Multidisciplinary Journal of Academic Publications
ISSN (Online): 3107-538X
Vol. 01, Issue 01, September 2025

ML TECHNIQUES FOR PHISHING DETECTION

Pooja Singh[1], Hrishita Maurya[2]

Assistant Professor, Department of Aeronautical Engineering, SSSUTMS, Sehore[1]

Research Scholar, Department of Aeronautical Engineering, SSSUTMS, Sehore[2]

Abstract

This review focuses on the application of AI tools to detect phishing sites, concentrating specifically in calculations such as Arbitrary Backwoods (LR), XGBoost, Simplistic Bayes (NB), Support Vector Machine(Bolstered vector machine). The study reminiscences the importance of comprehensive data preprocessing which consists, different precursory procedures like cleaning, extraction and standardization to improve the predictive accuracy and robustness of models. Among the tried methodologies, Angle Helping accomplished 97.6% precision which shows its strength in phishing identification and Guileless Bayes recorded the least exactness at 60.5%. Results show the fundamental role of selecting appropriate AI models and preprocessing methods to enhance phishing detection systems. This represents a significant advancement over traditional methods, especially in dealing with happy hour phishing attacks and managing restrictive data sets, providing an effective solution to enhance cyber security.

Keywords: *Phishing identification[1], AI, Inclination Helping[2], Arbitrary Backwoods[3], Network safety[4], Information preprocessing[5].*

1. Introduction

With the highest number of reported phishing attacks it poses a significant threat to online security, hooking unsuspecting users through fraudulent websites that mimic real ones. Traditional identification methods such as blacklist based systems often fall short in identifying newly arising or sophisticated phishing campaigns — especially happy hour attacks. As the complexity of these threats increases, a need for more advanced detection methods has risen. This examination examines the utilization of AI(ML) techniques for email phishing recognition and assesses the viability of various calculations, for example, Arbitrary Woods, XGBoost (XGB), Simple Bayes (NB), Supporting Vector Machine,Helping Iinclude+ Tree Classifiefr vein Socre Boost. Highlighting the core job of data pre-processing from extraction, cleaning to standardization in a stepwise manner this study aims at enhancing accuracy and reliability phishing detection models. Of the attempted models, Slope Supporting rose as second to none with a precision of 97.6%, while Naïve Bayes shown minimal presentation at only six percent three four in point five eight metrics indicating that it had nothing but luck in its prediction success or failure(forty nine basis points below abase line). These findings underscore the importance of selecting appropriate ML models and preprocessing methods, marking a significant advancement in combatting phishing attacks to bolster online security practices.

2. Literature Review

Almseidin et al. (2019) investigated several AI processes and component selection strategies that can be applied to increase spam detection for phishing. Through their review, the creators have shown that merging AI calculations and aspect selection can significantly increase detection accuracy.

Do et al. (2022) The profound learning systems for Phishing Detection are characterized by (Safaei et al., 2022) consciousness comprehensive scientific species. Their audit information the persistent difficulties and outlines future headings that highlight a need for more up to date modern strategies in managing developing phishing

systems. By clearly indicating classification performance of the study in this sense, such work starts to reveal those possibilities that deep learning can use traditional methods for phishing detection.

Jain and Gupta (2022) provide a comprehensive study on tactics by which phishing assault are achieved, defense mechanisms, also review of research issues. In their survey they cover the evolution of phishing tactics and security tools, highlighting an evergreen demand for new ways to deal with emerging threats.

Gandotra and Gupta, 2021. referred to this conversation by introducing their work on an effective AI oriented method for phishing detection. They suggest a theory which takes advantage of evolved AI algorithms to enhance detection accuracy, reflecting the trend toward use of AI methods in anti-phishing efforts.

Abutaha et al. In(2021) focus on the URL phishing site location by AI techniques, precisely based upon lexical features of URLs. Their approach illustrates the importance of parsing URL features to detect phishing attempts in practice.

