Journal of Al-Qadisiyah for Computer Science and Mathematics Vol. 16(3) 2024, pp Comp. 35–43

Available online at www.qu.edu.iq/journalcm
JOURNAL OF AL-QADISIYAH FOR COMPUTER SCIENCE AND MATHEMATICS
ISSN:2521-3504(online) ISSN:2074-0204(print)

Models in Review for the Analysis of Phishing Website URLs

Ali Salam Al-jaberi [a], Sura Fadhil Rahman [b] , Ihsan Faisal Raheem [c]

[a] *College of Computer Sciences and Information Technology , University of Al-Qadisiyah , Al-Qadisiyah, Iraq . Email:com.post02@qu.edu.iq*

[b] *Computer Techniques Engineering , Imam AL-Kadhun College , Al-Qadisiyah , Iraq . Email: surafadhilrahman@gmail.com*

[c] *Nizam college , Osmania University , Al-Qadisiyah , Iraq . Email: ihsanfaisal500@gmail.com*

ARTICLE INFO

Article history:
Received: 20 /07/2024
Rrevised form: 27 /08/2024
Accepted : 29 /08/2024
Available online: 30 /09/2024

Keywords:

Machine Learning, Phishing, Websites, XGBoost, URLs, Cybersecurity.

ABSTRACT

In this paper, we compare our method on DEFRAUD with current online API services and the state-of-the-art machine learning model for defending against phishing websites. Rule-based methods, in nature traditional such rules tend to get outdated quickly and are not capable of tracing new tactics used by the malicious ware every second. Over time, new proactive approaches enabled by machine learning (ML) have become more important as these solutions are flexible and adaptable in their ability to scan through modern data breaches for patterns from millions of datasets. In this study, we explore various machine learning algorithms: Logistic Regression (LR), K-Nearest Neighbors (KNN), Decision Trees (DT) Random Forest (RF), Support Vector Classifiers (SVC) and xgBoost for phishing website detection. Ensemble Methods like Random Forest, XGBoost have better accuracy/precision/recall. Metrics. While XGBoost is resource hungry, it is well known for out of the box support with huge data dimensions as well deep learning framework and avoiding overfitting. The study underscores the importance of integrating machine-learning models into practical cybersecurity applications. Future research should focus on improving these models and expanding their application across different domains to enhance cybersecurity defenses.

https://doi.org/ 10.29304/jqcsm.2024.16.31641

1. INTRODUCTION

The internet is essential for many facets of daily life. A network of computers connected by phone, fiber optic, wireless, satellite, and other telecommunications channels is called the Internet. This computer network is worldwide. Computers are machines that store data that may be accessed online. They are often referred to as hosts or servers. They used IP-TCP, or the Internet Protocol/Transmission Control Protocol, for communication. Numerous academic institutions, research centers, and other groups control the Internet instead of the government, which is not acknowledged as its owner. Phishing websites are a serious threat to cybersecurity in the contemporary digital environment. This risk factor has definitely tripled with increased usage in e-commerce, online banking, and social networking for such fraudulent websites. They are used to impersonate reliable

*Corresponding author

Email addresses:

Communicated by 'sub etitor'

businesses in order to obtain personal information of clients directly. The results may be bad, ranging from the user's privacy being compromised to monetary losses and data breaches [1, 2].

This can be complemented by the use of machine learning in this continuous battle against such phishing attacks. Traditional rule-based solutions often fail, in real-time threat scenarios, to keep up with the strategy of the attackers as they change constantly. In contrast, being able to recognize patterns and extract data from huge databases, machine learning provides proactive prevention of phishing attacks [3]. Additionally, due to its adaptability, it also foresees and provides defenses against new strategies of phishing.

Machine learning techniques have been applied to a number of studies in the identification of phishing attempts. Such approaches increase the accuracy and efficiency of phishing detection systems by using machine learning techniques, NLP, and predictive analytics in tandem. Existing algorithms of machine learning need to be drilled deeper and evaluated if they have to remain relevant in the face of the methods of phishing that are fast evolving [4,5].

