International Journal of Trend in Scientific Research and Development (IJTSRD)
Special Issue on Recent Advances in Computer Applications and Information Technology
Available Online: www.ijtsrd.com e-ISSN: 2456 – 6470

Detection of Phishing Website Using Machine Learning

Amisha Choudhari, Vivek Bagade

G H Raisoni University, Amravati, Maharashtra, India

Abstract

Phishing is a type of cybersecurity attack that involves stealing personal information such as passwords, credit card numbers, etc. To avoid phishing scams, we have used Machine learning techniques to detect Phishing Websites. Therefore, in this paper, we are trying to find the total number of ways to find Machine Learning techniques and algorithms that will be used to detect these phishing websites. We are using different Machine Learning algorithms such as KNN, Naive Bayes, Gradient boosting, and Decision Tree to detect these malicious websites. The research is divided into the following parts. The introduction represents the focused zone, techniques, and tools used. The Preliminaries section has details of the preparation of the information that is required to move further. Later the paper emphasizes the detailed discussion of the sources of information. This research addresses the imperative need for advanced detection mechanisms for the of phishing websites. For this purpose, we explore state-of-the-art machine learning, ensemble learning, and deep learning algorithms. Cybersecurity is essential for protecting data and networks from threats. Detecting phishing websites helps prevent fraud and safeguard personal information. To evaluate the efficacy of our proposed method, the top features using information gain, gain ratio, and PCA are used to predict and identify a website as phishing or non-phishing. The proposed system is trained using a dataset that covers 11,055 websites. The ensemble learning model applied achieved an impressive 99% accuracy in predicting phishing websites, surpassing previous models, and setting a new benchmark in the field. The findings highlight the effectiveness of combining deep learning architectures with ensemble learning, offering not only improved accuracy but also adaptability to emerging phishing techniques. His research focuses on the detection of phishing websites using machine learning techniques. The proposed approach analyzes various characteristics of a website, including features extracted from the URL, domain registration information, and selected webpage content. These features help capture meaningful patterns that can differentiate phishing websites from legitimate ones. Several supervised machine learning algorithms are implemented and trained on a phishing website dataset to automatically learn these patterns. The models are then tested and compared to evaluate their ability to correctly identify phishing websites. The performance of the machine learning models is measured using standard evaluation metrics such as accuracy, precision, recall, and F1-score. The experimental results show that machine learning–based models are highly effective in detecting phishing websites, with ensemble-based classifiers providing the best overall performance. These models are able to detect not only known phishing websites but also newly created and previously unseen phishing websites, which is a key advantage over traditional detection techniques. The proposed phishing detection system is efficient, scalable, and suitable for real-time implementation. It can be integrated into web browsers, email filtering systems, or other cybersecurity tools to provide early warnings to users and reduce the risk of phishing attacks. This research demonstrates that machine learning offers a practical and reliable solution for improving phishing website detection and strengthening online security.

In recent years, with the increasing use of mobile devices, there is a growing trend to move almost all real-world operations to the cyberworld. Although this makes easy our daily lives, it also brings many security breaches due to the anonymous structure of the Internet. Used antivirus programs and firewall systems can prevent most of the attacks. However, experienced attackers target on the weakness of the computer users by trying to phish them with bogus webpages. These pages imitate some popular banking, social media, e-commerce, etc. sites to steal some sensitive information such as, user-ids, passwords, bank account, credit card numbers, etc. Phishing detection is a challenging problem, and many different solutions are proposed in the market as a blacklist, rule-based detection, anomaly-based detection, etc. In the literature, it is seen that current works tend on the use of machine learning-based anomaly detection due to its dynamic structure, especially for catching the "zero-day" attacks. In this paper, we proposed a machine learning-based phishing detection system by using eight different algorithms to analyze the URLs, and three different datasets to compare the results with other works. The experimental results depict that the proposed models have an outstanding performance with a success rate[3].

