International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Enhancing Predictive Accuracy in Phishing Attack Detection: A Study on the Impact of Collinearity and Feature Selection in ML-based Logistic Regression Models

Sagar Aghera*[1], Nikhil Yogesh Joshi[2]

Submitted:13/03/2024 **Revised**: 28/04/2024 **Accepted**: 05/05/2024

Abstract: Phishing threats present dangers, for people and businesses alike emphasizing the need, for creating reliable detection techniques. It is crucial to establish phishing tactics to protect confidential data and avoid monetary damages. This study delves deeper into the intricacies of logistic regression models and how these models could effectively detect phishing attacks with a focus on impact of factors like collinearity and feature selection on predictive accuracy and model performance. In addition to logistic regression, different machine learning models, such as Decision Tree Classifier, Gaussian Naive Bayes, Logistic Regression, K Nearest Neighbors and Linear Discriminant Analysis were also considered to analyze the relationships between predictor variables and successful phishing attack likelihood and the predictive accuracy from each of the methods. By conducting experiments and comparisons we show that addressing collinearity issues and employing feature selection techniques significantly improve the predictive accuracy of logistic regression models compared to other common machine learning models. Through a methodical process of feature engineering focused on addressing collinearity among predictors, we achieved a substantial reduction of over 35% in the false negative rate for the logistic regression model which is crucial as false negatives are more costly. These findings provide insights, for enhancing the efficiency of phishing detection systems to strengthen cybersecurity defenses against emerging threats.

Keywords: *Phishing URL, Machine Learning, Logistic Regression, Collinearity.*

1. Introduction

The rapid expansion of the internet has transformed communication and connectivity, providing unprecedented convenience and opportunities for individuals and businesses globally. However, this digital realm is not immune to malicious actors seeking to exploit vulnerabilities for harmful purposes. One of the most prevalent cyber threats is phishing, a deceptive tactic where attackers impersonate legitimate or seemingly similar sounding entities to trick unsuspecting users into disclosing sensitive information like login credentials, financial details, or personal data. Phishing attacks frequently employ fraudulent URLs (Uniform Resource Locators) that mimic authentic websites, posing a significant challenge for cybersecurity professionals responsible for protecting users and organizations.

Detecting and mitigating phishing URLs are critical components of cybersecurity efforts, necessitating innovative techniques and robust methodologies to stay ahead of evolving threats. Traditional approaches to URL detection often rely on rule-based systems or blacklisting known malicious domains. In the current world scenario, phishing attackers continually adapt their tactics by using dynamic methods that allow them to circumvent blacklist-based detection approaches with relative ease. Conversely, machine learning (ML) techniques have emerged as a promising approach to combat the evolving tactics of phishing attackers. These methods leverage algorithms that analyse various features extracted from websites to determine their legitimacy. Numerous machine learning models have been employed and documented in the detection of phishing websites, demonstrating promising results and effective performance. In addition, machine learning algorithms have capability to change its behaviour by learning about circumventing techniques adapted by the bad actors.

Among the various techniques employed for phishing attack detection, logistic regression models have emerged as a promising avenue, offering a data-driven approach to distinguishing between benign and malicious URLs based on their features and characteristics. For detecting phishing attacks, one promising approach is using logistic regression models. These models analyse the features and characteristics of URLs to determine if they are safe or malicious.

Like other regression models, logistic regression can suffer from due to Collinearity [refer something]. Collinearity occurs when two or more predictor features are highly correlated, in the sense they contain similar information about the variance in outcome feature. This can lead to problems like unstable coefficient estimates, reduced model

[1] *Netskope Inc, – 30040, USA*
ORCID ID : 0009-0007-5561-7250
[2] *Fiserv Inc., – 30041, USA*
ORCID ID : 0009-0002-3868-9571
** Corresponding Author Email: sagaraghera@ieee.org*

interpretability, and inflated standard errors. One of the methods to identify collinearity in features is to use Correlation Matrix which identifies highly correlated pairs of features.

