Final Project Update – CIS 6670

Papa Kobina Orleans-Bosomtwe

Phishing URL Detection Using Machine-Learning Techniques

INTRODUCTION

Phishing is a form of cybercrime used by malicious attackers to obtain confidential information from users by creating a counterfeit website that mimics a legitimate website [1]. Machine learning, which has grown in popularity recently, uses computational methods to improve performance or make accurate predictions by incorporating experience [2]. This project aims to solve a rise in phishing websites by using machine learning techniques to detect phishing websites. By answering two crucial research questions, it is believed that machine-learning methods can help detect phishing websites. These questions are:

1. How can feature selection enhance the accuracy of Naïve Bayes and Random Forest in phishing detection?
2. What are the strengths and weaknesses of the algorithms above in detecting phishing websites regarding accuracy?

The author hopes that by answering these questions, the following contributions can be made to the efforts of phishing detection:

1. A robust phishing website detection model with high accuracy will be created using Naïve Bayes and Random Forest algorithms.
2. A comparative analysis of the Naïve Bayes and Random Forest algorithms in phishing website detection will be provided.

LITERATURE REVIEW

The paper "Anti-phishing Based on Automated Individual White-List" by Cao et al. [3] presents a novel method for combating phishing and pharming attacks through the use of Automated Individual White-Lists (AIWL). Unlike traditional blacklists, which struggle to keep up with the rapidly changing nature of phishing sites, AIWL maintains a personalized whitelist of familiar Login User Interfaces (LUIs) for each user. By employing a Naïve Bayesian classifier to manage the whitelist automatically, the system effectively alerts users when submitting sensitive information to unfamiliar sites, thus providing a robust defense against phishing and pharming attacks. The paper demonstrates that this approach significantly enhances online security by leveraging the stability of LUI features such as IP addresses and DOM paths, resulting in a low false warning rate.

The paper "Heuristic Nonlinear Regression Strategy for Detecting Phishing Websites" by Babagoli et al. [4] proposes a novel approach to phishing detection using a meta-heuristic-based nonlinear regression algorithm and feature selection techniques. The researchers utilize a dataset of 11,055 phishing and legitimate webpages, extracting 20 features through decision tree and wrapper methods, achieving a detection accuracy of up to 96.32%. The study employs harmony search (HS) and support vector machine (SVM) algorithms for prediction, with the nonlinear regression model optimized via HS achieving accuracy rates of 94.13% and 92.80% for training and testing, respectively. The findings highlight the superior performance of the nonlinear regression-based HS method compared to SVM in detecting phishing websites.

METHODOLOGY

This section will briefly discuss the chosen algorithms, performance metrics, dataset, and data preprocessing steps, including the extracted features.

Chosen Algorithms

Naïve Bayes algorithm – This classification algorithm applies Bayes' Theorem with the assumption of independence between features [5].

Random Forest algorithm – This machine learning algorithm constructs multiple decision trees, with each tree depending on the values of a random vector sampled independently [6].

Performance Metrics

1. Accuracy – Overall effectiveness of a classifier [7].

2. Precision – Class agreement of the data labels with the positive labels given by the classifier [7].

3. Recall – the proportion of Real Positive cases that are correctly Predicted Positive [8].

4. F1-Score – The weighted average of Precision and Recall. Typically favors algorithms with higher sensitivity [9].

5. Macro Average – Useful for understanding the model's performance without being influenced by class imbalance.

6. Weighted Average – Useful for understanding model performance in a way that accounts for class imbalance.

Dataset

The dataset is a balanced dataset that ensures both classes (phishing and benign) are equally represented. An imbalanced dataset could cause bias to the majority class [10].

Data Preprocessing: Extracted Features

Feature Selection efficiently and effectively prepares data for machine learning problems [11].