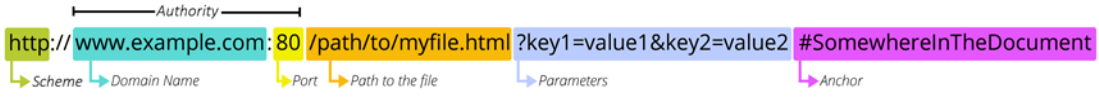
In this digital age, security and integrity of user data become very vital. Any development in Internet security is followed by a hack that tries to exploit the vulnerabilities that defenses are built against. The complexity these phishing attacks come in calls for improved techniques in their detection and prevention. Machine learning can be a feasible solution in this case. Because it builds based on continuous iterative updates and improvements from a variety of sources, it can therefore provide dynamic defense against the adaptive nature of attacks. The seamless inclusion of computational powers with human intuition shall be the future of cybersecurity. Beyond proving our commitment to protecting our clients from the subliminal yet extraordinarily impactful threats of phishing, this research is conducted for possible applications of machine learning and fosters confidence in our progressively digitalized community. Phishing attacks are becoming a serious cyber threat that puts people, businesses, and their personal data at risk. Despite improvements in cybersecurity defense, these assaults persist, causing significant financial losses, harm to one's reputation, and compromising data security. As the volume, complexity, and efficacy of phishing efforts rise, mitigation methods and prompt response are required to guard against this ubiquitous cyber threat.

2. Natural Language Processing (NLP)

This has been the ultimate goal of a computer-science field called Natural Language Processing: to give machines the ability to understand and communicate in human language. Contrasted, AI-driven techniques empower NLP to give robots the capacity for understanding, interpretation, and generation of human languages. In this respect, computational linguistics by rule-based approaches has already been combined with machine learning and statistical analysis with deep learning models. These technologies will now allow computers to perceive meaning, emotions, and purpose for textual and auditory data. Conversely, NLP powers technologies ranging from voice-activated answers to language translation and real-time summaries across enormous amounts of data. Numerous consumer goods, such as voice-activated GPS units, digital assistants, speech-to-text tools, and chatbots for customer support, use natural language processing, or NLP. Moreover, NLP has become increasingly prevalent in commercial solutions to increase worker productivity, expedite information retrieval, and improve operational effectiveness [8].

3. Uniform Resource Locator (URL)

An online resource's distinctive identification is its Uniform Resource Locator, or URL. Browsers to access a variety of resources, notably CSS files, pictures, and HTML pages, use uRLs. Every legitimate URL should, in theory, point to a different resource. Nevertheless, there are certain exceptions, such as URLs referring to removed or migrated resources. It is the server owner's duty to keep the resource and URL up to date because the web server is in charge of administering both. There are several components to a URL; some are required, while others are not [9]. The primary elements are shown at the URL below, and detailed descriptions of each are provided in the sections that follow



4. ASSOCIATED MODELS AND TECHNIQUES

4.1 Logistic Regression

It uses a logistic function to process the linear combination of input data, x, to a probability, p. Logistic regression is used to provide an estimate of the probability of an event in this case, phishing based on a linear combination of input variables [10].

$$p(x) = \frac{1}{1 + e^{-(x-u)/s}} \quad (1)$$

Where:

p logistic function***; u*** is location parameter; ***s*** is the scale parameter for the sigmoid function; ***x*** input variables.

4.2 K-Nearest Neighbors

Thus, the distances between the target sample and the samples near to it are calculated to find the most common class label among the K nearest neighbors. KNN is easy to use and flexible, and is applied [11].

Given two feature vectors with numeric values:

A = (a1, a2, ..., an) and B = (b1, b2, ... , bn)

The formula below is a distance measure known as Euclidean Distance where Ri is the range of the i Component:

Where:

Ri is the range of the ***i*** Component; I is the Component; ***a, b*** are vectors; ***d*** is the distances.

$$d = \sqrt{\sum_{i=1}^{n} \frac{(a_i - b_i)^2}{R_i^2}} = \sqrt{\frac{(a_1 - b_1)^2}{R_1^2} + \frac{(a_2 - b_2)^2}{R_2^2} + \ldots + \frac{(a_n - b_n)^2}{R_n^2}}$$