Takci et al. (2023) proposes a highlight choice and data transformation system that can achieve high accuracy. The new approach effectively identifies inboxes from which these strategies of spamming are effective, therefore advancing the larger goal to improve email security. In the age of refined spam strategies, their findings could not be more relevant.

Al-Khateeb et al. (2023) A 2016 model of a mindfulness attack is proposed.(2023) that targets to decrease the social engineering attacks in web applications. Their research highlights the importance of client awareness in mitigating risks associated with social engineering. Their proposed model highlights the need for ongoing education and training to empower defenders against such attacks.

Alazaidah et al. (2023) However, due affiliations can be utilised as potential by both spam and phishing identification frameworks we thus look at such new multi-name bunching methodology (2023). Their work introduces a smart method to apply relational learning as the basis of name resolution systems, perhaps improving accuracy by capturing subtler patterns in the data through working on multiple names at once.

Atlam & Oluwatimilehin, (2022) A survey of efficient writing on business email compromise (BEC) phishing detection by AI was conducted and led through Atlam & Oluwatimilehin, 2022. An overview of existing work on this line by comparing various AI modalities and their efficacy in detecting BEC attacks, it outlines current practices with specifics and points gaps inherent in the present research.

Nti et al. In (2022) they proposed a StackNet choice combination classifier for network interruption location. Although, their primary focus is on network disruption and the alternative fusion strategy they suggest may be adapted to more robust catch phishing systems.

Batah et al. (2022) provide an example of AI applications outside the traditional phishing, specifically in early diagnosis from cancer of the cervix. Although not directly related to phishing itself, their efforts shed light on just how versatile and powerful AI techniques are in different fields -- including healthcare, which could inspire similar novel methods toward cyber security.

3. Research Gap

Despite the massive progress in AI (ML) techniques for identification of phishing, there are some research gaps. Modern social engineering and mobile attack vectors are constantly evolving, which many of the current model systems struggle with. Additionally, we need larger data sets that capture different client environment and phishing scenarios. In addition, using consistent recognition frameworks alongside ML strategies and improving their generalizability across various platforms and settings is a future research topic.

4. Methodology

The proposed approach uses multiple managed learning techniques to improve the fidelity of fake website detection. The Kaggle data has 11,056 rows with 32 features. This dataset is divided based on entropy, the calibrated dataset presents more advanced accuracy. We then look at how accurate our divided dataset is. We use a combination of relationship analysis and an algorithmic model to identify best-fit loans for every leaf node. Based on this premium, it enhances the accuracy of the model. Unlike blacklisting methods, ML solutions can tackle zero-hour phishing attacks seen as party time through heuristical scans. ML techniques on the other hand also have the advantage of automatically creating order models by analyzing large datasets, eliminating the need to manually create heuristic rules. The architecture of this system and its role in the realm for ML techniques meanwhile playing key roles within phishing detection are shown above which is respectively as: (i) It preprocesses emails stored in an email database at every email individually before passing it to any machine learning solution, (ii) Discovers patterns within correctly predicted class labels during a direct feature importance analysis study with respect to the job subject "Title" similar names.

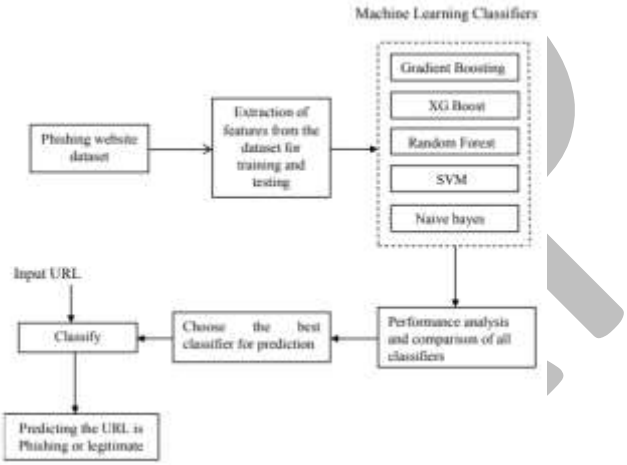

Fig. 1 Proposed Methodology

Dataset: In our model we have opted to connect some phishing dataset retrieved from different Web-based sources, among which Kaggle, with others collected by us. The dataset includes 11,056 lines and 32 features about different phishing and bona fide sites. In the model assessment stage, we conduct test on 30% of Kaggle phishing dataset (testing set) and train with the rest 70%.

