International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 18(15s) 2026 pp. 727-735
https://doi.org/10.70917/ijcisim-2026-4469
Article

Permutation-Based Feature Selection with Multi-Model Deep Learning for Phishing Attack Identification

Nazimunisa[1], Sathineni Chaitanya[2]

[1]Department of Computer Science and Engineering, Sree Dattha Institute of Engineering and Science, Sheriguda, Hyderabad, Telangana, India.
Email: Pnajma22@sreedattha.ac.in
[2]Department of Computer Science and Engineering, Sree Dattha Institute of Engineering and Sciences, Sheriguda, Hyderabad, Telangana, India.
Email: chaitanyasathineni@gmail.com

Abstract: Phishing is a serious danger to safety. It is employed to obtain confidential data from people by luring them into divulging it to a bogus site. In this article, I will be explaining how ML can be leveraged to identify fake emails. It also possesses DL, advanced feature selection and explainable intelligence. FFNN, DNN, TabNet, and Wide and Deep models all use a variety of ways to figure out which traits of phishing are the most important. These include measures of association such as chi-squared analysis, mutual information and permutation importance. To improve the stability of the system an ensemble vote classifier is used with the RF and Bagging with the DT. The SMOTEENN resampling method is used to make the classes more equal. It achieved a 98.7% success rate in the phishing websites dataset and 98.5% success rate in web page phishing dataset, which was more accurate than the individual DL models.The results are based on two public phishing experiments. To explain predictions and show the effect of features like LIME and SHAP, methods like Explainable Artificial Intelligence are used. That's why they are helpful to be open and to build trust. In the real world it is used on the Flask platform in this framework. It provides a safe way for users to register and log in, as well as an online, interactive interface. Users can type in any URL to be scanned and the system will immediately inform them if the URL is a phishing or not. This can lead to a more reliable identification of phishing in online security and making it easier to understand.

Keywords: Phishing detection, cybersecurity, deep learning, neural networks, feature selection, hyperparameter optimization, real-time detection, TabNet, wide and deep model

1. Introduction

The 21st century is a highly interconnected time in cyberspace. Consequently, the concern about cybersecurity is very high and affects people, businesses and groups all around the world. One of the most prevalent forms of cybercrime is phishing, and it poses a serious threat to the users because it can be used to deceive users to provide important information to the attacker, such as their passwords, bank information, and personal identifiers [1]. The term "phishing" is derived from the word "fishing" and refers to the scammer's attempt to obtain the personal information from the victim. These attacks are usually disguised as legitimate, such as that from a bank, government or social network. The attackers take advantage of people's trust to do illegal things [3]. Phishing is widely used in electronic platforms like email, social media, instant messaging and other online services. This makes it a multifaceted threat that could get past even the strongest security measures.

Phishing is developing from a basic email scam to a socially engineered scheme that denies contact and exploit the way people are [5]. This type of attack can be expensive, identity theft, damage your reputation or even be the cause of lawsuits. In addition, phishing attacks frequently are associated with data breaches, ransomware and other

unlawful online operations. This indicates that the threat is growing to additional regions in security [7]. As phishing evolves, much research has been performed to detect phishing with heuristic methods, feature selection and advanced ML methods [8]. It has been proven that selecting suitable features is crucial for enhancing detection performance, and the fusion of heuristic approach and ML models to improve performance [9].

A few ML approaches have been studied recently, which can be used to detect phishing. Some of the models previously tested to determine their ability of identifying malicious emails and phishing websites include FFNN, DNN, Wide and Deep models, and TabNet design [10]. The model will perform better by eliminating more fake positives and negatives, and by increasing its accuracy, via optimisation of algorithms and hyperparameter tweaking (grid search). Additionally, new methods of evaluating, for example, the anti-phishing score, can give us new information about the systems that are evaluating.

In this study, 111 traits are used in a structured manner to test the ML models used to detect phishing emails, using a dataset provided by PhishTank. A feed-forward model was found to be the most useful and the grid search optimization has made it even better in order to get the maximum accuracy and economy. The research is aimed at enhancing the cyber-security in the context of the new phishing attacks by using traditional feature selection techniques and modern ML techniques, and general assessment methods. This will make it easier to spot malicious emails in a variety of situations.

