# Unmasking Deception Strategies For Attack Detection

[1] V. Nivetha, [2]R.Tamilkodi, [3*]Repaka Subbarao, [4] Pitta Rupesh Sai Teja,
[5]Chinta Sowmya,[6]Chamarthi Leela Sri Krishna Pavan Murthy

[1,2 3 4 5 6] Department of Computer Science and Engineering (AIML&CS)
Godavari Institute of Engineering & Technology, Rajahmundry, Andhra Pradesh, India

[1]nivevenkat96@giet.ac.in,[2]tamil@giet.ac.in,
[3*]subbaraorepaka@gmail.com,[4]rupeshsaiteja9841@gmail.com,
[5]sowmyachinta123@gmail.com,[6]chamarthikrishnapavan@gmail.com

**Abstract.** In light of the increasing complexity of digital security risks, This work aims to utilize machine learning algorithms, particularly Support Vector Machines (SVM) and Naive Bayes, to bolster the identification and mitigation of deceptive practices in digital environments. The utilization of SVM and Naive Bayes algorithms in tandem offers a comprehensive approach to deception detection, leveraging the strengths of each algorithm to create a more robust and accurate system. Support Vector Machines represent a subset of supervised machine learning methods utilized for tasks involving classification and regression. The core aim of an SVM is to identify the optimal hyperplane that effectively segregates distinct classes within the feature space. SVMs are effective in high-dimensional spaces and excel especially in handling intricate data distributions. Conversely, Naive Bayes operates as a probabilistic classification algorithm rooted in Bayes' theorem. Naive Bayes is often surprisingly effective, especially in text classification and spam filtering. Naive Bayes is computationally efficient and requires a relatively small amount of training data. It's particularly well-suited for situations where the independence assumption does not markedly impact the classification performance. By this work on deception identification the accuracy that was generated when compare to other works is 90%, and time taken to predict is 12ms with the precision of 0.88.

**Keywords:** machine learning, Naïve Bayes, SVM, digital assaults, and phishing attacks

## 1 Introduction

Phishing attacks are a common and deceptive cyber threat, leveraging human susceptibilities to unlawfully obtain access to confidential data. In a typical phishing attack, malicious actors employ deceptive tactics, often masquerading as trustworthy entities, to Deceive individuals into divulging confidential data including login credentials, financial information, or private information.[8] Such attacks can emerge across multiple channels, such as email, online networks, and chat platforms, and even phone calls, making them a versatile and ever-evolving threat in the digital landscape, one of the


K. R. Madhavi et al. (eds.), *Proceedings of the International Conference on Computational Innovations and Emerging Trends (ICCIET 2024)*, Advances in Computer Science Research 112,
https://doi.org/10.2991/978-94-6463-471-6_133

key challenges in combating phishing attacks lies in their constantly evolving nature[9]. Cybercriminals continuously refine their strategies, adapting to security measures and exploiting new technologies to enhance the sophistication of their campaigns. As a result, organizations and individuals must remain vigilant, employ robust cyber security practices, and stay informed about the latest phishing tactics to mitigate the risk of falling victim to these deceptive schemes[10]. Counter-measures against phishing attacks include user education and awareness programs, multi-factor authentication, email filtering systems, and the use of advanced threat detection technologies[11]. By fostering a culture of cyber security awareness and implementing proactive measures, individuals and organizations can significantly diminish the effectiveness of phishing attempts and fortify their defenses against this persistent and ever-present threat. Key Characteristics of Phishing Attacks are Deceptive Communication, Social Engineering, Fraudulent Websites, Email Spoofing[12]. Phishing attacks can lead to significant consequences, including unauthorized access to accounts, financial loss, identity theft, and compromise of personal or corporate data. In order to lessen the susceptibility to phishing incidents, individuals and organizations employ preventive measures such as email filtering systems, user education and awareness programs, multi-factor authentication, and a cautious approach to unsolicited or unexpected communications.