4.3 Decision Tree

Using hierarchical judgments, decision tree algorithms categorize events into a tree-like structure. While each leaf node represents a class label, each internal node indicates a choice made in response to a particular property. Decision trees can handle both continuous and categorical data, and they are easy to understand. They could require assistance with complicated interactions, though, as they are prone to overfitting [12]. Decision trees employ a series of hierarchical decisions to partition the feature space. During classification, based on the values of the features, it walks from the root to a leaf node in the tree. The techniques followed by the decision tree are a variant of the ID3. In accordance with ID3, when entropy is zero, a branch will be the leaf node; otherwise, it will have an entropy more than zero and so needs further splitting. The possibility of a certain outcome is provided by entropy, and this may be utilized to ascertain the node's branching method. The usage of log functions in entropy makes it a mathematical concept.

$$\text{Entropy} = \sum_{i=1}^{c} -p_i * log_2(p_i)$$

Where: ***pi*** represents relative frequency and ***c*** represents the number of classes.

4.4 Random Forest

The Random Forest ensemble learning technique uses many selfsame decision trees, but each is trained on a different subset of features and data, randomly chosen. The ultimate classification is determined by summing the forecasts of all the various trees. Random Forest reduces overfitting and enhances generalization. It is resistant to noise in the data and supports high-dimensional feature spaces. However, it can be computationally expensive. RF deals with regression problems using mean squared error to calculate the distance of each node from the correct value, hence helping in selecting the branch that is going to benefit the forest most.

$$\mathrm{MSE} = \frac{1}{N}\sum_{i=1}^{N}(f_i - y_i)^2$$

Where:

N is the data points; ***fi*** is the value returned by model; ***yi*** is the actual value of Data point.

4.5 Support Vector Classifier

The SVC is a model for binary classification that segregates instances along the maximum margin hyperplane. That means an optimal hyperplane is constructed in higher-dimensional space obtained by projection of data to maximize the distance between the two classes. By kernel functions, it has the capabilities for handling high-dimensional data efficiently and capturing complex linkages. However, this could be affected by the choice of hyper parameters, and it may further exhibit long training times for large datasets. The optimal hyperplane that will maximize the distance between the two classes is found using the SVC. The decision function of classification is provided by the:

4.6 Linear Support Vector Classifier

Linear SVC is a variant of SVC that uses a linear kernel function. It functions exceptionally well for data that can be partitioned linearly. Linear SVC is a good way to scale large datasets and has good generalization capabilities [15]. Conversely, nonlinearly separable data may need support. When comparing linear SVC to support vector classifiers, the latter is frequently faster at convergent large datasets. SVC reduces the squared hinge loss, whereas linear SVC reduces the regular hinge loss. LSVC utilizes one versus rest, but the SVC classifier uses one vs one multiclass reduction. Linear SVC is a classifier that is used for linear support vector classification. It's a lot alike the Support Vector Classification (SVC) with a linear kernel, but it's made in terms of loglinear rather than the implementation of libsvm. This makes it possible for us to have a greater variety of penalties and loss functions to choose from, and it should be smoothly adaptable to datasets with large numbers of samples. Some of them are as follows:

a) Penalty: The penalty imposed for misclassification is the subject of the support vector machine's problem. LinearSVC gives you the option to specify the type of norm that is being penalized. The ‘l2’ penalty is the general one for SVC. The ‘l1’ is the coefficient being a vector that is very sparse.
b) Loss Function: The loss function may be specified. The 'hinge' loss is a standard SVM loss (used for example by the SVC class) and the 'squared hinge' loss is the square of the hinge loss.
c) Doule or Primal Optimization Problems: You are free to select the algorithm that will be used to solve the dual or primal optimization problem. Use the dual=False when n_features < n_samples case, naturally.
d) Regularization Parameter : The higher the value of the regularization, the lower the value of the coefficient C. Must be strictly positive. If
e) Multi-class Strategy: Sets the multi-class strategy for y being defined as more than two classes if any.

LinearSVC is a smart tool that tries to come up with a hyperplane in a space of n dimensions which is a clear and evident classification of the data points. Among the countless hyperplanes that are available to us, if we want to allocate the point data into two classes, there are countless hyperplanes that we can choose. The SVM algorithm's goal is to find the plane with the maximum margin, which is the largest gap between data of each class.

