particularly when features are skewed or correlated. This sensitivity stems from Naive Bayes' assumption of feature independence, which can be violated by the introduction of synthetic samples However, to investigate performance improvements, undersampling of the majority class was explored as a compatible alternative. This reduces the imbalance without introducing synthetic data.

IV. Feature Engineering

To enhance the model's classification of the smaller class, feature engineering techniques were applied. They include Polynomial Interaction Terms, and Low-Variance Feature Removal. Polynomial interaction terms were generated using scikit-learn's PolynomialFeatures, and a controlled comparison was made to evaluate how the model behaves with and without polynomial interaction terms. These terms were incorporated to capture non-linear relationships between features, addressing a fundamental limitation of linear models such as Logistic Regression, which cannot inherently model feature interactions. Both model configurations employed class_weight='balanced' to address class imbalance, and Min-Max scaling was applied when interaction terms were included to ensure numerical stability. Only interaction terms (no higher-degree powers) were included to avoid unnecessary model complexity. A low-variance filter was applied to eliminate features that have little to no variance across the dataset, as such features provide minimal predictive value.

Results

I. Baseline Model Performance

To establish a reference point for evaluating the effectiveness of the applied preprocessing and imbalance-handling strategies, we first assessed the performance of both Naive Bayes and Logistic Regression using the raw dataset. At this stage, only scaling was performed using RobustScaler for the Logistic Regression model, which requires it. Models were trained on the original, highly imbalanced dataset using default settings and evaluated on a stratified 30% test split that preserved the original class distribution. The classification threshold was set at the default value of 0.5. Table 1 presents the baseline performance metrics, including accuracy, precision, recall, and F1-score.

TABLE 1. BASELINE MODELS' PERFORMANCE METRIX.

Model	Accuracy	Precision	Recall	F1-Score
ComplementNB	0.31	0.02	0.90	0.03
Logistic Regression	0.99	1.00	0.00	0.00

Despite achieving a high overall accuracy of 0.99, the baseline Logistic Regression model failed to detect phishing emails altogether, with a recall of 0.00 for the minority class. This result shows a common pitfall in imbalanced classification, which is that high accuracy may simply reflect the dominance of the majority class rather than meaningful predictive performance. The model consistently favoured non-phishing predictions, thereby missing the phishing instances. This is a critical failure in security applications where false negatives can be costly. In contrast, the baseline Complement Naive Bayes (ComplementNB) model demonstrated a different trade-off. While the overall accuracy is lower than that of the Logistic Regression model at 0.31, it achieved a much higher recall of 0.90 for phishing emails. This suggests a stronger ability to identify rare class instances. However, this came at the expense of precision, which fell to 0.02, indicating a high rate of false positives. Notably, ComplementNB also misclassified many non-phishing emails (non-phishing recall: 0.30), reflecting difficulty in maintaining specificity under imbalance. These results underscore the limitations of both models in their unmodified forms. The poor F1-scores reinforce the need for specialized strategies to address the underlying class distribution.

II. Effect of scaling

The impact of feature scaling on logistic regression performance was evaluated using both StandardScaler and RobustScaler. For this dataset, neither scaling method meaningfully improved classification performance for the minority class. StandardScaler achieved 0 true positive predictions for phishing emails, while RobustScaler detected only 1 phishing email out of 2,085 (recall = 0.0005) (see Figure 4 and Figure 5). Both methods achieved similar overall metrics: precision, recall, and F1-score of 0.00 for the minority class, with overall accuracy of 99% driven primarily by correct classification of the majority class as seen in Table 2.

TABLE 2. BASELINE LOGISTIC REGRESSION MODEL PERFORMANCE WITH DIFFERENT SCALING METHODS.