## 2 Literature Survey

This project provides an OFS-NN, a model for identifying phishing websites that uses an optimal feature selection strategy, in [1]. To assess the contribution of each attribute to the discovery of these websites, a feature validity value (FVV) index was created for the proposed model. This team is now developing an algorithm to extract the most useful aspects of phishing websites using this newly created database. This selected strategy should be able to address the overfitting problem in the neural network to a significant degree. The best qualities are then used to train a neural network into a strong algorithm adept at identifying phishing URLs. The Fuzzy Rough Set approach (FRS) [2] hypothesis was established to give a mechanism for picking the most significant qualities from a limited set of standard datasets. These traits are then passed into a pair of classification algorithms to see whether they can detect any phishing attempts. To investigate how feature selection for FRS may be utilized to build a general phishing detection system, the models are trained on a fresh dataset containing 14,000 website samples. Even though understanding the features is critical to the model's accuracy, feature engineering is essential for devising solutions to detect phishing websites. While the features gathered from these categories are comprehensible, the time needed to obtain them poses a notable limitation. To solve this issue, the authors propose a multidimensional phishing detection feature (MFPD) approach [3], which emphasizes a quick detection technique using deep learning. The Web Crawler-based Phishing Attack Detector (WC-PAD) presents a three-stage phishing detection approach[4]. Online material, traffic, and a URL are examples of input features. At this moment, classification is finished. A deep learning-based method for instantly identifying phishing URLs is

called PhishingNet [5]. A detection system was created to stay up to date with phishing sites and the constantly shifting online landscape. This is entirely client-side and independent of any external resources as it takes into account a wide range of distinguishing factors from the source code of web pages and URLs. [6]. Parse tree validation [7] is a technique used to detect whether or not a website is a phishing effort.

## 3 Proposed Methodology

This work explains how to classify features from the phishing website database by establishing input and output components for the ELM classifier. ELM's results demonstrate superior performance when compared to those obtained using other classifiers (SVM and NB). High-performing categorization against phishing activity on websites is achieved by the use of below mentioned study's architecture, which is suitable in automated systems. This study exhibits superior test performance, achieving an 80.18% success rate, surpassing comparable research in the literature. Integrating data from several sources during model training is thought to increase proposed accuracy. The processing time must be shortened.

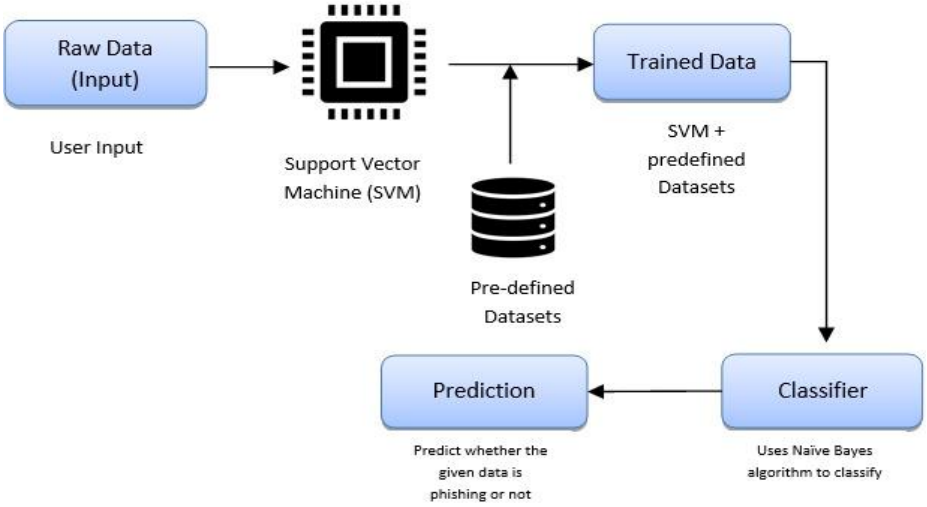


**Fig. 1.** System Architecture of Unmasking Domains and IP

### 3.1 Components of System Architecture:

**Raw Data:** Raw data denotes the untouched, original information collected and analyzed to identify potential phishing attacks. This raw data can come from various sources, such as emails, websites, network traffic, or user behavior.