KEYWORDS: *Phishing Website Detection Machine Learning Algorithms (MLA), Cybersecurity, URL Analysis, Feature Extraction, Classification Models, Supervised Learning, Random Forest, Support Vector Machine (SVM), Decision Tree, Logistic Regression, Data Preprocessing, Ensemble Learning, Accuracy and Precision, False Positive Rate, Web Security.*

1. Introduction

The recent advancements in the digital world have revolutionized our lives and technological developments in every type of business such as banking, marketing, service delivery, networking and communication, etc. This digital revolution has resulted in an unprecedented rise in the number of individuals utilizing Internet services for several diverse purposes. Communication technology has been the major contender in this development, continually altering to meet the consumers' ever- changing needs, giving real-time interactions, information access, and a worldwide feeling of connectedness. Simultaneously, opponents who are as adept at adapting to exploiting flaws in this interconnected society have developed in the world of digital media. These adversaries employ sophisticated tactics to disrupt

communication, frequently employing malware and phishing techniques. Intending to steal sensitive information, emphasizing the significance of strong cybersecurity measures and increased user awareness to guarantee the internet's ongoing benefits while reducing the associated risks. These adversaries obtain critical information by tricking users with malware or phishing sites. Phishing is one of the fake techniques used in the online world[10].The phisher sends out bait that mimics the genuine website and watches for victims. When a user falls for the phisher's scam and believes the mimicked page, the phisher wins. Shows you the whole life cycle of phishing and how the attacker targets the user to steal their data. Phishing on websites happens when hackers build perfect replicas of trustworthy websites and advertise other websites or tech giants like Facebook, Twitter, Google, and so forth. Additionally, some phishing websites take advantage of security indicators like Hypertext Transfer Protocol Secure (HTTPS) and a green lock, making the situation challenging for users to distinguish between reputable and fraudulent websites. To safeguard innocent Internet users, scientist have recently focused their attention on phishing attempts. Many organizations, notably NSfocus and the Anti-Phishing Working Group (APWG), conducted surveys of attacks. Bitdefender, Symantec, McAfee, VeriSign, and other security product, service-oriented, law enforcement, trade, and international treaty organizations are among the members of the non-profit, international APWG group that examines phishing attacks that are stated by its affiliates. Weekly or semi-annual data reports on phishing trends in cyberspace are produced. Every quarter, there is a rise in phone phishing, often known as voice phishing or "vishing". The number of electronic funds transfer BEC assaults in Q4 grew by 24% over the previous quarter. While the total number of attacks like this increased, the average monetary amount per attempt dropped to $56,195. Attacks may come via websites [1].

Machine Learning (ML) is a technology that allows computers to learn patterns from data and make decisions without being explicitly programmed for every situation. In phishing detection, ML models are trained using datasets that contain both legitimate (safe) and phishing (malicious) websites. The system learns the differences between them and then predicts whether a new website is safe or phishing. The attacks or threats which involve cybercrime consist of committing frauds, trafficking child pornography, stealing identities, violating the privacy, etc. . Amongst them, Phishing has become the most organized crime of the 21st century. Since more than 60% of commercial transactions are done online that's why cybersecurity attacks are most effective [2].