Data Preprocessing: Data preprocessing consists of a few steps such as cleaning, case selection, attribute extraction,sscaling and transformation. The goal here is to construct the training dataset robustly. Data cleaning solves all of these problems such as imputing missing values, reducing noise, identifying and removing outliers properly fixing anomalies. Data Integration: The process of combining data sets and datasets. Normalization information is the spot everything data/information will be normalized and coordinated across to keep one information, these blessings with same benefit. The details reduction process not only enhances the datasets but also conserves important information for accurate analytics.

Train-Test Split: To test the model with respect to detection of phishing sites, we partition the dataset into two complementary sets — Training set and Testing set. The data is then split apart as shown above twice in order to save 30% of it for testing, the other 70 will be used to train our model. This division allows the model to learn from the training data and be evaluated on testing data.

Machine Learning Algorithms

Random Forest: Random Forest is one of the most favored performing controlled AI algorithms it produces multiple decision trees to form a "forest." What are the Grouping and Relapse jobs. It is based on ensemble

learning, combining multiple classifiers to solve complex problems and improve model performance. Each choice tree in the Arbitrary Backwoods contributes to the final prediction, and forest takes average votes from all trees for classification tasks. In relapse, forest averages the outcomes from each tree. Pros: Arbitrary Timberland is exact, quick to train in comparison with other methods and particularly fruitful when managing large datasets. Performs well even in the presence of missing data sincronizzaiolsimus. A similar process is used in Bootstrap inspecting which involves selecting random subsets of data for each sample, effectively generating multiple example datasets. Conglomeration then combines these sample datasets into a single compact dimension to offer complete insight. Called change, this is basically the blunder presented by little vacillations in the preparation dataset. When the variation is large, the model may focus on irrelevant or noisy data instead of the underlying signal —the true pattern that you want your model to learn. This leads to overfitting, in which the model does well on the training data but fails to generalize of new unseen information during testing as smaller and larger values of a magnitude won't get separated. Campo Metainformaton are regular change heightens, while Staking happens likewise to the Bootstrap methodology and consistently has high fluctuation however repays by improving general model adequacy. In terms of Irregular Timberland, this method gives the model a high level of resilience while remaining both efficient and incredibly easy to build; an excellent trade-off between accuracy and speed.

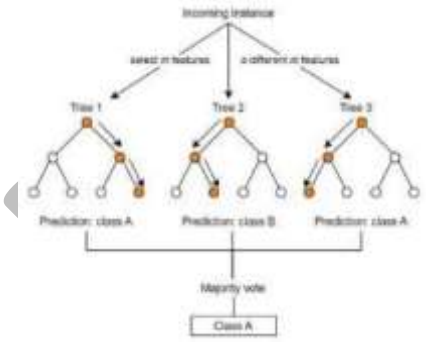

Fig. 2 Shows the Random Forest Simplified

XGBoost: XGBoost is short for Extreme Gradient Boosting, a method that uses gradient boosted decision trees known for their speed and performance. The ahelping high level of outfit learning procedure successively corrects errors in precedenticauseom manufactured models adding up the new model unless other amendments are no longer available. In XGBoost is uses an inclination drop recipe to confine the loss with each new model, streamlining memory and computational assets. It is designed to set up with a training environment which best suits available resources, so it works well for tasks where speed of learning and model performance are paramount. XGBoost is a versatile model as it can be used in both grouping and relapse models.