**Support Vector Machine (SVM)**: SVM operates by delineating distinct data characteristics through optimal hyperplanes. In SVM, the most effective hyperplane is identified as the one maximizing the separation between class boundaries. This method classifies data by locating the exceptional hyperplane that effectively separates diverse sets of information into their respective categories. The key features proximate to the separating vectors are termed support vectors.

**SVM Implementation:**
Step1: Import the phishing keywords dataset from the sklearn.datasets module.
Step2: Divide the dataset into input features and target variables.
Step3: Develop and train SVM classifiers employing the Radial Basis Function (RBF) kernel.
Step4: Generate a scatter plot illustrating the distribution of input features.
Step5: Illustrate the first decision boundary on the plot.
Step6: Depict the second decision boundary on the plot.

**Predefined Datasets**: Predefined datasets, also known as pre-existing or curated datasets, refer to collections of data that have been assembled, prepared, and made available for specific purposes, such as research, analysis, testing, or training machine learning models.

**Trained Data:** Trained Data generally refers to a dataset utilized for the training purpose of an AI-based model or algorithm to identify and classify phishing attacks. This dataset typically contains a collection of examples of both legitimate (non-phishing) communications and phishing attempts and websites.

**Classifiers:** Classifiers refer to algorithms or models that are used to classify incoming data, such as emails, websites, or messages, into different categories based on their likelihood of being phishing attempts or legitimate communications. Classifiers are a fundamental component of the systems designed for the identification of phishing activites, and they help automate the process of identifying and flagging potential threats associated with phishing.

**Naive Bayes:** Naive Bayes functions as a algorithm relying on probability that sorts data into categories based on how likely it is that a given item belongs in that category. Naive Bayes is an algorithm that makes no assumptions about the relationship between features. The initiative is designed to recognize potentially fraudulent accounts by analyzing factors such as the timing, date, language, and location of their posts. Even though some of these qualities depend on each other or on the presence of the other

characteristics, I still believe that they all contribute to a higher possibility that the false profile exists.

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} ( SEQ "equation" \n * MERGEFORMAT 1)$$

From the above formula (1),

1. P(A|B) indicates the updated probability, illustrating the chance of hypothesis A given the event B transpires.
2. P(B|A) denotes the chance probability, reflecting the probability of observed evidence B under the assumption that proposition A holds true.
3. P(A) represents the initial likelihood, indicating the likelihood of assumption A before any data is observed.
4. P(B) denotes the overall likelihood, representing the aggregate probability of data B.

**Naive Bayes Implementation:**
Step1: Perform data pre-processing tasks to prepare the dataset.
Step2: Apply the Naive Bayes algorithm to the training set for model fitting.
Step3: Use the trained model to predict test results.
Step4: Evaluate the accuracy of the predictions and create a confusion matrix for a detailed assessment.
Step5: Visualize the outcomes of the test set to gain insights into the model's performance.

**Prediction:** Prediction refers to the process of using a model or algorithm to make an educated guess or assessment about whether a particular piece of data, such as an email, website, or message, is likely to be a phishing attempt or a legitimate communication

## 4 Results and Discussion

In this work, we showcase the identifying of phishing URLs. As illustrated in Diagram 1, the input (URL) is provided, and the URL undergoes analysis by a support vector machine, comparing it with predefined datasets. Conversely, Figure 2 depicts the determination of the legitimacy of the given URL or the presence of phishing data through a classifier utilizing the Naïve Bayes algorithm.

