INTERNATIONAL JOURNAL OF SOFTWARE ENGINEERING AND COMPUTER SYSTEMS
E-ISSN: 2180-0650
VOLUME 10, ISSUE 1, 2024, pp. 75 – 84
DOI: https://doi.org/10.15282/ijsecs.10.1.2024.7.0125

RESEARCH ARTICLE

Performance Analysis of Selected Machine Learning Algorithms in the Detection of Phishing Attacks on Vulnerable Websites.

Fatima Enehezei Usman-Hamza[1] and Adeleke Raheem Ajiboye[2*]
[1]Department of Computer Science, Faculty of Communication and Information Sciences, University of Ilorin, 240001 Ilorin, Nigeria.
[2]School of Mathematics and Computing, Kampala International University, 256 Kasanga, Uganda.

ABSTRACT - Phishing is a type of cyberattack in which the attackers pretend to be trustworthy, to trick individuals into providing sensitive information. Phishing also involves directing web users to fake websites that closely resemble legitimate ones and asking their victims to enter their personal information. It is imperative to restrict all forms of phishing websites or URLs. The significance of ensuring that phishing is prevented cannot be over-emphasized; this is why most software developed to guard the system against phishing usually provides round-the-clock technical support. This study focuses on analysing the performance of some algorithms typically used in machine learning to develop models capable of detecting phishing websites. To achieve this central goal, massive phishing website detector datasets were retrieved from an online open repository, Kaggle. Relevant libraries in Python were explored for pre-processing, uploading, and partitioning of the datasets for training and testing the model. The models were created based on the developed algorithm and were followed by the implementation of all the selected machine-learning techniques. Evaluation of each model reveals that; Random Forest shows the highest accuracy of 96.7%, followed by Support Vector Machine which records 96.4%. However, evaluating the model created using the Naive Bayes Classifier shows the lowest accuracy of 60.5%. The merits of detecting phishing attacks before their occurrence include the prevention of data breaches, protection of reputations, improved cybersecurity awareness and so on. This study has uncovered the strengths of each algorithm in detecting phishing websites. It has also revealed the procedures that need to be followed to mitigate and curb the spread of phishing attacks.

ARTICLE HISTORY
Received : 21 May 2024
Revised : 3 December 2024
Accepted : 12 December 2024
Published : 20 December 2024

KEYWORDS
Phishing attack
Machine learning
Algorithm
Cyber-attack
Classification
Models.

1.0 INTRODUCTION

A phishing attack involves tricking a victim into taking some actions that benefit the attacker [1]. The complexity of these attacks can be detected with the right awareness. There is a significant change in how the networks and the internet have changed our way of communication, the way we learn, work and even play. Networks come in various sizes, ranging from small setups connecting just two computers to larger networks that link millions of devices. The Internet stands as the largest network in existence, providing the services that facilitate our connections and communications with family, friends, work, and other interests.

In recent years, the Internet of Things (IoT) has enabled the connection of anything and everything with a Mac address to the Internet [2]. It has led to a digital disruption in the physical world as we know it; by changing how we use technology. Recently, IoT technology has made it possible to connect several devices such as light bulbs, refrigerators, drones, pet feeders, sensors, smart Televisions and digital set-top boxes, CCTV cameras, automotive systems, and other devices for connecting to the internet [3]. Also, recently, cybersecurity has become a major issue due to our growing reliance on the internet. Today, we can hardly do anything without the internet. Attacks such as phishing is very dangerous for both people and businesses [3]. Hackers employ a sneaky method called phishing to deceive unwary individuals into revealing sensitive information like login details, credit card details, or personal information [4]. These assaults frequently take the shape of fake websites that closely mimic real ones, making it difficult for visitors to tell one from the other. The internet, particularly social media, is now our main source of information dissemination.