Model	**Scaler**	**Precision**	**Recall**	**F1-Score**	**Accuracy**
Logistic Regression	Standard Scaler	0.0	0.0005	0.00	0.99
Logistic Regression	Robust Scaler	1.0	0.00	0.00	0.99

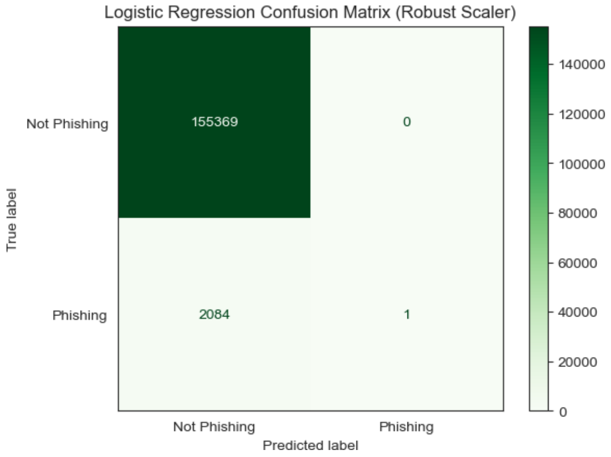


FIGURE 4. CONFUSION MATRIX OF LOGISTIC REGRESSION WHEN USING ROBUST SCALER.

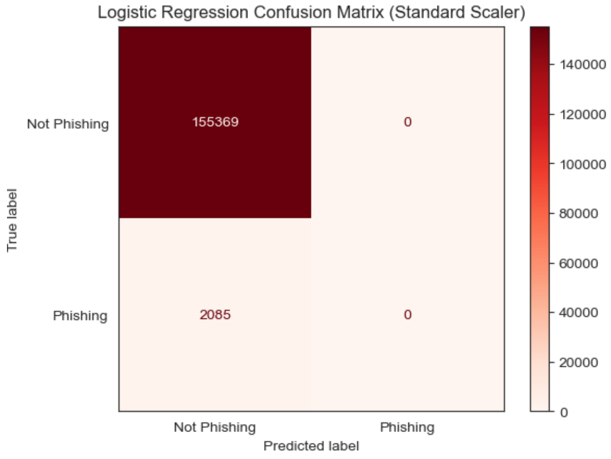


FIGURE 5. CONFUSION MATRIX OF LOGISTIC REGRESSION WHEN USING STANDARD SCALER.

III. Logistic Regression and Class Weight Adjustment

The class_weight='balanced' parameter was used to change the cost function of the Logistic Regression model. This parameter automatically modifies class weights in a way that is inversely proportional to class frequencies. This algorithmic modification fundamentally alters the optimization objective, imposing higher penalties for misclassifying minority class samples during training. The model's behavior drastically changed once balanced class weights were applied. After class weight modification, the baseline Logistic Regression model, which was unable to detect any phishing emails, saw a significant increase, with recall rising to 0.77. However, the accuracy dropped to 0.02 as a result of this improvement, suggesting that although the model was able to identify the majority of phishing emails, it also produced a large number of false positives. The basic conflict between sensitivity and specificity in imbalanced classification scenarios. The confusion matrix of this model can be seen in Figure 6.

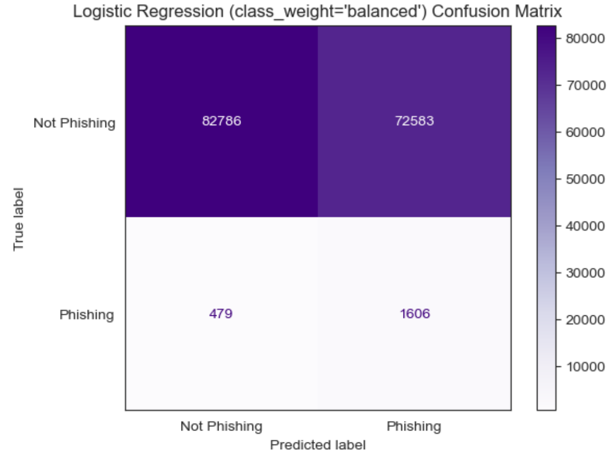


FIGURE 6. CONFUSION MATRIX OF THE LOGISTIC REGRESSION MODEL USING CLASS WEIGHT BALANCING.