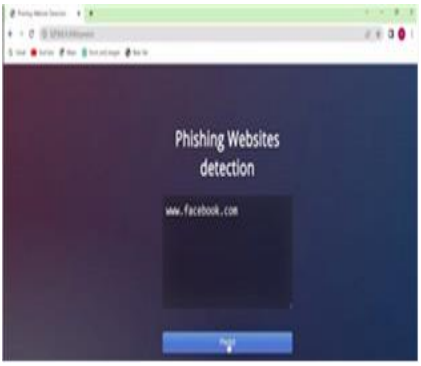


**Fig 2:** Takes the User Input the check whether the URL is phished or not.

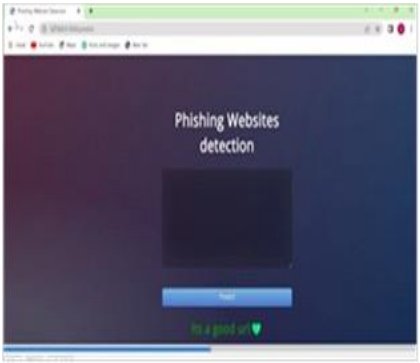


**Fig 3:** On validating the input, we can determine that the given URL contains no phishing data

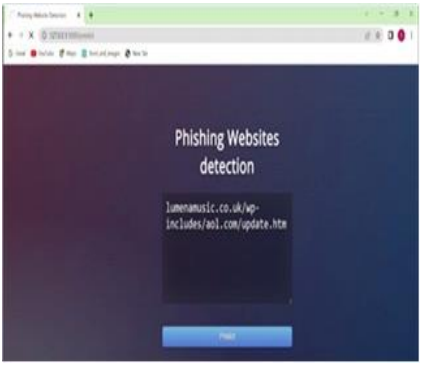


**Fig 4:** Takes the User Input the check whether the URL is phished or not.

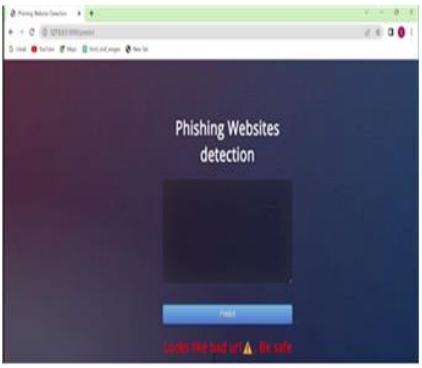


**Fig 5:** On validating the input, we can determine that the given URL contains phishing data

Conversely, in Figure 3, the input is directed to the SVM, mirroring the process outlined in Figure 1, where the input undergoes validation before being processed by the classifier. As depicted in Figure 4, the resulting output indicates the presence of phishing datasets, prompting the display of a warning message.

**Table 1.** Comparison between Proposed and Existing Models

| Paper Title | Algorithms | Acc | Precision | Time |
|---|---|---|---|---|
| E. Zhu et. al [1] | OFS-NN | 40.2% | 0.769 | 20ms |
| Mahdieh et al [2] | SVM & FRS | 46% | 0.778 | 15ms |
| P. Yang et al. [3] | CNN-BiLSTM | 71.6% | 0.635 | 18ms |
| Y. Huang et al.[4] | CNN-RNN | 69.5% | 0.627 | 26ms |
| Proposed Model | SVM | 90% | 0.886 | 12ms |
| Proposed model | Naïve Bayes | 93% | 0.927 | 8ms |

From the above Table.1, the work conducts an in-depth comparative examination of diverse algorithms employed in the detection of phishing attacks, with their performance metrics succinctly summarized in the above table. Zhu et al. [1] introduced the OFS-NN algorithm, achieving a 40.2% accuracy (Acc), a precision of 0.769, and a processing time of 20ms. In another work, Mahdieh et al. [2] employed SVM and FRS, achieving a 46% accuracy (Acc), a precision of 0.778, and a processing time of 15ms. Yang et al. [3] proposed the CNN-BiLSTM algorithm, demonstrating significant improvement with a 71.6% accuracy (Acc), a precision of 0.635, and a processing time of 18ms. Similarly, Huang et al. [4] utilized the CNN-RNN algorithm, achieving a 69.5% accuracy (Acc), a precision of 0.627, and a processing time of 26ms.