2. Related Work

Phishing is one of the most common problems with cybersecurity because more and more people are using the internet for interaction, making payments, taking and managing personal information. There is an increasing amount of research on ML and heuristic approaches to improve phishing in various aspects. Mosa et al. [11] did a large study on using ML algorithms to spot phishing URL attacks. Their results show that different supervised learning algorithms are good at classifying scam URLs. They found that they could alter the different components to be more accurate at detecting them and reduce the number of false positives. It also helped shape the flexible detection strategies that would be able to track new phishing attacks with data.

Phishing detection method by Yu et al. [12] was based on a neural network with numerous features to detect fake sites by analyzing many URL and web page features. Their approach demonstrated that the accuracy of the detection could be greatly enhanced by employing a combination of various types of features such as domain information, page content characteristics and hyperlink analysis features in comparison to a model which used only a single set of features. The key of this study is that neural network architectures can discover complicated patterns from phishing data. This can be a good way to automatically detect phishing attacks. Novakovic and Markovic [13] studied phishing by using neural networks to find URLs. Neural networks that have been carefully taught can tell the difference between real and fake URLs. Their method showed that URL analysis that tries to use machine learning might be much better than standard rule-based systems because it can change to fit new phishing schemes with little help from a person.

In their study [14], Salihu et al. looked at how useful a heuristic method is for finding phishing URLs. They came to the conclusion that domain-specific rules and pattern recognition is essential to the ability to easily and rapidly identify phishing attempts. They were able to classify phishing sites into categories by structural characteristics, URL properties and patterns of behavior in a short time with little computing power. Tanimu and Shiaeles [15] examined the suitability of using ML to identify scams. The study was centered on selecting appropriate algorithm and evaluating its performance based on its efficiency. They demonstrated that all of which are supervised learning models are members of the family of good phishing detectors when they are trained on well-prepared datasets that have large number of features. This demonstrates the significance of algorithmic evaluation in phishing detection systems in the real-world.

Sanchez-Paniagua et al. [16] created a method to find phishing sites. It was based on web technology and included new data and features which could be applied to multiple uses. Their research was concerned with the methods required for increasing the detection rates using a combination of the various properties of HTML, JavaScript and CSS. They expanded the features to structural as well as content-based features. This allowed for a more comprehensive monitoring system that was able to discover more sophisticated phishing sites that would not have been discovered by other less complete monitoring systems. Wei and Sekiya [17] discussed as well the critical role of selecting the right features for scam detection in machine learning. They suggested ways to find the most important features that can help make classification a lot more accurate. They found that selecting optimal features would make

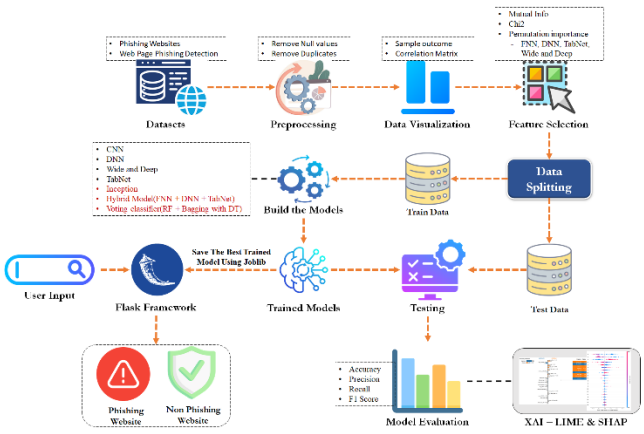
the model more understandable and reduce the power requirements for it, consequently improving phishing detection and making it more widely applicable.

Using different machine learning algorithms, Chinnasamy et al. [18] came up with a strong way to spot phishing attacks. One was conducted to compare the performance of the logistic regression classifier, KNN, and ensemble methods as classifiers. The results revealed that ensemble methods can achieve better results than logistic regression using the best features of logistic regression. The research demonstrated the need to prepare datasets and develop features to strengthen detections. While exploring the possibilities of employing machine learning for detecting fake websites, Dangwal and Moldovan [19] demonstrated the criticality of selecting the proper features. They also found that removing redundant and/or unused features enhanced the performance of the models in terms of both the error and the speed of learning, which is crucial for detection systems operating in real-time.