***CORRESPONDING AUTHOR | A.R. Ajiboye** | ✉ ajibabdulraheem@gmail.com
© The Author(s) 2024. Published by Universiti Malaysia Pahang Al-Sultan Abdullah Publishing. This is an open access article under the CC BY-NC 4.0 International license

In today's digital age, exploring the internet has become a vital aspect of our daily activities, providing immense convenience and connectivity. However, this increased reliance on online platforms also exposes individuals and organizations to various cybersecurity threats [5]. A website is vulnerable if it lacks certain features that could make it less secure. Phishing attacks, in particular, have emerged as a major concern, targeting unsuspecting users and aiming at stealing sensitive information. It is important to develop robust techniques for the timely identification and prevention of such fraudulent activities as phishing attacks become more sophisticated and frequent. Detecting phishing websites is very challenging and requires an enormous task due to their constantly evolving nature and ability to imitate legitimate websites convincingly. Traditional rule-based approaches and blacklisting techniques seem inadequate in keeping up with the ever-changing landscape of phishing attacks [6]. Consequently, the integration of machine learning algorithms has gained significant attention as a promising solution for accurate and efficient phishing detection.

The use of insufficient phishing datasets coupled with the techniques that have poor classification strength was mostly reported in the earlier studies, this necessitates a further study. This study focuses on implementing algorithms suitable for both classification and regression tasks with the specific objective of unveiling the effectiveness of these algorithms in fitting predictive models from massive phishing datasets. This study focuses on exploring some machine learning algorithms in creating models capable of detecting phishing websites. By leveraging the power of artificial intelligence and data-driven techniques, this study aims to develop a robust and adaptable system capable of identifying phishing websites accurately and in real-time. The proposed research will involve an in-depth analysis of various machine learning algorithms, including but not limited to decision trees, support vector machines (SVM), random forests, and deep learning models such as convolutional neural networks (CNN) and recurrent neural networks (RNN). Both legitimate and phishing websites will be analysed in a comprehensive dataset that includes diverse features and characteristics associated with phishing attacks.

To ensure the authenticity and reliability of the dataset, we consider a wide range of features such as URL structure, domain age, SSL certificates, website content, and visual cues to capture the subtle nuances that distinguish legitimate websites from phishing ones. Furthermore, the evaluation of the developed machine learning models involves testing on a separate set of labelled data, assessing their accuracy, precision, recall, and F1-score. Comparative analysis was later conducted to determine the most effective algorithms in terms of overall performance and ability to handle diverse phishing techniques. The results from this research provide some valuable insights into the applicability and efficacy of machine learning algorithms for detecting phishing websites as they enhance the underlying patterns and characteristics of phishing attacks. The study is capable of improving the cybersecurity landscape and developing proactive measures to safeguard organizations from potential threats.

Conclusively, this research paper aims to contribute to the ongoing efforts in combatting phishing attacks by leveraging the power of machine learning algorithms. The rest of this paper is structured as follows: The next section discusses a secured and unsecured URL; followed by a discussion of selected machine learning algorithms implemented in this study. In Section 4, some related studies reported in the literature were reviewed. Section 5 shows the material and methodology used in this study, while Section 6 discusses the results of evaluation measurements of each model based on some metrics. We conclude this study in Section 7.

2.0 SECURED AND UNSECURED URLs.

URL is an acronym for Uniform Resource Locator. It is a string of characters that serves as an address for locating resources on the internet. A URL provides a unique identifier for a specific webpage, file, or resource and specifies how to access it. A secured website is a site that has been implemented This is usually achievable through encryption protocols such as https that runs on port number 443. This protocol ensures the secured transmission of sensitive information such as passwords, credit card details, and others. These are target information for hackers that present their phishing website to look like a legitimate one.

Unsecured website lacks encryption measures, therefore, data exchanged between the user and the website is sent in plain text, making it susceptible to interception by dubious users. This is capable of posing significant risks, especially when dealing with strictly sensitive information. The kind of network security implementation adopted should take into account the environment and the requirements of the network. While the security of data is paramount, it must also allow for the best quality of service. Securing a network, therefore, requires using relevant protocols, technologies, infrastructures, and techniques to ensure data security and mitigate potential threats. Today, some external network security threats are spread over the internet, resulting in havoc. The most common external threats to networks include:

i. *Viruses, and Trojan horses* – Both are malicious software and a type of malware that attaches itself to a legitimate program. Trojan disguises as a harmless program to trick users into installing it.
ii. *Spyware and adware* – This belongs to a category of software unknowingly installed on a user's device with the sole aim of secretly harvesting sensitive information about the user. Both spyware and adware can negatively impact system performance and compromise user's privacy.
iii. *Zero-day attacks*- This is sometimes referred to as Zero-hour attacks. It is a type of cyber-attack that targets a software vulnerability that is unknown to the software vendor or the general public at the time of the attack and since the vulnerability is not yet patched by the developer, attackers exploit it before a fix is available.
iv. *Hacker*s – An attacker in this category is an IT-knowledgeable individual or group who uses their technical knowledge and skills to gain unauthorised access to systems, networks or data. Their motivations and approaches can vary widely, leading to different categories of hackers.
v. *Denial of service attacks* – This form of attack was designed to slow or crash applications on a network device. It is an attempt to make a computer system, service, or network unavailable to its intended users. This is achieved by overwhelming the target with a flood of malicious traffic or requests.
vi. *Data interception and theft* – This is an attack designed to intrude into the organization's privacy, to capture or extract data while it is being transmitted over a network. In the process, malicious actors intercept communications or hack into systems to steal sensitive or confidential information.
vii. *Identity theft* – This occurs when someone illegally obtains and uses another person's information without permission. Attacks in this category target credentials required for the login of a user to gain access to private data, usually to commit fraud.

The internal threats illustrated in Figure 1 should also be considered. Several studies have shown that data breaches mostly happen because of internal users within the network [7, 8]. Sometimes, lost or stolen devices and accidental misuse might be the fault of employees. In the business environment, so many employees are not trustworthy. The recently evolving Bring Your Device (BYOD) strategy, which is a policy that allows an individual to bring their devices makes corporate data to be much more vulnerable. Phishing is about using a tricky attempt to retrieve sensitive data [9], It is very important to put the right policy in place to curb the process; while creating a security policy, it is crucial to ensure that the security threats from both external and internal are well addressed.

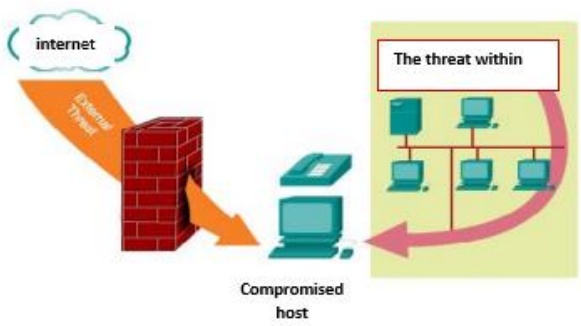

Figure 1. The threat to a network

Some key factors such as Vulnerabilities (V), Privileged users (P), Risky behaviours (R), Detection capability (D), Trust level (T), and Motivation (M) are variables capable of influencing the internal threat. Based on how these factors interact with one another. For instance, there is the possibility of an increased threat likelihood with variables: V.P.R.and M, while the threat impact might be inversely proportional to D and T. These variables can be combined into a function:

$$T_i = \frac{(V + P + R) \times M}{D \times T} \tag{1}$$

Since each variable may not contribute equally to the internal threat, it becomes pertinent to assign weight to each variable. The internal threat as illustrated in equation (1) becomes:

$$T_i = \frac{(w_1 V + w_2 P + w_3 R) \times w_4 M}{w_5 D \times w_6 T} \quad (2)$$

3.0 THE MACHINE LEARNING ALGORITHMS

Machine learning algorithms encompass various techniques that are crucial for predictive analytics. One of these techniques is Logistic Regression, which is used to model the relationship between one or more independent variables and a dependent variable that takes two values, such as 0 and 1, yes or no, true or false. It is specifically designed for classification tasks, and the transformation via the sigmoid activation function enables it to model complex and non-linear decision boundaries. One of its advantages is its simplicity, as it can be easily interpreted and is highly applicable for predicting binary outcomes. Another widely used technique is K-Nearest Neighbour (K-NN), which is suitable for both classification and regression tasks. Although it does not explicitly learn a model, it predicts based on the principle of similarity or proximity in the feature space. Its flexibility in distance metrics and reliance on the immediate neighbourhood data points make it a very powerful tool for specific tasks. K-NN uses distance metrics such as Euclidean, Manhattan, and Minkowski for effective data distribution. However, it has limitations, including sensitivity to outliers and the requirement to store the entire dataset, which can be computationally expensive.