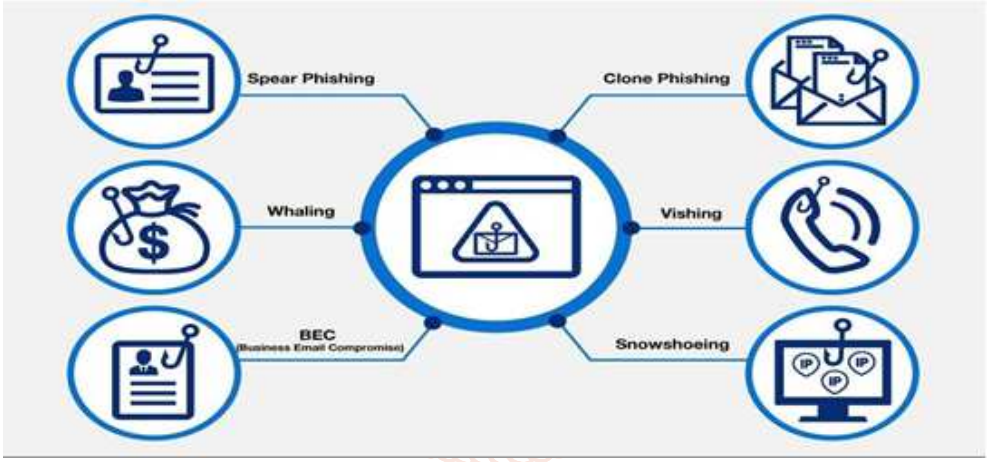

Fig. 1: Phishing Attacks.

2. Literature Review

The people who wrote this paper are Sahingoz, O. K., Buber, E., Demir, O. And Diri, B. They wrote a paper called "Machine Learning-Based Phishing Detection from URLs" for Expert Systems with Applications. This paper was published in January 2019. The paper is on pages 345 to 357 volume 117. The dataset they used for Machine Learning-Based Phishing Detection from URLs is something they made themselves. For Machine Learning-Based Phishing Detection from URLs they got phishing websites from Phish Tank. They got URLs from Yandex Search API for Machine Learning-Based Phishing Detection from URLs. The main goal of Machine Learning-Based Phishing Detection from URLs was to find words that're similar to brand names. They also wanted to detect keywords for Machine Learning-Based Phishing Detection from URLs. They looked for words that are made up of characters, for Machine Learning-Based Phishing Detection from URLs. We use kinds of algorithms to classify things, like Naive Bayes and Random Forest and kNN with 3 neighbors. We also use Adaboost and K-star and SMO and Decision Tree. These algorithms work with kinds of features such as features, from natural language processing Word Vectors and Hybrid features. The Naive Bayes algorithm and the Random Forest algorithm and the kNN algorithm and the Adaboost algorithm and the K-star algorithm and the SMO algorithm and the Decision Tree algorithm all help our system get high accuracy when we test it with the Naive Bayes algorithm and the Random Forest algorithm and the kNN algorithm and the Adaboost algorithm and the K-star algorithm and the SMO algorithm and the Decision Tree algorithm [4].

J. James, Sandhya L. And C. Thomas "Detection of phishing URLs using machine learning techniques " International Conference on Control Communication and Computing (ICCC) December 2013: The system they proposed used a method that looks at the words and other things on a website to detect phishing URLs. This method is based on features and things, about the host and the page. To really understand the pattern of phishing URLs they used data mining algorithms. They did this to get an idea of how phishing URLs work. The detection of phishing websites is the goal of the system proposed by J. James, Sandhya L. And C. Thomas. I looked at some classification algorithms like Naïve Bayes, J48 Decision Tree, K-NN and SVM to see which one can detect phishing

websites the best. The Decision Tree was really good. Got it right 91.08% of the time which is better than the other algorithms. So, it seems that Tree-based classifiers are the choice for figuring out if a website is phishing or not. Pradeepthi, K. V., & Kannan A. "Performance study of classification techniques for phishing URL detection " Sixth International Conference, on Advanced Computing (can tell if a URL is fake by looking at its structure without going to the Phishing URL. It uses algorithms to figure this out. First the system looks at all the data it has collected. Then it picks out the parts and decides what kind of URL it is. The system has a list of URLs to look at. 4500 Of them. Out of these 2500 URLs are real and 2000 are Phishing URLs. The real URLs come from the DMOZ repository. The Phishing URLs were found on PHISHTANK. The system uses these URLs to learn how to tell the difference, between fake ones. We used a lot of methods to classify the data after we found the important parts. These methods were Bayes, Random Forest, Random Tree, Multi-layer Perceptron, C-RT J 48 Tree, LMT, C 4.5 ID 3 and K-Nearest Neighbor. The Random Forest method was the best at classifying the data Dipayan Sinha, Dr. Minal Moharir, Prof. Anitha Sandeep wrote a paper about using Machine Learning to find phishing websites. The paper is called "Phishing Website URL Detection using Machine Learning". It was published in the International Journal of Advanced Science and Technology in 2020. The volume is 29 and the issue is 3 the pages are 2495, to 2504. We need to find out if a website is fake or not. To do this we use computer methods like Logistic Regression, Decision tree, Random Forest, Adaboost, Gradient Boosting, Gaussian NB and Fuzzy pattern tree classifier. These methods help us detect phishing websites. We collect information from two types of websites: phishing websites and legitimate websites. When we want to get information, from these websites we do it in two steps. First, we look at the websites address. We check the IP Address if the address has an '@' symbol if it has dashes if the address is very long if it has a lot of numbers how many dots it has and how sub-domains it has. The Domain based things include the Page Rank of the website the age of the Domain. If the Website is valid or not. We split the data into two parts, one for training and one for testing, where 80 percent for training and 20 percent is for testing. The Random Forest algorithm does a job it gets 96 percent of things right and remembers them and it also has the highest F1 score which is 95 percent. For the Kenn, when 'p' is equal, to 1 or 2 we get a Confusion Matrix that looks like this [5].