The article by Yahya [20] looked at a number of ML methods that can be used to find phishing websites. The researcher, however, spoke about the significance of the ensemble of many different classifiers and features for accurate prediction. His research work was on the issue of imbalanced data set and proposed methods for reducing its impact to obtain accurate recognition results, including oversampling and feature weighting. All of these studies provide a comprehensive view of the new breakthrough in research and discovery of fake emails. Their results indicate that the combined technique of feature engineering, feature selection, neural network structures, heuristic analysis, and ensemble learning could improve adaptability, speed and accuracy. Besides, new datasets, better evaluation techniques and optimization strategies have been developed and developed all the time, which enables more reliable phishing detection systems to process more data. All of these factors help to make it possible to conduct further studies on ways to make cybersecurity more effective, particularly with regards to increasingly sophisticated and dynamic hacking attacks.

3. Materials And Methods

The proposed framework uses advanced feature selection and DL algorithms, which allow it to be more effective in detecting phishing. There are two datasets of phishing entries that are freely available and already cleaned, which are required to do the analysis. The features are selected using the mutual information metric, Chi-square and permutation importance of the models Feed-FNN, DNN, TabNet and Wide and Deep. This enables us to identify the most salient features for use in the classification. Some of the DL models used to identify complex phishing patterns include CNN, DNN, Wide and Deep, TabNet and Inception. It is stronger if used in combination with FNN, DNN, TabNet and Voting Classifier, which is a combination of RF and Bagging with the assistance of partners, DT. SMOTEENN reduces imbalance among classes, and Explainable AI, LIME, and SHAP explain features. A Web app, written in Flask, sorts URLs into groups in real time.

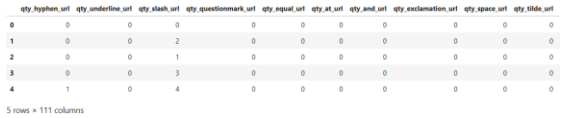


"Fig. 1. System Architecture"

In the illustration, ML is employed to identify scams on web pages. The first step is the datasets. The next step is Preprocessing, in which the data is cleaned. Next comes Feature Selection and Data Splitting, which come after Data Visualization. There are multiple models created, trained and evaluated, including CNN, DNN and Hybrid. The accuracy and the F1 score are some of the factors used to evaluate a model. The trained model is optimized and stored in the Flask system to accept user input and make a guess as to whether the site is phishing or not, based on two types of explainable AI: LIME and SHAP.

A) Dataset Collection

i) Phishing Websites Dataset: The following dataset is a refined version of the Phishing Websites dataset, provided by Mendeley Data, containing 57,405 records and 111 number characteristics related to the URL, domain and webpage features of phishing websites. There are details of the number of symbols, server configuration, SS certificates, redirections, and even the indexing status for each entry. The target label indicates if the website is phishing or not. This dataset provides a good foundation to train and test ML models to identify phishing. It also enables the deep analysis of a broad spectrum of online threat indicators within while preserving the integrity of the data, by preprocessing.

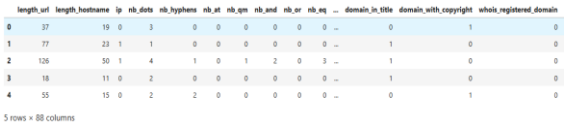


	qty_hyphen_url	qty_underline_url	qty_slash_url	qty_questionmark_url	qty_equal_url	qty_at_url	qty_and_url	qty_exclamation_url	qty_space_url	qty_tilde_url
0	0	0	0	0	0	0	0	0	0	0
1	0	0	2	0	0	0	0	0	0	0
2	0	0	1	0	0	0	0	0	0	0
3	0	0	3	0	0	0	0	0	0	0
4	1	0	4	0	0	0	0	0	0	0

5 rows × 111 columns

Fig. 2. Phishing Websites Dataset

ii) Web page phishing detection: There are 11,256 processed records in the phishing data set on Mendeley Data. These records contain 88 attributes including lexical, host-based and content-based attributes of URLs and websites. [24]. We're studying such things as length, number of symbols, domains, HTTPS, page structure, traffic data, and ranking signals. Each record is marked as either a scam record or a real record, which is what the target variable is. This type of data provides us with a broad spectrum of data that we can use to develop and evaluate ML models to identify phishing websites.