URL Length [12]

Count of dots

Count of hyphens

Presence of HTTPS

Length of domain

Path length

Query length

Domain age

RESULTS

Naive Bayes Model Performance

The Naive Bayes model achieved an overall accuracy of 78.6% [Figure 1]. This metric indicates that the model correctly classifies approximately 79% of the URLs in the test dataset. The model's performance can be further analyzed by examining each class's precision, recall, and F1-score.

- **Class 0 (Legitimate URLs)**: The precision is 0.72, meaning that when the model predicts a URL as legitimate, it is correct 72% of the time. The recall is notably high at 0.95, indicating that the model successfully identifies 95% of all legitimate URLs in the dataset. The F1-score, which balances precision and recall, is 0.82, reflecting the model's overall effectiveness for this class.
- **Class 1 (Phishing URLs)**: The precision is 0.92, suggesting that the model is highly reliable in identifying phishing URLs, with

only a small percentage of false positives. However, the recall drops to 0.63, indicating that the model misses 37% of the phishing URLs. The F1-score for phishing URLs is 0.74, highlighting a need for improvement in recall.

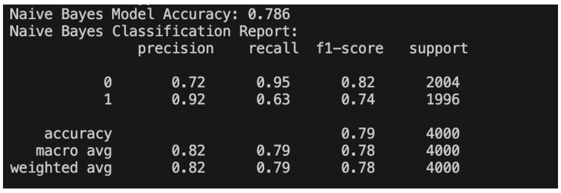
Overall, the Naive Bayes model is biased towards classifying URLs as legitimate, evident in its higher recall for legitimate URLs. While it excels in avoiding false positives for phishing URLs, its ability to detect all phishing attempts could be enhanced.

Random Forest Model Performance

The Random Forest model demonstrates a superior performance with an accuracy of 88.6% [Figure 2]. This indicates a significant improvement over the Naive Bayes model, with a correct classification rate of nearly 89%.

- **Class 0 (Legitimate URLs)**: The precision is 0.87, and the recall is 0.91, leading to an F1-score of 0.89. This balance between precision and recall suggests that the Random Forest model effectively identifies legitimate URLs while minimizing false positives.
- **Class 1 (Phishing URLs)**: The precision is 0.90, demonstrating the model's capability to identify phishing URLs accurately. The recall is 0.86, indicating that the model successfully detects 86% of the phishing URLs. The F1-score of 0.88 reflects the model's balanced performance in identifying phishing attempts.

The Random Forest model's higher accuracy and balanced precision and recall for both classes demonstrate its robustness in handling the phishing detection task. Its ability to reduce false negatives for phishing URLs offers a comprehensive security measure against phishing threats.

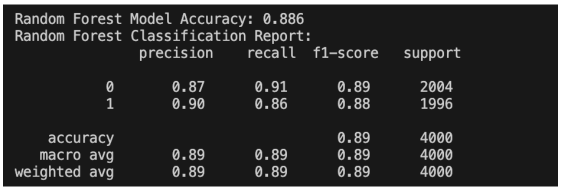


```
Naive Bayes Model Accuracy: 0.786
Naive Bayes Classification Report:
              precision    recall  f1-score   support

           0       0.72      0.95      0.82      2004
           1       0.92      0.63      0.74      1996

    accuracy                           0.79      4000
   macro avg       0.82      0.79      0.78      4000
weighted avg       0.82      0.79      0.78      4000
```

Figure 1: Naïve Bayes Results



```
Random Forest Model Accuracy: 0.886
Random Forest Classification Report:
              precision    recall  f1-score   support

           0       0.87      0.91      0.89      2004
           1       0.90      0.86      0.88      1996

    accuracy                           0.89      4000
   macro avg       0.89      0.89      0.89      4000
weighted avg       0.89      0.89      0.89      4000
```

Figure 2: Random Forest Results

DISCUSSION

Both models have the potential to benefit from enhanced feature selection, with Naïve Bayes possibly experiencing more significant improvements due to its reliance on feature independence. Random Forest generally outperforms Naïve Bayes in this context, providing higher accuracy and balanced metrics, albeit at the expense of increased computational complexity. In contrast, Naïve Bayes is more straightforward and faster but struggles with recall for phishing sites. Despite this challenge, Naïve Bayes excels in precision when it does make a positive prediction.