Similarly, Support Vector Machine (SVM) is a versatile technique that can be used for both classification and regression tasks. Originally designed for classification, SVM employs the kernel trick to map non-linearly separable data into a higher-dimensional space. Popular kernels include Polynomial, Radial Basis Function, and Sigmoid. This technique is effective in high-dimensional spaces, robust to outliers near decision boundaries, and less prone to overfitting. However, it is sensitive to hyperparameters, and feature scaling is mandatory for optimal performance, as it is sensitive to the range of feature values. The Naïve Bayes Classifier is based on the concept of probability and is particularly suitable for classification tasks. This classifier computes the probabilities for each class and assigns the data point to the class with the highest posterior probability. It is computationally efficient regardless of the dataset size, including text data, and its probabilistic approach makes its results easy to understand. Despite these advantages, Naïve Bayes faces limitations, such as the zero-frequency problem, which assigns zero probability to a missing feature category. It is also unsuitable for datasets with highly correlated attributes or complex relationships.

Another commonly used method is the Decision Tree, which is applicable to both classification and regression tasks. This technique models decisions in a tree-like structure by splitting datasets into subsets based on the most significant features at each node. Decision trees are easy to trace, interpret, and can handle both numeric and categorical data, as well as non-linear relationships. However, they can capture noise and overfit easily, reducing generalizability. Since decision trees use a greedy approach to split data at each node, they may not always achieve a globally optimal solution. Lastly, the Random Forest algorithm, an ensemble learning method, is also suitable for both regression and classification tasks. By combining multiple decision trees through bootstrap sampling, feature randomness, and ensemble decision-making, Random Forest improves predictive accuracy. This technique is robust to outliers, can tolerate missing data, and reduces overfitting. However, its complexity and the difficulty of interpreting its results can sometimes be drawbacks compared to simpler models. It is also not ideal for very high-dimensional sparse data. These machine learning algorithms highlight the range of tools available, each with unique strengths and limitations, for tackling various predictive tasks.

4.0 RELATED WORKS

Several studies have reported the detection of phishing attacks. For instance, [10] proposed a system for the detection of phishing websites using machine learning algorithms. The page-based and lexical feature extractions of URLs were analysed in the study. and later applied to form a database of feature values. The knowledge-based approach was used to mine the database with the implementation of different machine-learning methods. The evaluation of the various classifying algorithms was based on the data mining workbench using tools such as WEKA and MATLAB. The focus of the study was, however, limited to four machine learning techniques. The J48 Decision Tree technique was reported in the study to have shown the highest success rate relative to other selected classifying algorithms.

In the study proposed in [11], a framework was developed to analyse some features and abnormal behaviours of URLs using the technique of machine learning. The study also proposed the use of a rule-set to detect phishing, specifically Random Forest (RF) and SVM were used for the detection. The study unveils the suitability of the classification model created with the sole task of detecting phishing URLs. To have a detailed insight into the techniques used for phishing detection, the study proposed in [12], reviews the tools in the detection of phishing attacks on web pages. The study focused on extraction and implementation of a rule-based approach for phishing detection. The study also explored phishing websites retrieved from phishtank.com, which is a community-based phishing verification system, where suspicious websites are identified for user voting as a method of phishing detection and verification. Legitimate Internet banking websites used for the research were obtained from various web page directory services.

Using the techniques of machine learning for phishing attack detection has gained prominence in recent years. The study reported in [13] shows that better detection can be achieved by using Artificial Neural Network (ANN), as it finds this technique more accurate than the linear regression and support vector machine implemented in the course of the study. To reduce the potential risk of phishing attacks, the study further proposed several approaches for phishing detection, namely: rule-based, white and blacklist, heuristic, and the use of hybrid. A secure mechanism was proposed for phishing attacks and reported in [14]. The study's objective was to evaluate the degree to which phishing attacks could prove to be harmful. The study further offers some solutions that could protect the system against phishing attacks. Similarly, several reviews reported in the literature regarding some ways through which phishing attacks can be prevented are discussed in [15, 16].