V. Logistic Regression and SMOTE Oversampling Strategy

Synthetic Minority Oversampling Technique (SMOTE) was employed to address class imbalance at the data level by generating synthetic minority class samples in the feature space. Two distinct SMOTE configurations were evaluated to assess the impact of oversampling intensity:

1) Default SMOTE (sampling_strategy=1.0): Complete balancing strategy where synthetic samples are generated to achieve equal class distribution (50:50 ratio)
2) Conservative SMOTE (sampling_strategy=0.3): Partial balancing strategy limiting minority class representation to 30% of the majority class

When applied independently with default configuration, SMOTE demonstrated modest improvements over the baseline model, increasing recall from 0.00 to 0.05 and achieving precision of 0.07. While these improvements were marginal, they represented a meaningful advancement in minority class detection capability. The conservative SMOTE configuration (sampling_strategy=0.3) yielded comparable results, suggesting that aggressive oversampling may not always be necessary for performance gains. Detailed comparison between default and conservative SMOTE strategies revealed that the default SMOTE configuration (sampling_strategy=1.0) achieved superior recall (0.76) but marginally lower precision (0.02) compared to the conservative approach (recall = 0.05, precision = 0.07). The corresponding confusion matrices be found in Figure 7 and Figure 8.

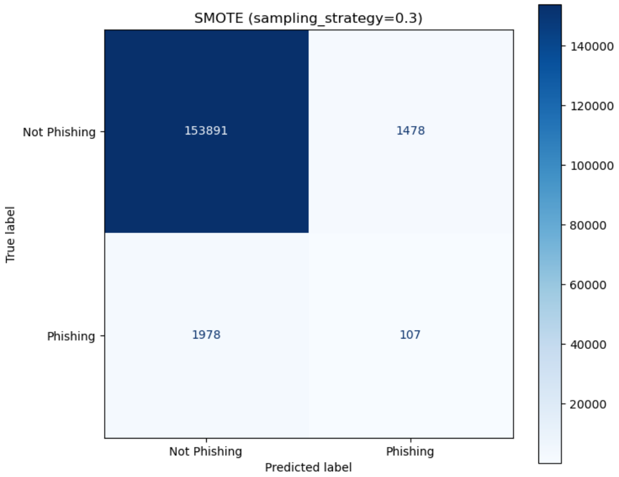


FIGURE 7. CONFUSION MATRIX OF LOGISTIC REGRESSION USING CONSERVATIVE SMOTE.

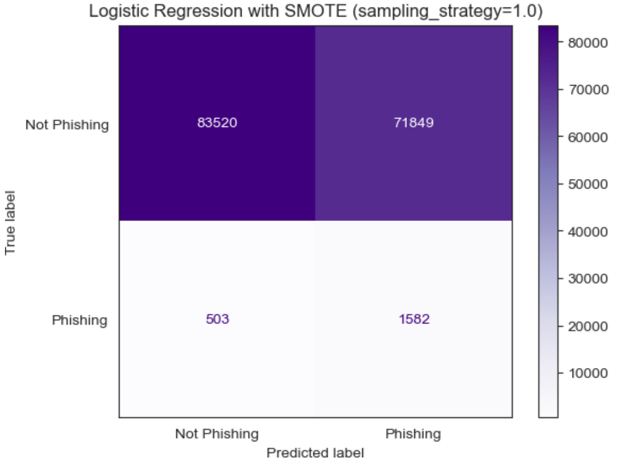


FIGURE 8. CONFUSION MATRIX OF LOGISTIC REGRESSION USING DEFAULT SMOTE.

V. Logistic Regression and Hybrid Approach

Contrary to expectations, the combination of SMOTE oversampling with balanced class weighting did not result in performance improvements beyond class weighting alone. The hybrid approach achieved the same performance metrics to class weighting in isolation: recall of 0.76 and precision of 0.02, compared to class weighting alone, which achieved recall of 0.77 and precision of 0.02. This convergence of performance metrics reveals a critical insight: the algorithmic adjustment through class weighting appears to dominate the impact of data-level modifications in this severely imbalanced scenario. The marginal difference in recall (0.77 vs 0.76) likely falls within statistical noise, suggesting that SMOTE's synthetic sample generation provides no meaningful additional benefit when combined with properly tuned class weights. Additionally, regardless of the SMOTE sampling strategy (0.3 or 1.0), the model did not change (it had the same values of every metric), while the confusion matrix can be found in Figure 9.