Naive Bayes:This is a simple technique based on Bayesian theorem and it assumes independence between the features in making predictions. It is quite effective to predict the category of a dataset and handles multi-class characterization tasks. If the independence assumption is true, then Naive Bayes can be more efficient than other algorithms such as logistic regression. It also has fewer setup requirements for ordering. Some Practical Applications of Credulous Bayes classifier includes: Spam Filtering Report Generation It is accepted as a poor estimator, despite its effectiveness. Yet the fact remains that Naive Bayes is a simple and powerful method, with the posterior probability of class c given predictor (feature) value x — P(c/x).

$$P(c|x) = \frac{P(x|c)P(c)}{P(x)}$$

Likelihood, Class Prior Probability, Posterior Probability, Predictor Prior Probability

$$P(c|X) = P(x_1|c) \times P(x_2|c) \times \cdots \times P(x_n|c) \times P(c)$$

Where,

The prior probability of the class is denoted as P(c),
while P(x/c) represents the probability of the predictor given the class,
P(x) is the prior probability of the predictor.

Support Vector Machine (SVM): Support Vector Machine (SVM) is a supervised AI algorithm used for both classification and regression tasks, but it is mainly utilized in solving the problem of classification. In SVM webpage, it is also face the same problem that each element value exerts as a dimension in high layered space. We need to find the hyperplane that best divides data into two classes. The calculation of the SVM means to expand such an outskirt between methods, hence tended to by help vectors — information focuses nearest in estimation to a given piece (support). SVM is a productivity algorithm at any rate in high-measurement spaces — both regarding the time it takes to prepare and expectation. Minimizing memory necessity by utilizing just some preparing focuses from choice capacity (called support vectors), additionally putting away model that data bowed, gives powerphylantic bifurcations for example conditions of help shape or gray supporting straightforwardly contributing areas strength its execution safekeeping limits against overfitting effect_prepainting_Framework6_. Also, SVM is able to provide probability estimates by default through five-fold cross-validation.

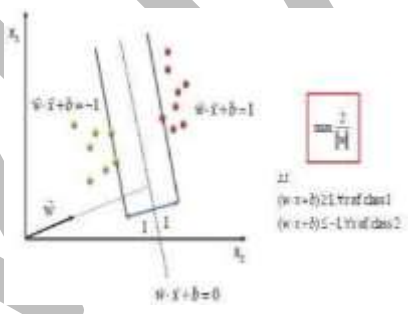

Fig. 3 Two support vectors with Hyperplane

Gradient Boosting: BaggingClassifierInclination supporting classifiers, a gathering of AI strategies; complete powerless learning models to make an amazing prescient model. This technique often uses decision trees and has gained popularity due to its effectiveness in classifying complex datasets. However, angle support will overfit the training data. Regularization techniques will help to overcome this problem by penalizing various aspects of the calculation, reducing overfitting and improving overall performance. Its skill to improve its predictive accuracy, along side this technique remain applied on Phishing detection applications.

5. Results

The tests conducted with the selected approach have provided good results, showing its efficacy in handling an imbalanced dataset, as commonly found in phishing detection tasks. Importantly, none of the analyses exhibited overfitting i.e., the model generalized well to new unseen data without becoming too tailored to the training set. The slope supporting playing out the best among tried classifiers with an astounding generally speaking

exactness of 97.6%. It shows that slope boosting was significantly more successful than other classifiers at accurately identifying phishing attempts. Slope aiding has high accuracy, suggesting it is well-suited for complicated as well as comprehensive phishing detection circumstances.

Other classifiers also performed well, but not as accurate as slope support. Surprisingly, the gromlins bayes classifier also had slight precision than any evaluated models with a rate of approximately 60.5%. This lower accuracy suggests that Naive Bayes was approximately less effective in identifying genuine and phishing sites.