	length_url	length_hostname	ip	nb_dots	nb_hyphens	nb_at	nb_qm	nb_and	nb_or	nb_eq	...	domain_in_title	domain_with_copyright	whois_registered_domain
0	37	19	0	3	0	0	0	0	0	0	...	0	1	0
1	77	23	1	1	0	0	0	0	0	0	...	1	0	0
2	126	50	1	4	1	0	1	2	0	3	...	1	0	0
3	18	11	0	2	0	0	0	0	0	0	...	1	0	0
4	55	15	0	2	2	0	0	0	0	0	...	0	1	0

5 rows × 88 columns

“Fig. 3. Web Page Phishing Detection Dataset”

B) Pre-Processing:

The pre-processing tasks ensure correct data, selection of the salient features and an unbiased assessment. They do it by cleaning the data, displaying the data, identifying the important features in the data, and partitioning the data into sets that will enable them to build a good phishing detection model.

i) Data Processing: This step makes sure the data is correct by getting rid of null values that could lead to errors during analysis and duplicate records that could make the model too complicated or biased. Cleaning the data provides you with a good foundation to work with once you've manipulated the data and trained a model. This makes sure that all the data in the ML process is of high quality and consistent.

ii) Data Visualization: Visualization aids comprehension of data organization and how things fit together. The attribute dependencies and impacts can be evaluated by using a correlation matrix and sample data can be used to illustrate class distributions. These types of graph give more attention to study and provide insight into various factors which impact phishing detection.

iii) *Feature Selection:* There are several methods with which feature selections are made to identify the most useful features. The importance of the features can be computed using mutual information or chi-square tests, while the importance of the permutations can be computed using DL based models such as FNN, DNN, TabNet or Wide & Deep. Top 20 traits are retained to ensure ease of understanding of the model and improved model performance:

C) Training and Testing:

The features data is divided into training and testing sets to ensure the quality and suitability of the data. This ensures that the models are trained on one data set, and then evaluated with another data set. This will make it easier to obtain a more realistic notion of the models' ability to predict and generalize for random sets of phishing and legal websites.

D) Algorithms

CNN (Convolutional Neural Network): Use convolutional and pooling layers to figure out a website's URLs and attributes, as well as its local dependencies and structure. Simplifies making accurate predictions by identifying small indicators of phishing and more complex correlations between variables; makes it easier to distinguish the real sites from the fake ones with certainty.

$$"S(i,j) = \sum_{m}\sum_{n} I(i+m, j+n) \cdot K(m,n) \quad (1)"$$

DNN (Deep Neural Network): Learns associations between site and URL traits that are not linear and have many levels. [26]. It helps to find high-level abstractions and relationships, as well as high-dimensional data, which makes it possible to find harmful site activity.

Wide and Deep: combines both linear and deep parts by modeling both high-level abstractions and clear feature interactions at the same time. It has more complex inter-relationships between categories and over time, makes better generalisations, higher accuracy, and finds subtle phishing trends in a large variety of datasets.

TabNet: The data from Web sites and URLs are tabular, and are processed serially to identify relevant features as they are encountered. [30]. Maps with a higher accuracy and is more readable. It does that by carefully handling complex interactions, and highlighting major indicators of phishing.

Inception: Parallel convolutions at different scales are used to look at the website's attributes and find both local and world patterns. Identifies more phishing indicators such as the structure of the URL, the features of the content, and the interactions with the metadata to increase accuracy of detection.

$$"O = concat\left(F_1(X), F_2(X), F_3(X), F_4(X)\right) \ (2)"$$

Hybrid Model (FNN + DNN + TabNet): The outputs of feedforward, DNN and TabNet are combined to look for basic, complex and attention-based patterns of features. Enhances the strength and usefulness of managing high dimensional, uneven web data in the real world, making it clearer and more precise.

Voting Classifier (RF + Bagging with DT): Combines DT, RF and Bagging to increase the stability of the prediction, reduce the variation and highlight the most important traits. It is very accurate and reliable results and performs better than the solo models in classifying phishing or real websites.

$$\hat{y} = argmax_c\left(\sum_{i=1}^{n} II\ (\hat{y}_i = c)\right) \quad (3)$$

E) Integration of XAI and Flask Framework:

In this case, when you integrate XAI (Explainable Artificial Intelligence) and the Flask framework, you can leverage ML models in an interesting and understandable way. If XAI methods were coupled with flask's light weight web functionalities, customers would be able to obtain predictions and understand how these were made. This approach enhances transparency, enabling you to understand the impact of various features, models, and prediction accuracy on the overall system. This is particularly useful for private jobs, such as identifying phishing e-mails or analyzing healthcare information.