In our study, we found few features were collinear based on 80% threshold with Correlation Matrix. These Collinear features were studied and analysed to understand its impact on the accuracy of the models. We conducted experiments to assess the impact of dropping collinear features individually and in batch. The results of both individual and batch feature removal were methodically documented and analysed.

The goal is to understand how to make logistic regression models are more accurate and efficient at catching phishing attacks. Our findings promise to offer valuable insights and guidance for the development of more effective and efficient phishing detection systems, thereby bolstering cybersecurity defences against the ever-evolving phishing attacks.

By carefully studying these factors, this research aims to help develop better and more reliable ways to detect phishing attacks. The end goal is to make the digital world safer and more secure for everyone.

2. Background & Related Works

Numerous approaches and remedies have been suggested and crafted by researchers to tackle phishing assaults. Given the dynamic nature of these attacks, strategies rooted in education, legal frameworks, or technical interventions have emerged as feasible solutions. This segment provides an exhaustive examination of prevailing machine learning-based solutions devised for countering phishing assaults.

In their study, Adeyemo et al. (2021) propose a novel ensemble approach that combines logistic model trees trained on different subsets of features. This ensemble strategy aims to diversify the base classifiers while leveraging their complementary strengths, ultimately enhancing the overall performance of the phishing detection system. With this strategy they (Adeyemo et al. (2021)) were able to achieve accuracy of 99.6%.

The study by Moedjahedy et al. (2022) demonstrates the efficacy of CCrFS in enhancing phishing detection performance through empirical evaluation using real-world datasets. In Moedjahedy et al. 's study (2022), an accuracy of 97.06% was attained for dataset 1 using 10 features, while for dataset 2, the accuracy reached 95.88% with the same number of features.

Vajrobol, Gupta, and Gaurav (2024) propose a novel approach to phishing URL detection by leveraging mutual information-based logistic regression. Mutual information is a measure of the statistical dependence between two random variables, making it well-suited for capturing the relationships between various features and the likelihood of a URL being malicious.With this approach, Vajrobol, Gupta, and Gaurav (2024) were able to get accuracy of 99.97%

Chiramdasu et al. (2021) propose a logistic regression-based approach for malicious URL detection, leveraging features extracted from the URLs themselves, such as domain reputation, URL length, and presence of suspicious keywords. By training a logistic regression model on a labeled dataset of benign and malicious URLs, the authors aim to develop an effective and efficient detection system capable of accurately identifying malicious URLs in real-time. Chiramdasu et al. (2021)'s study finally evaluates the performance of the system by considering various well-known metrics such as Accuracy, Precision, Recall, False-Positive rate and True-Positive rate and concluded with Logistic regression model being most efficient.

Sarma et al. (2021) conduct a comprehensive comparative analysis of machine learning algorithms for phishing website detection, including decision trees, support vector machines, random forests, and neural networks. By evaluating these algorithms on a diverse set of features extracted from phishing and legitimate websites, the authors aim to assess their effectiveness in accurately distinguishing between benign and malicious URLs. Through empirical evaluation and comparative analysis, the authors identify the strengths and limitations of each algorithm in terms of detection accuracy, false positive rate, and computational efficiency. In their study, random forest classifier had achieved the most efficient and highest performance scoring with 98% accuracy.

Abedin et al. (2020) investigate the use of machine learning classification techniques for phishing attack detection, including decision trees, support vector machines, random forests, and neural networks. By evaluating these techniques on a diverse dataset of phishing and legitimate emails, the authors aim to assess their effectiveness in accurately distinguishing between benign and malicious messages. Among these three classifiers, random forest performance is the highest with a precision of 97%.

These studies collectively underscore the efficacy of machine learning techniques in combating phishing attacks, with each approach offering valuable insights and advancements towards enhancing cybersecurity measures. Majority of the available literature do not address Collinearity and interpretability of machine learning model. In our novel approach, we deal with Collinearity, simplicity and interpretability on logistic regression model.