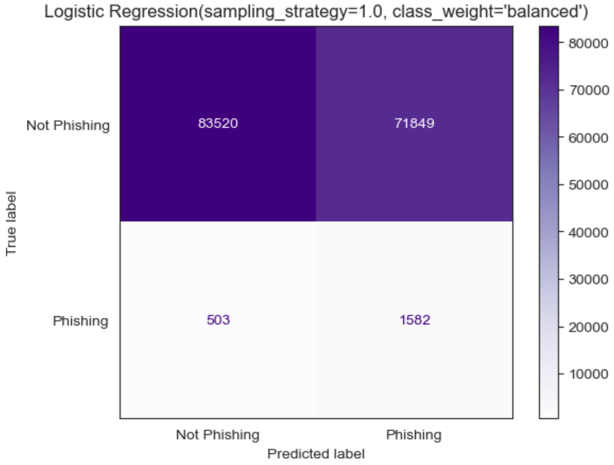


FIGURE 9. CONFUSION MATRIX OF LOGISTIC REGRESSION USING THE HYBRID APPROACH.

VI. *Naïve Bayes with Undersampling*

Given the fundamental incompatibility between SMOTE's synthetic data generation and Naive Bayes' core assumptions - specifically, the conditional independence assumption - undersampling of the majority class (non-phishing emails) was employed as an alternative balancing strategy. This approach preserves the integrity of the original feature distributions by removing excess legitimate email samples rather than introducing synthetic variations that could violate the algorithm's probabilistic framework. The undersampling procedure reduced the training dataset to achieve perfect class balance, resulting in 4,864 phishing samples and 4,864 non-phishing samples. This 1:1 ratio eliminates the prior probability bias that typically favors the majority class in imbalanced scenarios, theoretically allowing the Naive Bayes classifier to learn more balanced decision boundaries. Figure 10 shows the corresponding confusion matrix. However, the results reveal that undersampling failed to address the underlying performance issues of the Naive Bayes classifier. Precision remained low at 0.02, unchanged from the baseline performance, indicating that the model continues to generate an excessive number of false positives. This suggests that the poor performance stems not from class imbalance alone. The slight decrease in recall from 0.90 to 0.89 demonstrates a marginal reduction in the model's ability to identify phishing emails, while the negligible F1-score improvement of 0.0003 falls well within statistical noise (see Figure 11).

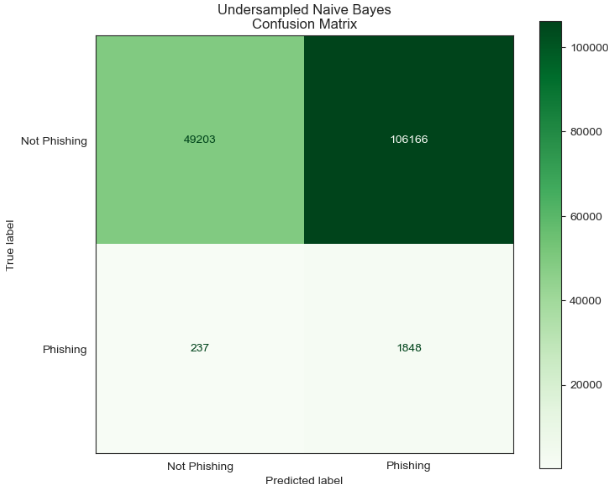
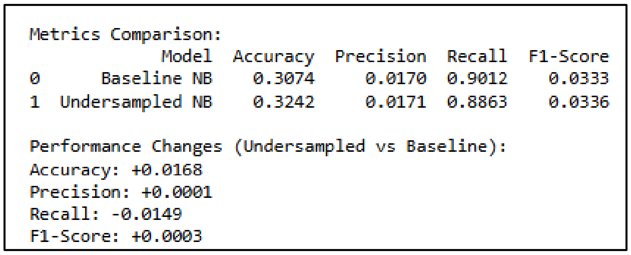


FIGURE 10. CONFUSION MATRIX OF NAÏVE BAYES USING RANDOM UNDERSAMPLING.