The distance between a data point x_i and the decision boundary can be calculated for Linear SVC:

$$\hat{y} = \begin{cases} 1 & : w^T x + b \geq 0 \\ 0 & : w^T x + b < 0 \end{cases} \quad (5)$$

Where:

$\boldsymbol{w^T}$ is the weight vector ; **b** is the parameter ; $\hat{y}$ *is* linear hyperplane ; **x** is the parameter.

4.7 XGBoost

Extreme Gradient Boosting, or XGBoost, is a distributed, scalable gradient-boosted decision tree (GBDT) machine learning system. It is the greatest machine-learning tool for regression, classification, and ranking problems and offers parallel tree boosting [16].

Phishing detection is a supervised learning problem where a target variable (yi) is predicted using training data (xi). The model training examples are fed by pairs (x1, y1), (x2, y2), (xn,yn) that include the extracted feature vector x and the associated tag y, which is either 1 for phishing websites or 0 for authentic websites. Finding specific dataset criteria that the model may use to assess the legality of the URL in the future is the main goal of the study. In this case, each parameter becomes a tree and raises the threshold for making a choice. The prediction may improve noticeably by integrating and improving these trees, even though their performance may not be as planned. XGBOOST uses training data xi to predict the target variable yi repeatedly until the model's parameters are improved. The proposed phishing detection model may be represented mathematically as follows:

$$\hat{y}_i = \phi(X_i) = \sum_{k=1}^{K} f_k(x_i), f \in F$$

Where:

K describes the number of trees and **f**, the function in the function space of **F**.

4.8 NAIVE BAYES

The probability-based classifier Naive Bayes is based on the Bayes theorem and the feature independence condition. With the input attributes, it computes the posterior probability of a class label. Naïve Bayes classifiers can handle excellent-dimensional feature spaces and need little in the way of training data, while having high processing efficiency [17]. However, because of the independence assumption, they could oversimplify the relationships between the characteristics.

These models and techniques have been used in earlier studies on phishing website identification, with differing degrees of effectiveness. For instance, Machan used URL characteristics and logistic regression to identify phishing websites [18]. Website properties are analyzed for detection using KNN and decision tree algorithms [19]. Using URL characteristics and website content, Random Forest can recognize phishing websites. The use of several machine learning models and techniques for phishing website identification is highlighted in the research study. Every model or approach has advantages and disadvantages as well as pertinent research to back it up. The Bayes model is easy to establish and works incredibly well when dealing with large datasets. The Bayes model can be trained to outperform many sophisticated classification methods. In the Bayes theorem, the posterior probability p(c/x) is calculated out of p(c), p(x/c) and p(x).

$$p\left(\frac{c}{X}\right) = \frac{\mathrm{p}\left(\frac{x}{c}\right)\mathrm{p}(c)}{2p(x)} \quad (7)$$

Where:

P(C) and ***P(X)*** are the prior probability of class and predictor, respectively; ***P(x/c)*** is the Probability of the predictor given class; ***P(c/x)*** is the posterior probability of class ***C*** given predictor (***x***, attribute).

5. Related work

In 2021, Musa et al., offer their contribution to the discussion of growing threats from online criminals, in particular, phishers who never stop inventing various methods of deception. In this regard, the authors underline the necessity of sophisticated systems for phishing detection. The model they have developed is based on the Extreme Gradient Boosted Tree algorithm using the XGBOOST method. According to the results of the trials, the model based on XGBOOST had accuracy of 97.27%, outperforming both Random Forest and Probabilistic Neural Network models, which produced 95.66% and 96.79% accuracies respectively. This would explain how XGBOOST might help in enhancing efforts geared toward detecting phishing scams [20].

In 2021, Naik, N. N., speaks about the increasing activities done by computers that were earlier done by humans and also the various advantages and disadvantages of such a change in scenario. One of the major disadvantages brought to light is the rise in phishing attempts through the internet. From that day on, phishing techniques and countermeasures evolved together. Machine learning algorithms played a very important role in detecting and foiling such assaults. The research proposes a scheme that makes use of the Extreme Gradient Boost algorithm to identify phishing URLs with high accuracy. In a comparative analysis, accuracy results using this model are stated as XGBoost: 0.858, Decision Tree: 0.859, Random Forest: 0.859, Multilayer Perceptron: 0.851, Support Vector Machines: 0.801[21]. These results are obtained with respect to the performance of the model against other machine learning algorithms.