3. Methodology

3.1. Dataset

The dataset utilized in this study, derived from the PhiUSIIL

framework developed by Prasad and Chandra (2023), encompasses a diverse array of features extracted from URLs. These features include both lexical characteristics (such as URL length and the presence of suspicious keywords) and domain-specific attributes (such as domain reputation and age). The dataset is designed to facilitate a comprehensive analysis of phishing URLs, enabling the evaluation of various detection methodologies. There are no nulls onr duplicate values in the dataset.

The PhiUSIIL Phishing URL Dataset is a comprehensive dataset consisting of 134,850 legitimate URLs and 100,945 phishing URLs. The majority of the URLs included in this dataset are recent, ensuring relevance and up-to-date analysis. Features for this dataset are extracted from both the source code of the webpages and the URLs themselves. Notable features such as CharContinuationRate, URLTitleMatchScore, URLCharProb, and TLDLegitimateProb are derived from these existing data points, providing a robust foundation for phishing detection research.

Our research will employ this dataset to evaluate logistic regression against other machine learning algorithms, and then investigate the effectiveness of logistic regression models in phishing detection, particularly focusing on the role of collinearity and feature selection. By systematically addressing these factors, we aim to enhance the predictive accuracy of logistic regression models and compare their performance with other machine learning techniques. This approach will allow us to identify the most effective strategies for phishing URL detection, contributing to the ongoing efforts to fortify cybersecurity defenses against this pervasive threat.

3.2. Data Preprocessing

The data needed some preprocessing to ensure it was suitable for training and testing the machine learning models. Initially, the dataset included four categorical features: 'URL,' 'Domain,' 'TLD,' and 'Title.' Upon transforming these categorical features into numerical formats, it became evident that they did not contribute valuable information to the model. This could be due to a variety of reasons such as lack of variance, or irrelevance to the target variable. Consequently, these four features were excluded from the dataset.

After this refinement process, the dataset originally comprising 54 features was further analyzed for relevance and utility. A thorough feature selection process was undertaken, which involved evaluating the significance of each feature through correlation analysis and their impact of the predictive accuracy. As a result of this selection process, 35 features were identified as the most relevant and useful for the machine learning models. These selected features were then used for the final training and testing phases, ensuring that the models were built on the informative and effective subset of the original data.

3.3. ML Models

For this study, we used Decision Tree Classifier, Gaussian Naive Bayes, Logistic Regression, K Nearest Neighbors and Linear Discriminant Analysis to analyze the relationships between predictor variables and successful phishing attack prediction. The dataset was split into train data and test data with test size being 33%. This resulted into 157982 URLs in training data and 77813 URLs in test data. To make sure the dataset is randomly divided for training and testing was done using scikit-learn multiple times to assess the fact that the split is independent of the final scores. This approach guaranteed that the data split was both random and could be reproduced. The machine learning models were trained and validated using scikit-learn, a popular Python library for machine learning. For determining the best performing model, we used accuracy score and confusion matrix as benchmarking parameter(s). When it comes to spotting phishing URLs it's crucial to watch out for False negatives, which occur when phishing URLs are wrongly identified as non-phishing ones. This can be harmful to any firm as it can lead to data theft, identity theft, malware infections, and ultimately compromising organizations or individual identity. Among these five ML models, Logistic regression performance is the highest with an accuracy of 99.56% and least number of False negatives (88), as seen in **Table 1**.

Table 1. Confusion Matrix for Logistic regression model

Model	TP	FP	TN	FN	Accuracy
Logistic Regression	**33138**	**247**	**44340**	**88**	**99.56%**
Gaussian Naïve Bayes	17371	16014	44215	213	79.14%
Decision Tree Classifier	32947	438	44028	400	98.92%
K-Nearest Neighbors	33098	287	44201	227	99.33%
Linear Discriminant Analysis	32557	828	43534	894	97.78%

3.4. Addressing Collinearity

With Logistic regression being best performing model when compared to other models - Decision Tree Classifier, Gaussian Naive Bayes, K Nearest Neighbors and Linear Discriminant Analysis, we took a step further and did feature engineering with respect to Collinearity. For identifying collinearity in features, we used Correlation

Matrix which identifies highly correlated pairs of features. We calculated the correlation matrix to collinearity in our dataset. This matrix gauges the connection, between feature pairs with correlation values ranging from -1 to 1. A value near 1 signifies a linear relationship while a value near -1 indicates a strong negative linear relationship. Values close to 0 indicate minimal to no linear relationship.