The use of a hybrid approach was proposed for detecting phishing in a website [17]. The hybrid were essentially classification algorithms designed to identify different types of phishing web pages. The approach was reported to be effective. Also, a preventive technique against phishing attacks on networks proposed in [18], the study identified the use of URLs with an IP address, attributes, domain name, and link text as a preventive measure against phishing attacks. This helps to check and monitor each mail received against the internal configuration of the network server to verify the security standard, similarity index with the already blacklisted information in its internal configurations.

The above reviews have shown earlier attempts to use machine learning techniques for phishing detection, however, this leaves some gaps that still need to be filled. Fitting of models from an insufficient dataset can cause underfitting and implementing an algorithm that has poor classification strength is capable of making such models less accurate. The proposed study fits its models from over 11,000 datasets retrieved from Kaggle, an online open repository [19]. This study also shows the implementation of more machine learning algorithms in conformance with the algorithm presented in Figure 3 using Python codes in a Jupyter environment.

5.0 METHODS AND MATERIALS

5.1. Importing and Cleaning of Dataset

The phishing website datasets were imported to the Jupyter environment for proper analysis. The library to achieve this is shown in Figure 2. An exploratory data analysis to gain insights into the dataset was carried out. The dataset comprised 11,054 records and 32 features, as revealed by the shape of the data frame. The exploration of the datasets involves examining the structure of the data, and shapes The dataset is then visualized using a heatmap correlation, that reveals the patterns or trends in the dataset. This was necessary to find the distribution of the data and how the features are related to each other.

The dataset has 31 parameters or input attributes, the last column is the class label for identifying a phishing website or otherwise, it consists of 1 or -1. This is a linear problem and the data involved are mainly Boolean. The dataset was loaded into the data frame using the *Pandas* library as shown in Figure 2. The datasets were properly put into shape for compatibility during algorithm implementation. This paved the way for fitting models from the data through training in the Jupyter environment where the actual coding in Python occurred.

```
[1]: import pandas as pd
     df = pd.read_csv('phishing.csv')
     df.shape

(11054, 32)
```

Figure 2. Importing the dataset using Pandas library

The algorithm represented in Figure 3 shows the steps required to fit the prediction model from the dataset based on the selected modelling techniques.

i.	Import the required libraries: pandas, sklearn
ii.	Import a modelling technique from sklearn
iii.	Read the dataset using pandas
iv.	Split the dataset into input attributes and labels.
v.	X ← input attributes
vi.	Y ← label attribute
vii.	Model ← modelling technique
viii.	model.fit (X, Y)
ix.	Predict the label using the input attributes
x.	Evaluate the performance of the prediction model
xi.	Repeat the steps after importing another technique in step ii.

Figure 3. Algorithm for creating a model for prediction

The steps as represented in Figure 3 require the importation of libraries, specifically pandas and scikit learn. Pandas is required to import the dataset into the environment where analysis is to take place, in this case, Jupiter Notebook. Sklearn is the library that holds most of the modelling techniques required to explore the imported data. To unveil the relevance of each attribute in the dataset for prediction, the heat map of the dataset was determined. A heat map is a data visualization tool that shows the intensity of data values with the help of colour variations. The chart is about the feature importance of the attributes. Its importance can be revealed in pattern variation, simplification of data and decision-making. The map is crucial for transforming complex data into visual insights. The secured hypertext transfer protocol (*https)*, which runs on port number 443 is one of the features used for the model development in this study; https is essential for protecting user data, maintaining trust, and meeting modern security standards on the web.

5.2 Models Training and Testing

Supervised machine learning was the approach used to train the dataset because of the actions of prediction from the dataset with features-label pairs. Machine learning models were built from these feature-label pairs comprising training and testing sets. The splitting of datasets was achieved through the use of the *sklearn* library. Importation of the *train_test_split* from *model_selesction* in the *sklearn* library partitions the datasets into 80% for training, while others were reserved for testing. The fitting of the model from the data reported in [20], has shown that such a splitting ratio makes the entire data to be well-represented during the training process. The central goal was to achieve accurate predictions for new, and unseen data. The models were subsequently fitted from the

datasets using the following algorithms: Logistic Regression, K-Nearest Neighbours, Support Vector Machine, Naive Bayes, Decision Tree, and Random Forest. All these algorithms are contained in the *sklearn* library which is well explored in this study.