In 2021, Goud and Mathur, explain Phishing: one of the prevalent hacks done through URLs to gather personal information. Ideal methodology for the classification of the URLs by Recursive Feature Elimination based on Machine Learning: The Use of Extra Tree Classifier performs Auto-feature selection, and the feature set model is compounded with 112 features that comprise preprocessing, feature relationship, and Auto-feature selection, amongst others. The model established 29 core features and obtained a classification level of 93% out of 112 features when URL was analyzed. This model mainly utilized ensemble, stacking, boosting, and bootstrap aggregation for feature selection. For further information, kindly refer to [22].

In 2021, Tabassum et al. contribute to the discussion of the problem of phishing attacks, which often take place via emails or websites, aiming at illicitly gaining users' sensitive information and causing huge financial damage to many people. In this effort of protection from all these threats, the present study tries to provide an effective framework for detecting phishing websites. Tabassum et al. integrate the concepts of bagging and boosting and introduce a hybrid model involving SVM, Decision Tree, Random Forest, and XGBoost. This technique uses some features of both the legitimate and the phishing websites to minimize the chances of a phishing attack. They assessed classification algorithms against many subsets of features selected using different techniques for feature selection in order to find the most useful and efficient set of characteristics. The accuracy rate for the proposed hybrid technique, therefore, came out to be 98.28%, hence beating the state-of-the-art methods [23].

6. Result and discussion

This section shows the results of our study on several machine learning models developed for the purpose of detecting phishing websites. In this study, we compare models such as XGBoost, Naive Bayes, K-Nearest Neighbors, Decision Trees, Random Forests, Support Vector Classifiers, and Linear Support Vector Classifiers. Our main evaluation criteria were recall, accuracy, and precision. We will present a comparison of different models and an overview of our findings. The capability of logistic regression in computing the probability that a website is a phishing site was evaluated. It did quite balanced, with intermediate recall, accuracy, and precision values, though. This was because of its linear nature, which let it have problems dealing with complex interactions. KNN performed

very well both in memory and accuracy, hence proving to be effective in phishing website detection. Though a bit less accurate, the model proved to indicate some level of false positives. Computation can be high on large datasets, but simplicity and ease of implementation make it quite feasible. The Decision Tree model showed high readability and use. Although it had a propensity to overfit, which might limit its applicability to unobserved data; it fared well in terms of accuracy and precision. Because Random Forest reduces overfitting and improves generalization, it performs better than a single Decision Tree. Its great precision and accuracy make it a reliable option for phishing detection. However, because of the ensemble technique, it may be computationally costly. SVC kernel functions let it to handle high-dimensional data and grasp intricate connections with ease. It requires rigorous hyper parameter adjustment to obtain great accuracy. Compared to other models, it took longer to train, especially for larger datasets.

Considering knowledge that can be divided linearly, linear SVC demonstrated high efficiency and scalability. Its usefulness in increasingly complicated scenarios was limited by its poor performance with nonlinearly separable data, while it excelled on big datasets. Among the best-performing models, XGBoost stood out for its remarkable accuracy, precision, and recall. It was very successful and efficient because of its scalability and parallel tree boosting. XGBoost has demonstrated supremacy in phishing detection tasks by consistently outperforming other algorithms in many experiments. Due to the independence assumption among characteristics, Naive Bayes shown limitations despite its simplicity and computational efficiency. Even though it did what was expected, random forest and XGBoost performed better than it. Based on our results, ensemble methods performed better in all evaluation metrics. Random Forest and, in particular, XGBoost did the best among all models. Ensemble models are perfect in phishing detection due to their resilience to overfitting and the capacity to treat high-dimensional input. Especially remarkable was XGBoost, given high accuracy and scalability.

Results underline the importance of machine learning in phishing detection. Machine learning models can organize dynamic protection against new methods of phishing, while classic rule-based systems are helpless. In this respect, our study justifies that ensemble approaches, particularly XGBoost, provide reliable and robust solutions for detecting phishing websites. Further research is required to focus on the implementation of these models in real-life applications and increase the ability of these models to protect customers against sophisticated cyber threats. This research is concluded by providing the show of workability of machine learning models in defending phishing attempts. Advanced algorithms or ensemble methods can be designed that actually implement such robust systems protecting user data and strengthening confidence in digital interactions.