To spot features we set a correlation threshold of 0.80 (80%). This threshold was selected based on the practice of correlations above 0.80 as a sign of multicollinearity, which can impact the performance and interpretability of machine learning models.

From the results of the correlation matrix, we identified six features with correlation values exceeding the 0.80 threshold. **Fig. 1** shows the Correlation Heatmap of the identified six highly collinear features.

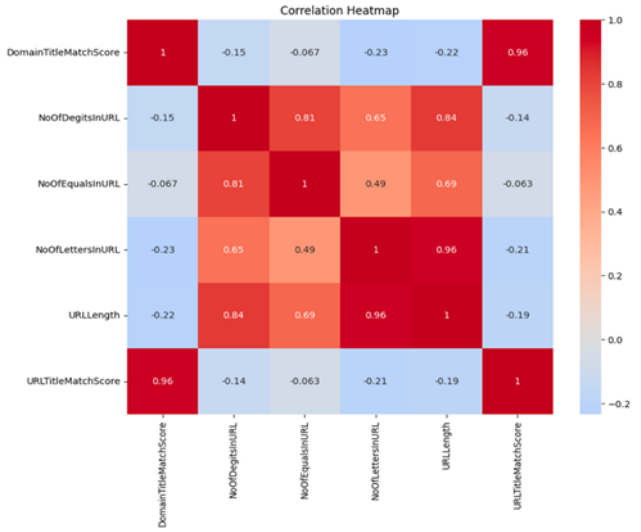

Fig. 1 Correlation Heatmap of highly collinear features.

Recognizing and dealing with collinear features is essential, for enhancing both model performance and interpretability. We experimented with dropping collinear features both individually and in combination while calculating the performance numbers in form Accuracy and ROC-AUC score. In our hypothesis, collinearity may or may not be bad [reference]. A detailed analysis related to this hypothesis is demonstrated and discuss in following section.

4. Results & Discussions

This section shows the results of our tests for detecting phishing URLs. We used five different methods: Decision Tree Classifier, Gaussian Naive Bayes, Logistic Regression, K-Nearest Neighbors, and Linear Discriminant Analysis. We selected logistic regression as our best-performing model based on its accuracy score and the number of false negatives. We then continued to optimize the machine learning model further by taking into consideration effects of feature collinearity. We took into account the issue of collinearity (when predictor variables are highly correlated) and experimented and analyzed the effect of dropping individual collinear features, in combinations and dropping all collinear features together.

From the Correlation Matrix, the identified collinear features with 0.8 or above threshold are highlighted in Table 2. The performance measures for each of these combinations are analyzed and demonstrate using accuracy and ROC-AUC score in following sub sections.

Table 2. Collinear features list

Feature Name	Correlation value
DomainTitleMatchScore	0.961
URLTitleMatchScore	0.961
URLlength	0.956
NoOfLettersInURL	0.956
NoOfEqualsInURL	0.83
NoOfDegitsInURL	0.81

4.1. Individual dropping

Firstly, we evaluated the impact of removing each collinear feature individually. This approach allowed us to observe the specific contribution of each feature to the model's performance. For each iteration, a single collinear feature was removed from the dataset and we computed ROC-AUC score & Accuracy scores with each iteration and compared with the original results we obtained when none of the features were dropped. Collected results are displayed in Table 3. As per the results obtained, we validated the Ben-Farag and El-Saeiti (n.d.) findings – when there is no class imbalance problem (class balance) the effect of multicollinearity becomes less severe. Conversely, in our results, we saw slightly degraded performance when dropping collinear features individually making original Logistic regression model without dropping any features the most efficient one when using Accuracy score and ROC-AUC score as benchmark metric.