This set-up would use Flask as the user interface to work with and handle input data and show the output, while XAI methods would explain how the model came to its conclusions. Integrating these two things makes sure that complicated DL or ensemble models can be read, trusted, and used. This establishes a bridge between high performance AI and real-world applications.

4. Experimental Results

Accuracy: A test is said to be accurate if it can correctly tell the difference between cases of patients and healthy people. To determine the validity of the test, the percentage of true positives and true negatives should be determined among all the cases that are tested. In math terms, this can be written as:

$$"Accuracy = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{FP} + \mathrm{TN} + \mathrm{FN}} \quad (4)"$$

Precision: To determine what percentage of the correct cases were checked good is a method of Precision. However; the way to find the precision is:

$$"Precision = \frac{\text{True Positive}}{\text{True Positive } + \text{False Positive}} \quad (5)"$$

Recall: It's a form of ML that enables you to discover all the important examples in a class. The outcome is a measure of how well the model performs in identifying instances of a class. It is the ratio of correct answer out of all of the positive observations.

$$"Recall = \frac{\text{TP}}{\text{TP } + \text{ FN}} \quad (6)"$$

F1-Score: The F1 Score is a way to measure how accurate a ML model is. Combines a model's recall and precision. The accuracy measure can be used to discover the probability that the model will be correct in making predictions.

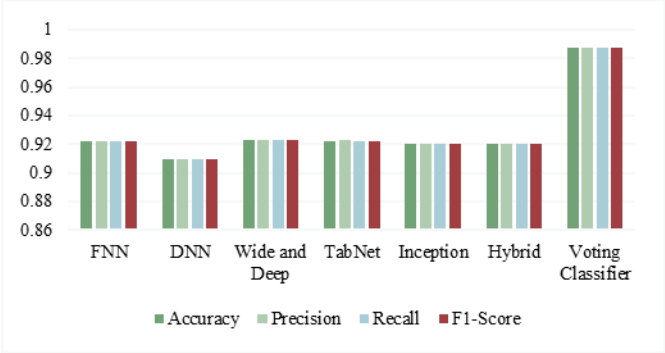
$$"F1\ Score = 2 * \frac{Recall \text{ X } Precision}{Recall + Precision} * 100 \quad (7)"$$

"Table.1 Performance Evaluation – Phishing Dataset"

ML Model	Accuracy	Precision	Recall	F1-Score
FNN	0.922	0.922	0.922	0.922
DNN	0.909	0.909	0.909	0.909
Wide and Deep	0.923	0.923	0.923	0.923
TabNet	0.922	0.923	0.922	0.922
Inception	0.920	0.920	0.920	0.920
Hybrid	0.920	0.920	0.920	0.920
Voting Classifier	0.987	0.987	0.987	0.987

Table 1 shows how well different ML models work at finding scam attempts. It also indicates that in general, the Voting Classifier is the best.

Fig. 4. Comparison Graph Phishing Dataset



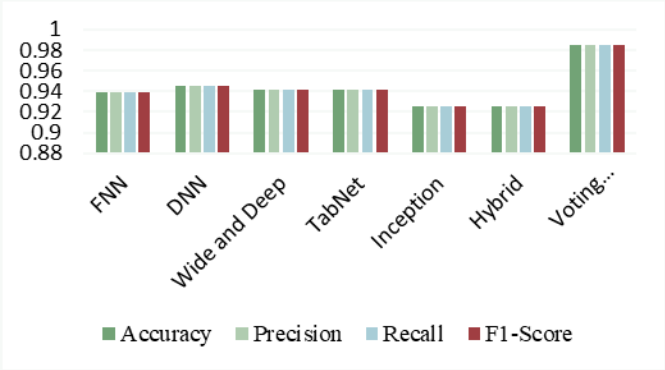
The results of the model are shown in Figure 4. The accuracy, precision, recall and f1 score are indicated in green, lightgreen, blue, and red, respectively. The colors indicate that the Voting Classifier performs well when compared with other classifiers.