5.3 Model Evaluation

Implementation of each of the machine learning algorithms led to the creation of a classification model. Each model created is capable of detecting a legitimate or phishing website. Each model created was evaluated for correctness. The evaluation was done based on some metrics which include: Accuracy, f1_score, Recall, and Precision as shown in Table 1. Similar metrics were also used in [21]; they are standard machine learning evaluation metrics. Each metric conforms to the following formulas:

$$Accuracy = \frac{TP + TN}{P + N} \tag{3}$$

$$F1_{score} = \frac{TP}{TP + \frac{1}{2FP} + FN} \tag{4}$$

$$Recall = \frac{TP}{TP + FP} \tag{5}$$

$$Precision = \frac{TP}{TP + FP} \tag{6}$$

where: TP represents true positive, TN represents true negative, FN represents false negative, FP represents false positive, P represents positive, and N represents negative.

6.0 RESULTS AND DISCUSSION

Measuring the performance of a machine learning model is typically achieved or determined using several metrics and this is usually problem-based. The problem being solved may be classification, regression or clustering. For instance, accuracy, precision, recall, and f1 score are suitable measurements for classification problems. Also, the confusion matrix is one of the techniques that can be used to measure classification performance. The confusion matrix is a table that describes the performance of a classification model. The confusion matrix table usually shows the actual versus predicted values for each class. In the current study, the performance of the model created for the detection of phishing websites is measured using the metrics shown in Table 1. The chart in Figure 4 shows the pictorial representation of how each technique implemented performs concerning accuracy, f1_score, Recall and Precision. F1_score is the harmonic mean of precision and recall. The value of f1_score is close to 1 in Table 1, it is an indication of a good balance between precision and recall.

Table 1. Model performance based on some evaluation metrics

	ML Model	Accuracy	f1_score	Recall	Precision
0	Logistic Regression	0.934	0.941	0.943	0.927
1	K-Nearest Neighbours	0.956	0.961	0.991	0.989
2	Support Vector Machine	0.964	0.968	0.980	0.965
3	Naïve Bayes Classifier	0.605	0.454	0.292	0.997
4	Decision Tree	0.961	0.965	0.991	0.993
5	Random Forest	0.967	0.970	0.992	0.991

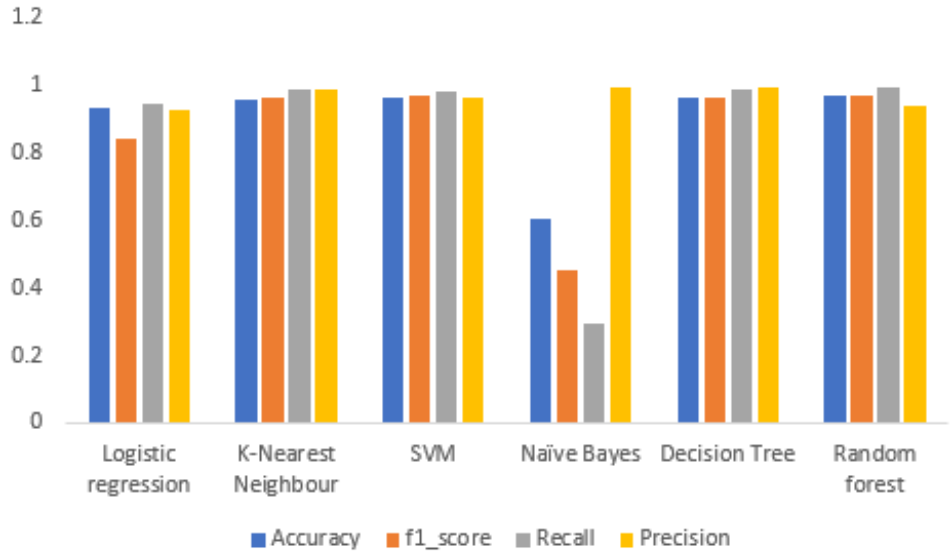

Figure 4. Graphical representations of the output results of the models in terms of accuracy, f-score, recall, and precision

Evaluation of each model created reveals some accuracy values. The results indicate that the model created using Random Forest is the most accurate among the six machine learning algorithms implemented in this study with 96.7%, followed by Support Vector Machine (SVM), which is 96.4% accurate. Table 1 shows the details. The results of F1_score show that Random Forest records the highest value of 0.970, this is closely followed by Support Vector Machine which has a value of 0.968. As for Recall, Random Forest also maintains consistency by recording the highest value of 0.992. An earlier survey of Random Forest performance [22] has shown the effectiveness of the technique, the result of this study also reveals a similar pattern.