Table 3. Accuracy & ROC-AUC score for individual collinear feature dropping vs original result

Dropped collinear feature	Accuracy score	ROC-AUC score
Without dropping any	**0.9957**	**0.9953**
DomainTitleMatchScore	0.9951	0.9947
URLTitleMatchScore	0.9939	0.9935
URLlength	0.9824	0.9825
NoOfDegitsInURL	0.9766	0.9767
NoOfEqualsInURL	0.9954	0.9953
NoOfLettersInURL	0.9768	0.9766

4.2. Batch dropping

In next iteration, All identified collinear features (those with correlation values above the 0.80 threshold) were removed simultaneously and then in pairs. The logistic regression model was re-trained with the remaining features for each iteration of batching. Performance of each iteration were gauged based on Accuracy & ROC-AUC score and the results of this experiments are highlighted in Table 4. When dropping all of the collinear features, we observed 2% degradation in Accuracy & ROC-AUC score. Only time, the slight improvement in Accuracy & ROC-AUC score was seen for a case when both URLTitleMatchScore & DomainTitleMatchScore were dropped together. Both had very high Correlation value of 0.961 and are highly correlated with each other. Rest of the combination had degradation in model performance as noted in Table 4. Midi, H., Sarkar, S. K., & Rana, S. (2010) & Alin (2010)'s approach says that for moderate to large sample sizes, the approach to drop one of the correlated variables help to reduce multicollinearity and improve performance. In our study, by dropping just one feature didn't help as concluded from previous sub-section but dropping batch of highly correlated help in reducing multicollinearity and improved performance was also achieved.

Table 4. Accuracy & ROC-AUC score for batch dropping

Dropped collinear feature	Accuracy score	ROC-AUC score
Without dropping any	**0.9957**	**0.9953**
Dropping all	0.9732	0.9733
URLTitleMatchScore & DomainTitleMatchScore	**0.9960**	**0.9959**
NoOfLettersInURL & URLLength	0.9763	0.9762
NoOfDegitsInURL & NoOfEqualsInURL	0.9735	0.9735

We did calculate confusion matrix for the two selected models (base model without dropping any feature and after dropping URLTitleMatchScore & DomainTitleMatchScore) from Table 4. The results, as described in Table 5, presented the fact that after dropping of URLTitleMatchScore & DomainTitleMatchScore helped in reducing the False negatives count by more than 35% which is quite impressive. This indicates that we are able to improve the misclassification of phishing URLs as non-phishing URLs by more than 35% (88 → 55) after doing feature engineering with respect to feature collinearity.

Table 5. Confusion matrix for stock Logistic regression model and after dropping of two highly collinear features.

Model	TP	FP	TN	FN
Base Logistic Regression	33138	247	44340	88
Logistic regression after dropping URLTitleMatchScore & DomainTitleMatchScore	33132	253	44373	**55**

5. Conclusion

In this study, we evaluated the effectiveness of various machine learning algorithms for detecting phishing URLs. Our analysis included Decision Tree Classifier, Gaussian Naive Bayes, Logistic Regression, K-Nearest Neighbors, and Linear Discriminant Analysis. Through rigorous testing and evaluation, we identified Logistic Regression as the best-performing model, particularly when considering accuracy and false negative rates. This conclusion guided our subsequent optimization efforts focused on addressing feature collinearity.

Our study systematically evaluated the impact of removing collinear features on the performance of a logistic regression model used for classifying URLs. By examining the effects of individual and batch removals of collinear features, we were able to gain insights into the relationship between multicollinearity and model performance.

Our findings validate the conclusions of Ben-Farag and El-Saeiti, demonstrating that in the absence of class imbalance, multicollinearity exerts a less pronounced effect on model performance. The initial analysis, which involved the individual removal of collinear features, revealed that the model without any feature removal outperformed all other variations in terms of Accuracy and ROC-AUC scores.