Table.2 Performance Evaluation – Web Phishing Dataset

ML Model	Accuracy	Precision	Recall	F1-Score
FNN	0.939	0.939	0.939	0.939
DNN	0.945	0.945	0.945	0.945
Wide and Deep	0.942	0.942	0.942	0.942
TabNet	0.942	0.942	0.942	0.942
Inception	0.926	0.926	0.926	0.926
Hybrid	0.926	0.926	0.926	0.926
Voting Classifier	0.985	0.985	0.985	0.985

The results in Table 2 are from different ML models to detect phishing. Voting Classifier is demonstrated to be the most effective one generally.

Fig. 5. Comparison Graph Web Phishing Dataset



The accuracy (green), precision (light green), recall (blue), and F1-Score (red) are displayed in Figure 5. Clearly, the Voting Classifier does better than all the others.

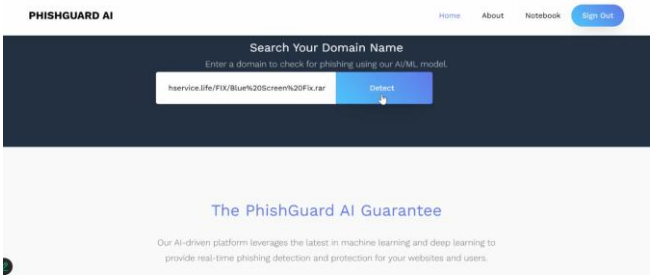


Fig. 6. Enter Input URL

As seen in figure 6, the user can type in a name, making it possible to find phishing or real sites automatically.

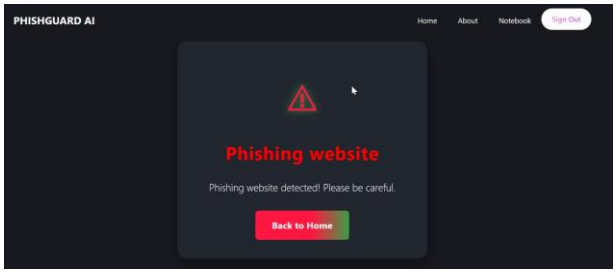


Fig. 7. Predicted Result

The system is able to accurately classify a visiting domain as a phishing domain as seen in Fig. 7. This shows how good automatic detection systems are.

Fig. 8. Enter Input URL

In Fig. 8, users can submit a domain, and the system will immediately let them know whether it is a phishing site or not.

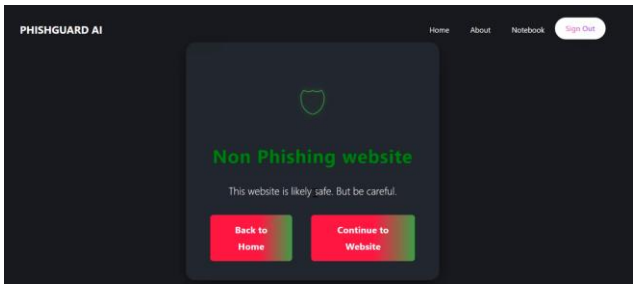


Fig. 9. Predicted Result

Fig. 9 shows reliable automatic detection when the system figures out that the domain that is visiting it is not a phishing site.

5. Conclusion

In the study, the authors demonstrated a good method for the detection of phishing emails, which used advanced feature selection algorithms, DL and ensemble models. The best way to discover the most beneficial components in the models and make them more effective is to eliminate unnecessary components. This can be achieved with techniques such as permutation importance, chi-square, or mutual information in models such as FNN, DNN, TabNet and Wide & Deep. CNN and a combination of FNN, DNN and TabNet DL models were used to make the prediction better. The SMOTEENN was used to balance the classes and thus, it would be better for the generalization to real phishing scenarios. The voting classifiers using RF and Bagging with DT showed consistent results to all models examined with a success rate of 98.7 % on phishing web pages dataset and 98.5 % on web page phishing dataset. Furthermore, SHAP and LIME were combined to facilitate interpretability of the impact that features have on their predictions. This renders the system very accurate, transparent and dependable. So it can be implemented as a real-time app in a phishing detection context.