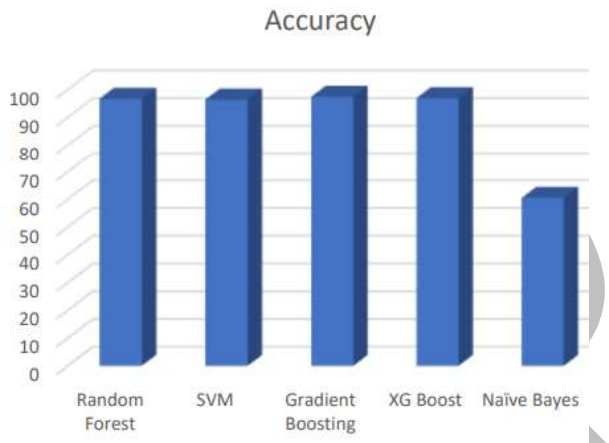

Fig. 2 Comparison of Classification Model Accuracies

6. Conclusion

A comparison of various AI methods for phishing detection revealed significant insight into their performance. This dataset paired data collated from Kaggle with other sources, featuring 11,056 instances of 32 columns in size made up of both phishing and non-phishing websites. After processing with cleaning, normalization and component extraction they started pushing the model preparation/evaluation stage for a weightier collection of data. In the calculations endeavored, Irregular Backwoods (LW), XGBoost, Guileless Bayes and Inclination Supporting So appeared as new continued exactness of 97.6%. This high accuracy reflects within Angle Supporting's best ability of detecting phishing attempts and dealing with big datasets. Random Timberland and XGBoost also performed well, with a high performance in predictive value generalization for Random Forests but slow processing speed. On the other hand The Gullible Bayes classifier identified phishing sites and legitimate site with a perfection of somewhere near to 60.5% [1]

Scope of Future Research:

1. **Enhanced Data Diversity:** Expand datasets to include a wider variety of phishing attacks and legitimate websites from diverse sources. This can help improve model robustness and generalizability across different phishing tactics.

2. **Advanced Algorithms Exploration:** Investigate and test emerging machine learning algorithms and deep learning techniques, such as neural networks and ensemble methods, to further improve detection accuracy and adaptability.

3. **Real-Time Detection Systems:** Develop and evaluate real-time phishing detection systems that can promptly identify and mitigate threats as they occur, integrating with existing security infrastructure.

4. **Feature Engineering:** Explore advanced feature engineering techniques to uncover new patterns and indicators of phishing attempts that may not be captured by current features.

5. **Cross-Platform Testing:** Assess model performance across various platforms and environments, including web browsers, mobile applications, and email systems, to ensure comprehensive coverage.

Suggestion:

- **Use Synthetic Data:** Augment real datasets with synthetic phishing examples for better model training.
- **Explore Hybrid Models:** Combine multiple machine learning techniques to enhance detection accuracy.
- **Boost Security Awareness:** Implement phishing simulation tools for user training.
- **Continuous Monitoring:** Regularly update and monitor phishing detection models to stay ahead of new threats.
- **Protect User Privacy:** Design models that safeguard user data while ensuring effective detection.
- **Support Open Research:** Engage in data sharing and collaborative research in the cybersecurity community.