In contrast, the novel model presented in this work relies on SVM & Naive Bayes as a classifier, yielding remarkable results with a 90% accuracy (Acc), a precision of 0.886, and a processing time of 12ms. Notably, an alternative version of the proposed model incorporates the Naïve Bayes algorithm, showcasing superior performance metrics with a 93% accuracy (Acc), a precision of 0.927, and an impressively efficient processing time of 8ms. These findings underscore the efficacy of the proposed model, particularly when employing the Naïve Bayes algorithm, positioning it as a robust solution for phishing detection that outperforms the compared algorithms in regards to both accuracy (Acc) and processing velocity.

## 5 Conclusion

The work presented here introduces Anti-Phishing Extension, a solution for dealing with phishing information. Three parts make up the proposed system: redirecting user information, malicious scheme recognition, and translating IP addresses from URLs. The illegal use of financial accounts, credit cards, social media profiles, and so on is an example of identity theft. The proposed APE approach enhances the rate and precision with which phishing assaults are identified. The JavaScript code utilized in this project illustrates the proposed APE (Automatic Page Evaluation) approach shows a considerable reduction in CPU utilization while maintaining higher accuracy when compared to previous techniques.

## References

[1] E. Zhu, Y. Chen, C. Ye, X. Li, and F. Liu. Ofs-nn: An effective phishing websites detection model based on optimal feature selection and neural network. IEEE Access, 7:73271–73284, 2019.Reid G. Smith and Joshua Eckroth. Building ai applications: Yesterday, today, and tomorrow. AI Magazine, 38(1):6–22, Mar. 2017.

[2] Mahdieh Zabihimayvan and Derek Doran. Fuzzy rough set feature selection to enhance phishing attack detection, 03 2019.

[3] P. Yang, G. Zhao, and P. Zeng. Phishing website detection based on multidimensional features driven by deep learning. IEEE Access, 7:15196–15209, 2019.

[4] Y. Huang, Q. Yang, J. Qin, and W. Wen. Phishing url detection via cnn and attention-based hierarchical rnn. In 2019 18th IEEE International Conference On 55 Trust, Security And Privacy In Computing And Communications/13th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE), pages 112–119, 2019.

[5] Steven Aftergood. Cybersecurity: The cold war online. Nature, 547:30+, Jul 2017. 7661.

[6] M. M. Yadollahi, F. Shoeleh, E. Serkani, A. Madani, and H. Gharaee. Anadaptive machine learning based approach for phishing detection using hybrid features. In 2019 5th International Conference on Web Research (ICWR), pages 281–286, 2019.

[7] C. E. Shyni, A. D. Sundar, and G. S. E. Ebby. Phishing detection in websites using parse tree validation. In 2018 Recent Advances on Engineering, Technology and Computational Sciences (RAETCS), pages 1–4, 2018.

[8] Y. Xin, L. Kong, Z. Liu, Y. Chen, Y. Li, H. Zhu, M. Gao, H. Hou, and C. Wang. Machine learning and deep learning methods for cybersecurity. IEEE Access, 6:35365– 35381, 2018.

[9] Neha R. Israni and Anil N. Jaiswal. A survey on various phishing and anti-phishing measures. International journal of engineering research and technology, 4, 2015.

[10] Pingchuan Liu and Teng-Sheng Moh. Content based spam email filtering. pages 218–224, 10 2016.

[11] N. Agrawal and S. Singh. Origin (dynamic blacklisting) based spammer detection and spam mail filtering approach. In 2016 Third International Conference on Digital Information Processing, Data Mining, and Wireless Communications (DIPDMWC), pages 99–104, 2016.