7. Evaluation Metrics

The performance for each model was estimated by the following metrics:

1. **Accuracy**: Calculates the percentage of accurately detected cases—both true positives and true negatives—among all instances.

$$\text{Accuracy}=\frac{TP+TN}{TP+TN+FP+FN} \quad (8)$$

Where:

TP stands for True Positive; **FP** means False Positive, **TN** stands for True Negative, and **FN** means False Negative.

2. **Precision:** Evaluates the ratio of accurate positive forecasts to all positive forecasts.

$$\text{Precision}=\frac{TP}{TP+FP} \quad (9)$$

Where:

TP stands for True Positive and **FP** means False Positive.

3. **Recall:** Shows the percentage of real positive cases that the model accurately detected.

$$\text{Recall}=\frac{TP}{TP+FN} \quad (10)$$

Where:

TP stands for True Positive and **FN** means False Negative.

8. Conclusion and future work

This paper's result depicts how pivotal machine learning is to the process of phishing detection. Through continued innovation and refinement, we can make much safer online spaces that shield consumers from the ever-present dangers of phishing scams. Since phishing websites severely threaten cybersecurity, the methods for detection have to be strong. In this paper, we evaluate several machine-learning models with respect to accuracy, precision, and recall for the identification of phishing URLs. On the side of robustness and reliability, ensemble methods—Random Forest and XGBoost in particular—performed better than other models in a pretty consistent way. These results may give an idea of how machine learning can greatly improve phishing detection systems. While the more straightforward models, like Logistic Regression and Naive Bayes, have some benefits, it can be established from the comparative review of a good number of machine learning algorithms that these latter methods are incapable of keeping up effectively with the very subtle and continuously evolving nature of phishing attempts. In contrast, ensemble approaches—especially XGBoost—are both very accurate, scalable, and able to adapt quickly to new types of phishing. compare our method on DEFRAUD with current online API services and the state-of-the-art machine learning model for defending against phishing websites. Rule-based methods, in nature traditional such rules tend to get outdated quickly and are not capable of tracing new tactics used by the malicious ware every second. Over time, new proactive approaches enabled by machine learning (ML) have become more important as these solutions are flexible and adaptable in their ability to scan through modern data breaches for patterns from millions of datasets.

In future work:

1. These models should be integrated in the real-time system to test their applicability and efficacy. In a real-time detection system, there needs to be a balance between speed and accuracy for reducing the false positives and negatives.
2. It will be more trained and resilient with more updated and different instances of phishing. It can provide better protection by having region- and language-specific phishing URLs.
3. In this way, one could come up with models of hybrid nature, including several machine-learning algorithms to enable exploiting the power of each method, hence providing a much higher degree of protection against sophisticated phishing attacks.
4. One can learn strategies of adversarial training to strengthen model defense against the attacker's evasion strategy. For example, this will be done by using adversarial instances during training and thus help models improve their resilience.

References

[1] A. Mandadi, S. Boppana, V. Ravella and R. Kavitha, "Phishing website detection using machine learning," in 2022 IEEE 7th Int. Conf. for Convergence in Technology (I2CT), Mumbai, India, pp. 1–4, 2022. https:// doi.org/10.1109/i2ct54291.2022.9824801
[2] S. Kuraku and D. Kalla, "Emotet malware—A banking credentials stealer," IOSR Journal of Computer Engineering, vol. 22, pp. 31–41, 2020.