7. References

1. Alazaidah, R., Samara, G., Almatarneh, S., Hassan, M., Aljaidi, M., and Mansur, H. "Multi-Label Classification Based on Associations." Applied Sciences, vol. 13, no. 8, 2023, p. 5081.
2. Al-Khateeb, M., Al-Mousa, M., Al-Sherideh, A., Almajali, D., Asassfeha, M., and Khafajeh, H. "Awareness Model for Minimizing the Effects of Social Engineering Attacks in Web Applications." International Journal of Data and Network Science, vol. 7, no. 2, 2023, pp. 791-800.
3. Takci, H., Nusrat, F., and Women, B. "Highly Accurate Spam Detection with the Help of Feature Selection and Data Transformation." International Arab Journal of Information Technology, vol. 20, no. 1, 2023, pp. 29-37.
4. Almseidin, M., Zuraiq, A. A., Al-Kasassbeh, M., and Alnidami, N. "Phishing Detection Based on Machine Learning and Feature Selection Methods." International Association of Online Engineering, 2019. Retrieved July 9, 2023.
5. Do, N. Q., Selamat, A., Krejcar, O., Herrera-Viedma, E., and Fujita, H. "Deep Learning for Phishing Detection: Taxonomy, Current Challenges and Future Directions." IEEE Access, 2022.
6. Jain, A. K., and Gupta, B. B. "A Survey of Phishing Attack Techniques, Defence Mechanisms and Open Research Challenges." Enterprise Information Systems, vol. 16, no. 4, 2022, pp. 527-565.
7. Atlam, H. F., and Oluwatimilehin, O. "Business Email Compromise Phishing Detection Based on Machine Learning: A Systematic Literature Review." Electronics, vol. 12, no. 1, 2022, p. 42.
8. Batah, M. S. A., Alzyoud, M., Alazaidah, R., Toubat, M., Alzoubi, H., and Olaiyat, A. "Early Prediction of Cervical Cancer Using Machine Learning Techniques." Jordanian Journal of Computers and Information Technology, vol. 8, no. 4, 2022, pp. 357-369.
9. Nti, I. N., Narko-Boateng, O., Adekoya, A. F., and Somanathan, A. R. "Stacknet Based Decision Fusion Classifier for Network Intrusion Detection." International Arab Journal of Information Technology, vol. 19, no. 3 A, 2022, pp. 478-490.
10. Gandotra, E., and Gupta, D. "An Efficient Approach for Phishing Detection Using Machine Learning." Multimedia Security: Algorithm Development, Analysis and Applications, 2021, pp. 239-253.
11. Abutaha, M., Ababneh, M., Mahmoud, K., and Baddar, S. A. H. "URL Phishing Detection Using Machine Learning Techniques Based on URLs Lexical Analysis." In 2021 12th International Conference on Information and Communication Systems (ICICS), IEEE, 2021, pp. 147-152.
12. Al-Mousa, M. R. "Analyzing Cyber-Attack Intention for Digital Forensics Using Case-Based Reasoning." arXiv preprint, arXiv:2101.01395, 2021.
13. Alazaidah, R., Almaiah, M. A., and Al-Luwaici, M. "Associative Classification in Multi-Label Classification: An Investigative Study." Jordanian Journal of Computers and Information Technology, vol. 7, no. 2, 2021, pp. 166-179.

14. Alazaidah, R., Ahmad, F. K., and Mohsin, M. F. M. "Multi-Label Ranking Based on Positive Pairwise Correlations Among Labels." The International Arab Journal of Information Technology, vol. 17, no. 4, 2020, pp. 440-449.

15. Rashid, J., Mahmood, T., Nisar, M. W., and Nazir, T. "Phishing Detection Using Machine Learning Technique." In 2020 First International Conference of Smart Systems and Emerging Technologies (SMARTTECH), IEEE, 2020, pp. 43-46.

16. Alluwaici, M., Junoh, A. K., AlZoubi, W. A., Alazaidah, R., and Al-luwaici, W. "New Features Selection Method for Multi-Label Classification Based on the Positive Dependencies Among Labels." Solid State Technology, vol. 63, no. 2s, 2020.

17. Sahingoz, O. K., Buber, E., Demir, O., and Diri, B. "Machine Learning Based Phishing Detection from URLs." Expert Systems with Applications, vol. 117, 2019, pp. 345-357.

18. Jain, A. K., and Gupta, B. B. "A Machine Learning Based Approach for Phishing Detection Using Hyperlinks Information." Journal of Ambient Intelligence and Humanized Computing, vol. 10, 2019, pp. 2015-2028.

19. Gupta, B. B., Arachchilage, N. A., and Psannis, K. E. "Defending Against Phishing Attacks: Taxonomy of Methods, Current Issues and Future Directions." Telecommunication Systems, vol. 67, 2018, pp. 247-267.

20. Alluwaici, M., Junoh, A. K., Ahmad, F. K., Mohsen, M. F. M., and Alazaidah, R. "Open Research Directions for Multi-Label Learning." In 2018 IEEE Symposium on Computer Applications & Industrial Electronics (ISCAIE), IEEE, 2018, pp. 125-128.

21. Alazaidah, R., Ahmad, F. K., Mohsen, M. F. M., and Junoh, A. K. "Evaluating Conditional and Unconditional Correlations Capturing Strategies in Multi-Label Classification." Journal of Telecommunication, Electronic and Computer Engineering (JTEC), vol. 10, no. 2-4, 2018, pp. 47-51.