[12] Polamuri, S.R. Stroke detection in the brain using MRI and deep learning models. Multimed Tools Appl (2024). https://doi.org/10.1007/s11042-024-19318-1

[13] S. Patil and S. Dhage. A methodical overview on phishing detection along with an organized way to construct an anti-phishing framework. In 2019 5th International Conference on Advanced Computing Communication Systems (ICACCS), pages 588–593, 2019.

[14] Kumar, Voruganti Naresh, U. Sivaji, Gunipati Kanishka, B. Rupa Devi, A. Suresh, K. Reddy Madhavi, and Syed Thouheed Ahmed. "A Framework For Tweet Classification And Analysis On Social Media Platform Using Federated Learning." Malaysian Journal of Computer Science (2023): 90-98.

[15] Saurabh Saoji. Phishing detection system using visual cryptography, 03 2015.

[16] C. Pham, L. A. T. Nguyen, N. H. Tran, E. Huh, and C. S. Hong. Phishing-aware: A neuro-fuzzy approach for anti-phishing on fog networks. IEEE Transactions on Network and Service Management, 15(3):1076–1089, 2018.
[17] K. S. C. Yong, K. L. Chiew, and C. L. Tan. A survey of the qr code phishing: the current attacks and countermeasures. In 2019 7th International Conference on Smart Computing Communications (ICSCC), pages 1–5, 2019. 54
[18] G. Egozi and R. Verma. Phishing email detection using robust nlp techniques. In 2018 IEEE International Conference on Data Mining Workshops (ICDMW), pages 7–12, 2018.
[19] J. Mao, W. Tian, P. Li, T. Wei, and Z. Liang. Phishing-alarm: Robust and efficient phishing detection via page component similarity. IEEE Access, 5:17020–17030, 2017.
[20] G. J. W. Kathrine, P. M. Praise, A. A. Rose, and E. C. Kalaivani. Variants of phishing attacks and their detection techniques. In 2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI), pages 255–259, 2019.
[21] Muhammet Baykara and Zahit Gurel. Detection of phishing attacks. pages 1–5, 03 2018.
[22] Prof. Gayathri Naidu. A survey on various phishing detection and prevention techniques. International Journal of Engineering and Computer Science, 5(9), May 2016.
[23] Reid G. Smith and Joshua Eckroth. Building ai applications: Yesterday, today, and tomorrow. AI Magazine, 38(1):6–22, Mar. 2017.
[24] Panos Louridas and Christof Ebert. Machine learning. IEEE Software, 33:110–115, 09 2016.
[25] Michael Jordan and T.M. Mitchell. Machine learning: Trends, perspectives, and prospects. Science (New York, N.Y.), 349:255–60, 07 2015.
[26] T. Nathezhtha, D. Sangeetha, and V. Vaidehi. Wc-pad: Web crawlingbased phishing attack detection. In 2019 International Carnahan Conference on Security Technology (ICCST), pages 1–6, 2019.
[27] Aleksandar Milenkoski, Marco Vieira, Samuel Kounev, Alberto Avritzer, and Bryan Payne. Evaluating computer intrusion detection systems: A survey of common practices. ACM Computing Surveys, 48:12:1–, 09 2015.
[28] Chirag N. Modi and Kamatchi Acha. Virtualization layer security challenges and intrusion detection/prevention systems in cloud computing: a comprehensive review. The Journal of Supercomputing, 73(3):1192–1234, Mar 2017.
[29] Eduardo Viegas, Altair Santin, Andre Fanca, Ricardo Jasinski, Volnei Pedroni, and Luiz Soares de Oliveira. Towards an energy-efficient anomaly-based intrusion detection engine for embedded systems. IEEE Transactions on Computers, 66:1–1, Jan 2016. 53.
[30] S. Parekh, D. Parikh, S. Kotak, and S. Sankhe. A new method for detection of phishing websites: Url detection. In 2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT), pages 99–952, 2018.