[3] A. Kulkarni and L. L. Brown, "Phishing websites detection using machine learning," International Journal
of Advanced Computer Science and Applications, vol. 10, 2019.https://doi.org/10.14569/ijacsa.2019.0100702
[4] D. Kalla and A. Chandrasekaran, "Heart disease prediction using machine learning and deep learning," International Journal of Data Mining & Knowledge Management Process (IJDKP), vol. 13, no. 3, 2023. https://doi.org/10.5121/ijdkp.2023.13301
[5] A. Safi and S. Singh, "A systematic literature review on phishing website detection techniques," Journal of King Saud University—Computer and Information Sciences, 2023. https://doi.org/10.1016/j.
jksuci.2023.01.004
[6] S. Das Guptta, K. T. Shahriar, H. Alqahtani, D. Alsalman and I. H. Sarker, "Modeling hybrid feature- based phishing websites detection using machine learning techniques," Annals of Data Science, 2022.https:// doi.org/10.1007/s40745-022-00379-8
[7] D. Kalla, F. Samaah, S. Kuraku and N. Smith, "Phishing detection implementation using databricks and artificial Intelligence," International Journal of Computer Applications, vol. 185, no. 11, pp. 1–11, 2023. https://doi.org/10.5120/ijca2023922764
[8] Nadkarni, P. M., Ohno-Machado, L., & Chapman, W. W. (2011). Natural language processing: an introduction. Journal of the American Medical Informatics Association, 18(5), 544-551.
[9] Azeez, N., Awotunde, O., & Oladeji, F. (2020). Approach for Identifying Phishing Uniform Resource Locators (URLs). Covenant Journal of Informatics and Communication Technology.
[10] P. Gupta and A. Mahajan, "Phishing website detection and prevention based on logistic regression," International Journal of Creative Research Thoughts, vol. 10, pp. 2320–2882, 2022.
[11] T. A. Assegie, "K-nearest neighbor based URL identification model for phishing attack detection," Indian Journal of Artificial Intelligence and Neural Networking, vol. 1, no. 2, pp. 18–21, 2021. https://doi.org/10.54105/ijainn.b1019.041221
[12] D. Ahmed, K. Hussein, H. Abed and A. Abed, "Phishing websites detection model based on decision tree algorithm and best feature selection method," Turkish Journal of Computer and Mathematics Education, vol. 13, no. 1, pp. 100–107, 2022
[13] G. Ramesh, R. Lokitha, R. Monisha and N. Neha, "Phishing detection system using random forest algorithm," International Journal for Research Trends and Innovation, vol. 8, pp. 510, 2023.
[14] D. Aksu, A. Abdulwakil and M. A. Aydin, "Detecting phishing websites using support vector machine algo-rithm," Pressacademia, vol. 5, no. 1, pp. 139–142, 2017.https://doi.org/10.17261/pressacademia.2017.582
[15] V. Jakkula, "Tutorial on support vector machine (SVM),"2011. [Online]. Available: https://course.ccs.neu.edu/cs5100f11/resources/jakkula.pdf (accessed on 15/04/2023)
[16] Chen, T., & Guestrin, C. (2016, August). Xgboost: A scalable tree boosting system. In Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining (pp. 785-794).
[17] G. Kamal and M. Manna, "Detection of phishing websites using Naïve bayes algorithms," International Journal of Recent Research and Review, vol. XI, no. 4, pp. 34–38, 2018.
[18] F. Mbachan, "Phishing URL prediction using logistic regression," 2022. https://doi.org/10.13140/RG.2.2.11606.93767
[19] H. Rajaguru and S. R. Sannasi Chakravarthy, "Analysis of decision tree and K-nearest neighbor algorithm in the classification of breast cancer," Asian Pacific Journal of Cancer Prevention, vol. 20, no. 12, pp. 3777–3781, 2019. https://doi.org/10.31557/APJCP.2019.20.12.3777
[20] Musa, H., Gital, A. Y., Zambuk, F. U., Umar, A., Umar, A. Y., & Waziri, J. U. (2019). A comparative analysis of phishing website detection using XGBOOST algorithm. Journal of Theoretical and Applied Information Technology, 97(5), 1434-1443. https://www.researchgate.net/publication/333134242_A_comparative_analysis_of_phishing_website_detection_using_XGBOOST_algorithm

[21] Naik, N. N. (2021). Modelling Enhanced Phishing detection using XGBoost (Doctoral dissertation, Dublin, National College of Ireland).
[22] Goud, N. S., & Mathur, A. (2021). Feature Engineering Framework to detect Phishing Websites using URL Analysis. International Journal of Advanced Computer Science and Applications, 12(7).
[23] Tabassum, N., Neha, F. F., Hossain, M. S., & Narman, H. S. (2021, May). A hybrid machine learning based phishing website detection technique through dimensionality reduction. In 2021 IEEE international black sea conference on communications and networking (BlackSeaCom) (pp. 1-6). IEEE.