CONCLUSION AND FUTURE WORK

The comparison between the Naive Bayes and Random Forest models highlights the latter's superior ability to classify URLs as legitimate or phishing accurately. The Random Forest model's ensemble approach, which leverages multiple decision trees, provides greater flexibility and accuracy in capturing complex patterns within the data. While the Naive Bayes model offers a simple and computationally efficient solution, its tendency to misclassify phishing URLs suggests the need for additional features or alternative techniques to enhance recall. In contrast, the Random Forest model's strong performance across all metrics makes it a more reliable choice for real-world phishing detection applications. These results underscore the importance of selecting appropriate models for cybersecurity tasks, where the cost of false negatives can be significant. The Random Forest model's balanced and robust performance offers a promising approach to enhancing online security and protecting users from phishing threats. To further validate the robustness of these models, they can be tested on more extensive and diverse datasets. Future research could also focus on optimizing feature selection to improve model accuracy.

REFERENCES

[1] Shahrivari, V., Darabi, M. M., & Izadi, M. (2020). *Phishing Detection Using Machine Learning Techniques* (arXiv:2009.11116). arXiv. http://arxiv.org/abs/2009.11116

[2] Mohri, M., Rostamizadeh, A., & Talwalkar, A. (2018). *Foundations of Machine Learning, second edition*. MIT Press.

[3] Cao, Y., Han, W., & Le, Y. (2008). Anti-phishing based on automated individual white-list.

Proceedings of the 4th ACM Workshop on Digital Identity Management, 51–60. https://doi.org/10.1145/1456424.1456434

[4] Babagoli, M., Aghababa, M. P., & Solouk, V. (2019). Heuristic nonlinear regression strategy for detecting phishing websites. *Soft Computing*, *23*(12), 4315–4327. https://doi.org/10.1007/s00500-018-3084-2

[5] Rish, I. (2001). An Empirical Study of the Naïve Bayes Classifier. *IJCAI 2001 Work Empir Methods Artif Intell*, *3*.

[6] Breiman, L. (2001). Random Forests. *Machine Learning*, *45*(1), 5–32. https://doi.org/10.1023/A:1010933404324

[7] Sokolova, M., & Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, *45*, 427–437. https://doi.org/10.1016/j.ipm.2009.03.002

[8] Powers, D. M. W. (2020). *Evaluation: From precision, recall and F-measure to ROC, informedness, markedness and correlation* (arXiv:2010.16061). arXiv. https://doi.org/10.48550/arXiv.2010.16061

[9] Sokolova, M., Japkowicz, N., & Szpakowicz, S. (2006). Beyond Accuracy, F-Score and ROC: A Family of Discriminant Measures for Performance Evaluation. In A. Sattar & B. Kang (Eds.), *AI 2006: Advances in Artificial Intelligence* (pp. 1015–1021). Springer. https://doi.org/10.1007/11941439_114

[10] Guo, H., & Viktor, H. L. (n.d.). *Learning from Imbalanced Data Sets with Boosting and Data Generation: The DataBoost-IM Approach*.

[11] Li, J., Cheng, K., Wang, S., Morstatter, F., Trevino, R. P., Tang, J., & Liu, H. (2017). Feature Selection: A Data Perspective. *ACM Comput. Surv.*, *50*(6), 94:1-94:45. https://doi.org/10.1145/3136625

[12] Ahammad, S. H., Kale, S. D., Upadhye, G. D., Pande, S. D., Babu, E. V., Dhumane, A. V., & Bahadur, Mr. D. K. J. (2022). Phishing URL detection using machine learning methods. *Advances in Engineering Software*, *173*, 103288.

https://doi.org/10.1016/j.advengsoft.2022.103288