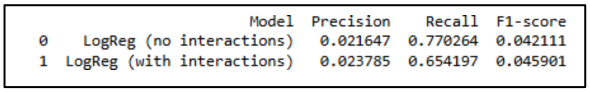
```
Metrics Comparison:
             Model  Accuracy  Precision  Recall  F1-Score
0      Baseline NB    0.3074     0.0170  0.9012    0.0333
1  Undersampled NB    0.3242     0.0171  0.8863    0.0336

Performance Changes (Undersampled vs Baseline):
Accuracy: +0.0168
Precision: +0.0001
Recall: -0.0149
F1-Score: +0.0003
```

FIGURE 11. SIDE-BY-SIDE METRICS COMPARISON OF BASELINE NAÏVE BAYES AND UNDERSAMPLED NAÏVE BAYES.

VII. *Feature Engineering Effects – Polynomial Interaction Terms*

Firstly, a low-variance filter with a threshold of 0.01 was applied to identify features with minimal variability that could potentially introduce noise without contributing predictive value. The analysis revealed that no features met the removal criteria, indicating that all engineered features retained sufficient variance and potential predictive utility.

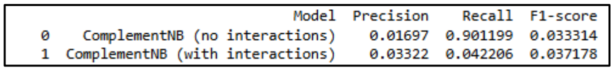
The inclusion of interaction terms yielded mixed results for Logistic Regression (Figure 12). While the F1-score demonstrated modest improvement from 0.042 to 0.046 for the phishing class, this gain came at the cost of reduced recall (0.77 to 0.65), with precision remaining constant at 0.02. This trade-off suggests that while the model became more selective in its positive predictions, it simultaneously became less sensitive to actual phishing instances.



```
                        Model  Precision    Recall  F1-score
0    LogReg (no interactions)   0.021647  0.770264  0.042111
1  LogReg (with interactions)   0.023785  0.654197  0.045901
```

FIGURE 12. SIDE-BY-SIDE METRICS COMPARISON OF BASELINE LOGISTIC REGRESSION AND LOGISTIC REGRESSION WITH INTERACTION TERMS.

The Complement Naive Bayes model showed a severe performance degradation when interaction terms were introduced, with recall decreasing from 0.90 to 0.04 despite marginal improvements in F1-score (0.03 to 0.04) and precision (0.02 to 0.03) (Figure 13). This severe recall reduction can be attributed to several interconnected factors. First, the polynomial interaction terms substantially increased the feature dimensionality, creating a high-dimensional sparse feature space potentially leading to a curse of dimensionality on the model. Second, the conditional independence assumption underlying Naive Bayes becomes more problematic when interaction terms are explicitly included. The model treats each interaction term as an independent feature, leading to redundant information that can distort the probability calculations. Third, the required Min-Max scaling transformation fundamentally altered the data distribution. Complement Naive Bayes is designed to work optimally with count-based or naturally non-negative features, but scaling transforms the feature space in ways that may not align with the algorithm's probabilistic assumptions. The scaling process can compress the natural variance patterns that Naive Bayes relies upon for classification. Finally, the increase in feature space dimensionality likely increased the class imbalance problem, making the model prefer the safety of predicting the majority class to avoid the uncertainty.



```
                             Model  Precision    Recall  F1-score
0    ComplementNB (no interactions)    0.01697  0.901199  0.033314
1  ComplementNB (with interactions)    0.03322  0.042206  0.037178
```

FIGURE 13. SIDE-BY-SIDE METRICS COMPARISON OF SCALED NAÏVE BAYES AND NAÏVE BAYES WITH INTERACTION TERMS.