Random Forest is an extended version of a decision tree that uses multiple classifiers to predict future instances, increasing the accuracy and correctness of the predictions. The results pattern slightly changed as the highest value for this metric follows the order Random Forest, Decision Tree (DT) and K-Nearest Neighbour (KNN). Naïve Bayes Classifiers have the highest precision in this study, closely followed by Decision Tree. However, Neive Bayes performs poorly in other metrics considered.

Most of the related studies reported in the literature implement 1 or 2 machine learning algorithms for the detection of phishing websites. The use of the hybrid method reported in [17], implements two separate algorithms. The implementation of 6 machine learning algorithms unveils the strengths and weaknesses of more algorithms for phishing detection within the domain of machine learning. The massive data explored in the course of fitting each classification model in this study stabilizes the model created and shows how such datasets could enhance learning as revealed in all the results generated.

7.0 CONCLUSIONS

In this research, we utilized various machine-learning algorithms to identify phishing URLs. Throughout the study, we tested the effectiveness of the selected machine learning algorithms. We examined specific traits and abnormal URL behaviours, and we later applied machine-learning techniques to achieve the study's goals. It's important to note that relying on machine learning models fitted with a limited dataset can lead to inaccuracies and such results could be misleading. In the course of this study, models were fitted from over 11,000 records of high dimensions of relevant features. Also in this study, six machine-learning algorithms were implemented and the model created through each technique was subsequently evaluated for correctness. The simulation of each model created using an untrained dataset as input shows the consistency of the results. This study has revealed the effectiveness and practicality of detecting phishing URLs through the use of machine learning predictive models.

Generally, machine learning allows computers to learn from data and improve their performance on tasks over time. Fitting each model from the same dataset with all the six algorithms implemented in this study has provided a deeper understanding of the capabilities of each technique. To improve the effectiveness of the phishing URL detection model in future studies, we recommend using deep learning techniques to enhance feature learning. Although; deep learning is a subset of machine learning, it can automatically discover intricate patterns in a large amount of data. This study has contributed to the field of machine learning by formulating and implementing the algorithm through which the model can be fitted from the phishing detector dataset; it has also contributed significantly to the literature.

ACKNOWLEDGEMENTS

We would like to express our profound gratitude to the reviewers of this research article for their invaluable input, which has significantly enhanced the quality of this work, we are indeed very grateful.

AUTHORS CONTRIBUTION

Fatima Enehezei Usman-Hamza (Retrieval of data from Kaggle, Data Pre-Processing; Writing & Editing)

Adeleke Raheem Ajiboye (Implementation of the Machine Learning Algorithms using Python, Writing & Editing)

CONFLICT OF INTEREST

The authors declare no conflicts of interest

REFERENCES

[1] A. Athulya and K. Praveen, "Towards the detection of phishing attacks," in *2020 4th international conference on trends in electronics and informatics (ICOEI)(48184)*, 2020, pp. 337-343.

[2] K. Karimi and G. Atkinson, "What the Internet of Things (IoT) needs to become a reality," *White Paper, FreeScale and ARM*, pp. 1-16, 2013.

[3] K. Kimani, V. Oduol, and K. Langat, "Cyber security challenges for IoT-based smart grid networks," *International journal of critical infrastructure protection*, vol. 25, pp. 36-49, 2019.

[4] H. Ibrahim, "A Review on the Mechanism Mitigating and Eliminating Internet Crimes using Modern Technologies: Mitigating Internet crimes using modern technologies," *Wasit Journal of Computer and Mathematics Science*, vol. 1, pp. 50-68, 2022.