More work could be done to make this phishing detection system more flexible to new phishing attacks by incorporating real time web traffic to the datasets. If NLP frameworks are applied to the text found on websites, domain names and embedded script, they can significantly enhance the detection accuracy. Considering more complex group methods and reinforcement learning would be very helpful to be more flexible in varying situations. Graph-based techniques can be applied to discover relationships between domains, IP addresses, and URLs that aren't apparent. In the future, lightweight models can be studied and applied in areas of restricted resources and mobility of humans. Furthermore, AI-driven processes that could be enhanced will make things more clear, thus increasing user confidence. Finally, integrating browser add-on with the business level security systems may make them more user-friendly and more likely to be enacted.

References

1. Zara, U., Ayub, K., Khan, H. U., Daud, A., Alsahfi, T., & Gulzar, S. (2024). Phishing website detection using deep learning models. IEEE Access.
2. Sahingoz, O. K., BUBE, E., & Kugu, E. (2024). Dephides: Deep learning based phishing detection system. Ieee Access, 12, 8052-8070.
3. Şengel, Ö. (2024, October). Analysis of Learning Techniques for Phishing Website Detection. In 2024 Innovations in Intelligent Systems and Applications Conference (ASYU) (pp. 1-6). IEEE.

4. Aldakheel, E. A., Zakariah, M., Gashgari, G. A., Almarshad, F. A., & Alzahrani, A. I. (2023). A deep learning-based innovative technique for phishing detection in modern security with uniform resource locators. Sensors, 23(9), 4403.
5. Do, N. Q., Selamat, A., Krejcar, O., Yokoi, T., & Fujita, H. (2021). Phishing webpage classification via deep learning-based algorithms: An empirical study. Applied Sciences, 11(19), 9210.
6. R. Jayaraj, A. Pushpalatha, K. Sangeetha, T. Kamaleshwar, S. U. Shree, and D. Damodaran, ‘‘Intrusion detection based on phishing detection with machine learning,’’ Meas., Sensors, vol. 31, Feb. 2024, Art. no. 101003. [Online]. Available: https://www.sciencedirect.com/
7. science/article/pii/S2665917423003392
8. M. R. Chinguwo and R. Dhanalakshmi, ‘‘Detecting cloud based phishing attacks using stacking ensemble machine learning technique,’’ Int. J. Res. Appl. Sci. Eng. Technol., vol. 11, no. 3, pp. 360–367, Mar. 2023, doi: 10.22214/ijraset.2023.49422.
9. C. Rajeswary and M. Thirumaran, ‘‘The LSTM-based automated phishing detection driven model for detecting multiple attacks on tor hidden services,’’ J. Intell. Fuzzy Syst., vol. 44, no. 6, pp. 8889–8903, Mar. 2023, doi: 10.3233/jifs-224142.
10. B. Subba, ‘‘A heterogeneous stacking ensemble-based security framework for detecting phishing attacks,’’ in Proc. Nat. Conf. Commun. (NCC), Feb. 2023, pp. 1–6, doi: 10.1109/NCC56989.2023.10068026.
11. K. R. Nataraj, D. K. Yashaswini, R. Hema, N. S. Pawar, and S. Yashaswi, ‘‘Phishing attack detection using machine learning,’’ in Proc. 4th Int. Conf. Data Sci., Mach. Learn. Appl. Singapore: Springer, 2023, pp. 355–370, doi: 10.1007/978-981-99-2058-7_33.
12. D. T. Mosa, M. Y. Shams, A. A. Abohany, E.-S. M. El-Kenawy, and M. Thabet, ‘‘Machine learning techniques for detecting phishing URL attacks,’’ Comput., Mater. Continua, vol. 75, no. 1, pp. 1271–1290, 2023, doi: 10.32604/cmc.2023.036422.
13. S. Yu, C. An, T. Yu, Z. Zhao, T. Li, and J. Wang, ‘‘Phishing detection based on multi-feature neural network,’’ in Proc. IEEE Int. Perform., Comput., Commun. Conf. (IPCCC), Nov. 2022, pp. 73–79, doi: 10.1109/IPCCC55026.2022.9894337.