DISCUSSION

The baseline results dramatically illustrate the accuracy paradox in imbalanced classification. Logistic Regression's 99% accuracy masked complete failure to detect phishing emails (0% recall), while Complement Naive Bayes's lower accuracy (31%) reflected meaningful minority class engagement (90% recall). This emphasizes that traditional accuracy metrics can be misleading in security applications, where missing a malicious email far exceeds the cost of flagging a legitimate email as suspicious. The contrasting behaviors reveal algorithmic differences: Logistic Regression favored the majority class to maximize accuracy, while Complement Naive Bayes demonstrated inherent bias toward minority class detection due to its probabilistic framework and design for handling imbalanced data. Class weighting proved most effective for Logistic Regression, increasing recall from 0% to 77% while maintaining reasonable precision by penalizing false negatives more heavily during training. Surprisingly, SMOTE oversampling provided only marginal improvements, suggesting synthetic sample generation may be less effective in high-dimensional email classification tasks where synthetic samples may not capture subtle linguistic patterns distinguishing phishing emails. The hybrid SMOTE and class weighting combination failed to outperform class weighting alone, with convergence of performance metrics suggesting algorithmic adjustment dominated synthetic data generation impact. This challenges assumptions that combining imbalance handling techniques yields additive benefits. Consistent performance across SMOTE sampling strategies (0.3 vs. 1.0) indicates oversampling degree may be less critical than approach choice itself. Undersampling for Naive Bayes showed minimal improvement despite perfect class balance, with precision remaining low (0.02), indicating performance issues stemmed from deeper algorithmic characteristics rather than class distribution alone. This suggests Naive Bayes struggles with complex, high-dimensional feature interactions where conditional independence assumptions may be violated. Polynomial interaction terms produced contrasting responses: modest Logistic Regression improvements versus severe Naive Bayes degradation (recall dropping from 90% to 4%). This decline resulted from conditional independence assumption violations, curse of dimensionality, and Min-Max scaling incompatibility with Naive Bayes's count-based feature preference. These findings emphasize that feature engineering must align with algorithmic assumptions rather than being applied universally.

CONCLUSION

This report was produced as part of a university course project and represents an exploratory study. The work has not been peer-reviewed and is shared for educational and reproducibility purposes. This study shows that while both Naïve Bayes and Logistic Regression algorithms can be adapted for email phishing detection, neither achieves satisfactory performance on highly imbalanced datasets. Several critical insights were revealed by the systematic comparative analysis while focusing on minority class detection under various preprocessing strategies. The research demonstrated that algorithm selection and preprocessing techniques must be carefully aligned to achieve effective phishing detection in imbalanced scenarios. Moreover, traditional accuracy metrics are inadequate for security applications where false negatives carry severe consequences, requiring recall-focused evaluation approaches. Notably, class weighting proved most beneficial for Logistic Regression's classification performance. Lastly, feature engineering techniques must align with algorithmic assumptions, as polynomial interactions that modestly improved Logistic Regression severely degraded Naive Bayes performance. These results challenge common assumptions about combining imbalance handling techniques and emphasize the importance of algorithm-specific optimization strategies.

REFERENCES

[1] P. Verma, A. Goyal, and Y. Gigras, "Email phishing: text classification using natural language processing," *Computer Science and Information Technologies*, vol. 1, no. 1, 2020, doi: 10.11591/csit.v1i1.p1-12.

[2] J. Hong, "The state of phishing attacks," *Commun ACM*, vol. 55, no. 1, pp. 74–81, Jan. 2012, doi: 10.1145/2063176.2063197.

[3] N. Marshall, D. Sturman, and J. C. Auton, "Exploring the evidence for email phishing training: A scoping review," *Comput Secur*, vol. 139, 2024, doi: 10.1016/j.cose.2023.103695.

[4] S. Furnell, "Phishing: can we spot the signs?," *Computer Fraud & Security*, vol. 2007, no. 3, pp. 10–15, Mar. 2007, doi: 10.1016/S1361-3723(07)70035-0.

[5] H. He and A. E. Garcia, "Learning from Imbalanced Data," *IEEE Trans Knowl Data Eng*, vol. 21, no. 9, pp. 1263–1284, Jun. 2009, doi: 10.1109/TKDE.2008.239.

[6] H. Han, W.-Y. Wang, and B.-H. Mao, "Borderline-SMOTE: A New Over-Sampling Method in Imbalanced Data Sets Learning," 2005. doi: https://doi.org/10.1007/11538059_91.