[5] Ö. Aslan, S. S. Aktuğ, M. Ozkan-Okay, A. A. Yilmaz, and E. Akin, "A comprehensive review of cyber security vulnerabilities, threats, attacks, and solutions," *Electronics*, vol. 12, p. 1333, 2023.

[6] I. H. Sarker, H. Janicke, M. A. Ferrag, and A. Abuadbba, "Multi-aspect rule-based AI: Methods, taxonomy, challenges and directions toward automation, intelligence and transparent cybersecurity modeling for critical infrastructures," *Internet of Things*, p. 101110, 2024.

[7] D. M. Cappelli, A. P. Moore, and R. F. Trzeciak, The CERT guide to insider threats: how to prevent, detect, and respond to information technology crimes (Theft, Sabotage, Fraud): Addison-Wesley, 2012.

[8] I. Confente, G. G. Siciliano, B. Gaudenzi, and M. Eickhoff, "Effects of data breaches from user-generated content: A corporate reputation analysis," *European Management Journal*, vol. 37, pp. 492-504, 2019.

[9] A. Basit, M. Zafar, X. Liu, A. R. Javed, Z. Jalil, and K. Kifayat, "A comprehensive survey of AI-enabled phishing attacks detection techniques," *Telecommunication Systems*, vol. 76, pp. 139-154, 2021.

[10] J. James, L. Sandhya, and C. Thomas, "Detection of phishing URLs using machine learning techniques," in *2013 international conference on control communication and computing (ICCC)*, 2013, pp. 304-309.

[11] C. Do Xuan, H. Thanh, and N. T. Lam, "Optimization of network traffic anomaly detection using machine learning," *International Journal of Electrical & Computer Engineering (2088-8708)*, vol. 11, 2021.

[12] T. O. Ojewumi, G. Ogunleye, B. Oguntunde, O. Folorunsho, S. Fashoto, and N. Ogbu, "Performance evaluation of machine learning tools for detection of phishing attacks on web pages," *Scientific African*, vol. 16, p. e01165, 2022.

[13] F. Salahdine, Z. El Mrabet, and N. Kaabouch, "Phishing attacks detection a machine learning-based approach," in *2021 IEEE 12th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)*, 2021, pp. 0250-0255.

[14] G. Mohamed, J. Visumathi, M. Mahdal, J. Anand, and M. Elangovan, "An effective and secure mechanism for phishing attacks using a machine learning approach," *Processes*, vol. 10, p. 1356, 2022.

[15] A. Sadiq, M. Anwar, R. A. Butt, F. Masud, M. K. Shahzad, S. Naseem, *et al.*, "A review of phishing attacks and countermeasures for internet of things-based smart business applications in industry 4.0," *Human behavior and emerging technologies*, vol. 3, pp. 854-864, 2021.

[16] M. F. Ansari, P. K. Sharma, and B. Dash, "Prevention of phishing attacks using AI-based Cybersecurity Awareness Training," *Prevention*, vol. 3, 2022.

[17] M. Dadkhah, S. Shamshirband, and A. W. Abdul Wahab, "A hybrid approach for phishing website detection," *The Electronic Library*, vol. 34, pp. 927-944, 2016.

[18] M. Adil, R. Khan, and M. A. N. U. Ghani, "Preventive techniques of phishing attacks in networks," in *2020 3rd International Conference on Advancements in Computational Sciences (ICACS)*, 2020, pp. 1-8.

[19] *Phishing website Detector*. Available: Retrieved from: https://www.kaggle.com/datasets/eswarchandt/phishing-website-detector

[20] J. Seidu, A. Ewusi, JSY Kuma, YY Ziggah, and HJ Voigt, "Impact of data partitioning in groundwater level prediction using artificial neural network for multiple wells," *International Journal of River Basin Management* vol. 21 (4), 2023.

[21] M. O. Diaz Jr, "A domain-specific evaluation of the performance of selected web-based sentiment analysis platforms," *International Journal of Software Engineering and Computer Systems*, vol. 9, pp. 01-09, 2023.

[22] A. B. Shaik and S. Srinivasan, "A brief survey on random forest ensembles in classification model," in *International Conference on Innovative Computing and Communications: Proceedings of ICICC 2018, Volume 2*, 2019, pp. 253-260.