However, when we advanced to batch removal of highly collinear features, we observed subtle outcomes. Specifically, while most combinations of feature removals led to degraded model performance, the simultaneous removal of URLTitleMatchScore and DomainTitleMatchScore, which had very high correlation of 0.961, resulted in a slight improvement in both Accuracy and ROC-AUC scores. This finding aligns with the strategies suggested by Midi, Sarkar, and Rana (2010) and Alin (2010), indicating that removing batches of highly correlated features can indeed ease multicollinearity and enhance model performance.

The analysis of confusion matrices further substantiated these results. Notably, dropping the highly correlated URLTitleMatchScore and DomainTitleMatchScore features significantly reduced the number of false negatives

by over 35%, thereby improving the classification of phishing URLs. This improvement highlights the effectiveness of targeted feature engineering in addressing multicollinearity and enhancing model robustness.

In summary, while individual removal of collinear features did not produce performance gains, strategic batch removals, especially of highly correlated features, can lead to meaningful improvements. These insights provide a valuable framework for handling multicollinearity in logistic regression models, ultimately contributing to more accurate and reliable predictive performance leading to effective phishing detection systems.

6. Acknowledgements

We extend our gratitude to Shantanu Neema, Senior Data Scientist, for his invaluable guidance throughout the machine learning modeling process., and Prasad,Arvind and Chandra,Shalini. (2024) for providing the pre-processed Phishing URL dataset.

Author contributions

Sagar Aghera: Conceptualization, Methodology, Software, Finalizing draft **Nikhil Yogesh Joshi:** Data curation, Draft preparation, Model validation, Review

Conflicts of interest

The authors declare no conflicts of interest.

References

[1] Adeyemo, V.E., Balogun, A.O., Mojeed, H.A., Akande, N.O., Adewole, K.S. (2021). Ensemble-Based Logistic Model Trees for Website Phishing Detection. In: Anbar, M., Abdullah, N., Manickam, S. (eds) Advances in Cyber Security. ACeS 2020. Communications in Computer and Information Science, vol 1347. Springer, Singapore.

[2] Moedjahedy, J., Setyanto, A., Alarfaj, F. K., & Alreshoodi, M. (2022). CCrFS: Combine Correlation Features Selection for Detecting Phishing Websites Using Machine Learning. Future Internet, 14(8), 229.

[3] Vajrobol, V., Gupta, B. B., & Gaurav, A. (2024). Mutual information based logistic regression for phishing URL detection. Cyber Security and Applications, 2, 100044.

[4] Chiramdasu, R., Srivastava, G., Bhattacharya, S., Reddy, P. K., & Gadekallu, T. R. (2021, August). Malicious url detection using logistic regression. In 2021 IEEE International Conference on Omni-Layer Intelligent Systems (COINS) (pp. 1-6). IEEE.

[5] Sarma, D., Mittra, T., Bawm, R. M., Sarwar, T., Lima, F. F., & Hossain, S. (2021). Comparative analysis of machine learning algorithms for phishing website detection. In Inventive Computation and Information Technologies: Proceedings of ICICIT 2020 (pp. 883-896). Springer Singapore.

[6] Abedin, N. F., Bawm, R., Sarwar, T., Saifuddin, M., Rahman, M. A., & Hossain, S. (2020, December). Phishing attack detection using machine learning classification techniques. In 2020 3rd International Conference on Intelligent Sustainable Systems (ICISS) (pp. 1125-1130). IEEE.

[7] Prasad,Arvind and Chandra,Shalini. (2024). PhiUSIIL Phishing URL (Website). UCI Machine Learning Repository. https://doi.org/10.1016/j.cose.2023.103545.

[8] Midi, H., Sarkar, S. K., & Rana, S. (2010). Collinearity diagnostics of binary logistic regression model. Journal of Interdisciplinary Mathematics, 13(3), 253–267.

[9] Ben-Farag, S. O., & El-Saeiti, I. N. (2022) Effect and Influence of Class Imbalance and Multicollinearity in Binary Logistic Regression (A Comparative Simulation Study).

[10] Alin, A. (2010). Multicollinearity. Wiley interdisciplinary reviews: computational statistics, 2(3), 370-374.