[7] A. Alhuzali, A. Alloqmani, M. Aljabri, and F. Alharbi, "In-Depth Analysis of Phishing Email Detection: Evaluating the Performance of Machine Learning and Deep Learning Models Across Multiple Datasets," *Applied Sciences (Switzerland)*, vol. 15, no. 6, Mar. 2025, doi: 10.3390/app15063396.

[8] R. Rahman and F. Fauzi Abdulloh, "Performance of Various Naïve Bayes Using GridSearch Approach In Phishing Email Dataset," *sinkron*, vol. 8, no. 4, 2023, doi: 10.33395/sinkron.v8i4.12958.

[9] A. Anggraina, R. Primartha, and A. Wijaya, "The combination of logistic regression and gradient boost tree for email spam detection," in *Journal of Physics: Conference Series*, 2019. doi: 10.1088/1742-6596/1196/1/012013.

[10] N. V Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," 2002. doi: https://doi.org/10.1613/jair.953.

[11] A. S. Hussein, T. Li, C. W. Yohannese, and K. Bashir, "A-SMOTE: A new preprocessing approach for highly imbalanced datasets by improving SMOTE," *International Journal of Computational Intelligence Systems*, vol. 12, no. 2, 2019, doi: 10.2991/ijcis.d.191114.002.

[12] 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 (Switzerland)*, vol. 14, no. 1, 2023, doi: 10.3390/info14010054.

[13] A. Dubey, F. Radenovic, and D. Mahajan, "Scalable Interpretability via Polynomials," 2022, doi: 10.48550/arXiv.2205.14108.

[14] G. Forman, "An Extensive Empirical Study of Feature Selection Metrics for Text Classification," 2003. Accessed: May 31, 2025. [Online]. Available: https://www.jmlr.org/papers/v3/forman03a.html

[15] N. Altwaijry, I. Al-Turaiki, R. Alotaibi, and F. Alakeel, "Advancing Phishing Email Detection: A Comparative Study of Deep Learning Models," *Sensors*, vol. 24, no. 7, Apr. 2024, doi: 10.3390/s24072077.

[16] B. K. Gontla, P. Gundu, P. J. Uppalapati, K. Narasimharao, and S. M. Hussain, "A Machine Learning Approach to Identify Phishing Websites: A Comparative Study of Classification Models and Ensemble Learning Techniques," EAI Endorsed Transactions on Scalable Information Systems, vol. 10, no. 5, 2023, doi: 10.4108/eetsis.vi.3300.

[17] S. Eftimie, R. Moinescu, and C. Racuciu, "Spear-Phishing Susceptibility Stemming From Personality Traits," IEEE Access, vol. 10, 2022, doi: 10.1109/ACCESS.2022.3190009.

[18] S. Uyun and E. Sulistyowati, "Feature selection for multiple water quality status: Integrated bootstrapping and SMOTE approach in imbalance classes," *International Journal of Electrical and Computer Engineering*, vol. 10, no. 4, 2020, doi: 10.11591/ijece.v10i4.pp4331-4339.

[19] F. E. Botchey, Z. Qin, K. Hughes-Lartey, and K. E. Ampomah, "Predicting Fraud in Mobile Money Transactions using Machine Learning: The Effects of Sampling Techniques on the Imbalanced Dataset," Informatica (Slovenia), vol. 45, no. 7, 2021, doi: 10.31449/inf.v45i7.3179.

[20] X. Wang *et al.*, "Machine learning-enabled risk prediction of chronic obstructive pulmonary disease with unbalanced data," *Comput Methods Programs Biomed*, vol. 230, 2023, doi: 10.1016/j.cmpb.2023.107340.

[21] D. Thakur, T. Gera, V. Bhardwaj, A. A. AlZubi, F. Ali, and J. Singh, "An enhanced diabetes prediction amidst COVID-19 using ensemble models," *Front Public Health*, vol. 11, 2023, doi: 10.3389/fpubh.2023.1331517.

[22] E. Cratchley, "Email Phishing Dataset," 2025. Accessed: Jun. 01, 2025. [Online]. Available: https://www.kaggle.com/datasets/ethancratchley/email-phishing-dataset