14. J. Novakovic and S. Markovic, ‘‘Detection of URL-based phishing attacks using neural networks,’’ in Proc. Int. Conf. Theor. Appl. Comput. Sci. Eng. (ICTASCE), Sep. 2022, pp. 132–136.
15. S. A. Salihu, I. D. Oladipo, A. A. Wojuade, M. AbdulRaheem, A. O. Babatunde, A. R. Ajiboye, and G. B. Balogun, ‘‘Detection of phishing URLs using heuristics-based approach,’’ in Proc. 5th Inf. Technol. Educ. Develop. (ITED), Nov. 2022, pp. 1–7, doi: 10.1109/ITED56637.2022.10051199.
16. J. Tanimu and S. Shiaeles, ‘‘Phishing detection using machine learning algorithm,’’ in Proc. IEEE Int. Conf. Cyber Secur. Resilience (CSR), Jul. 2022, pp. 317–322, doi: 10.1109/CSR54599.2022.9850316.
17. M. Sánchez-Paniagua, E. Fidalgo, E. Alegre, and R. Alaiz-Rodríguez, ‘‘Phishing websites detection using a novel multipurpose dataset and web technologies features,’’ Expert Syst. Appl., vol. 207, Nov. 2022, Art. no. 118010, doi: 10.1016/j.eswa.2022.118010.
18. Y. Wei and Y. Sekiya, ‘‘Feature selection approach for phishing detection based on machine learning,’’ in Proc. Int. Conf. Appl. CyberSecur. (ACS), H.R.HassenandH.Batatia,Eds.,Cham,Switzerland:Springer, Jan.2022, pp. 61–70.
19. P. Chinnasamy, N. Kumaresan, R. Selvaraj, S. Dhanasekaran, K. Ramprathap, and S. Boddu, ‘‘An efficient phishing attack detection using machine learning algorithms,’’ in Proc. Int. Conf. Advancements Smart, Secure Intell. Comput. (ASSIC), Bhubaneswar, India, Nov. 2022, pp. 1–6.
20. S. Dangwal and A.-N. Moldovan, ‘‘Feature selection for machine learning based phishing websites detection,’’ in Proc. Int. Conf. Cyber Situational Awareness, Data Anal. Assessment (CyberSA), Dublin, Ireland, Jun. 2021, pp. 1–6.
21. F. Yahya, ‘‘Detection of phishing websites using machine learning approaches,’’ in Proc. Int. Conf. Data Sci. Appl. (ICoDSA), Bandung, Indonesia, 2021, pp. 40–47.
22. F. Salahdine, Z. E. Mrabet, and N. Kaabouch, ‘‘Phishing attacks detection a machine learning-based approach,’’ in Proc. IEEE 12th Annu. Ubiquitous Comput., Electron. Mobile Commun. Conf. (UEMCON), New York, NY, USA, Dec. 2021, pp. 250–255.
23. A. Alswailem, B. Alabdullah, N. Alrumayh, and A. Alsedrani, ‘‘Detecting phishing websites using machine learning,’’ in Proc. 2nd Int. Conf. Comput. Appl. Inf. Secur. (ICCAIS), Riyadh, Saudi Arabia, May 2019, pp. 1–6.
24. M. Almseidin, A. A. Zuraiq, M. Al-Kasassbeh, and N. Alnidami, ‘‘Phishing detection based on machine learning and feature selection methods,’’ Int. J. Interact. Mobile Technol. (iJIM), vol. 13, no. 12, p. 171, Dec. 2019. [Online]. Available: https://www.learntechlib.org/p/216410
25. M. Baykara and Z. Z. Gürel, ‘‘Detection of phishing attacks,’’ in Proc. 6th Int. Symp. Digit. Forensic Secur. (ISDFS), Antalya, Turkey, Mar. 2018, pp. 1–5.
26. K. L. Chiew, K. S. C. Yong, and C. L. Tan, ‘‘A survey of phishing attacks: Their types, vectors and technical approaches,’’ Expert Syst. Appl., vol. 106, pp. 1–20, Sep. 2018.
27. H. Zuhair, A. Selamat, and M. Salleh, ‘‘Feature selection for phishing detection: A review of research,’’ Int. J. Intell. Syst. Technol. Appl., vol. 15, no. 2, p. 147, 2016.
28. Zara, U., Ayub, K., Khan, H. U., Daud, A., Alsahfi, T., & Gulzar, S. (2024). Phishing website detection using deep learning models. IEEE Access.
29. Bhimavarapu, U. (2025). Phishing Attack Response and Risk Mitigation Using Deep Learning and Machine Learning. In Critical Phishing Defense Strategies and Digital Asset Protection (pp. 109-120). IGI Global Scientific Publishing.