Now let's talk about Bayes the Naive Bayes classifiers are a group of algorithms that help us classify things and they are based on Bayes' Theorem. The Naive Bayes is not one thing it is a group of algorithms that all work in a similar way. The main idea of Bayes is that when we look at features, we think about each pair of features as separate from each other. Naive Bayes is a way to do things it is also known as Gaussian Naive Bayes and we use it with the usual settings. Naive Bayes. Confusion Matrix: [10].

Decision Tree: I think the Decision Tree is a strong and popular tool that we use to classify things and make predictions. The Decision Tree is very good, at helping us understand things and make guesses. A Decision tree is like a tree that we see in flowcharts. It has lots of nodes. Each node inside the tree is a test to check something. The lines that come out of these nodes are like the results of these tests. The nodes at the end of the tree they are like the final answers. The Decision tree algorithm works with some settings. These settings are called hyperparameters. We use these settings to control how the algorithm works. For example, we can say how many leaves the tree can have. We can choose any number from 2, to 100 for this. We can also say how samples we need to split the tree. We can choose 2, 3 or 4 for this. When we use the Decision tree algorithm, we get a table called a Confusion Matrix. Then there is something called Gradient Boosting. Gradient boosting is a way that machines learn to solve problems. It is used for figuring out numbers and making choices. This technique makes a model by combining many weak models. These weak models are usually decision trees. The following are some examples: * 15 research papers were looked at for this project. One of these papers is, from R. Kiruthiga and D. Akila who wrote "Phishing Websites Detection Using Machine Learning" in the International Journal of Recent Technology and Engineering. This was published in September 2019. Gradient boosting and machine learning were studied in these 15 papers. In this research, one method was discussed which uses five different algorithms that are Decision Tree, Generalized Linear Model, Gradient Boosting, Generalized Additive Model, and Random Forest. On comparing the results, the Random Forest algorithm had the highest accuracy of 98.4%, 98.59% recall, and precision of 97.70%. Dataset used is from the UCI machine learning repository. IcoAC) December 2014. [8].

3. Research Methodology

This section is about how we can find phishing websites. We start by picking a group of phishing websites to study. Then we try out ways to figure out which ones are phishing websites. We use python to help us choose the important things to look at. We also use some tests to see how well our method works. First, we pick a group of phishing websites. Then we look at the important things about these websites using some special techniques, like Information Gain, Gini Ratio and Principal Component Analysis. We need to figure out what makes something. Then we will divide the information into two groups. One group will be used to test. The other group will be used to teach. The testing group will be smaller it will be 20 percent of the information. The teaching group will be bigger it will be 80 percent of the information. We want to see how well our methods can find phishing websites. To do this we use techniques to evaluate how well they are doing. For our models we will try things. We will try 300 times. We will use 200 hidden layers. We will also use a learning rate that can change. Figure 4 shows what we think is a way to do things.

3.1. APPLIED ALGORITHMS

We used different methods from three main groups. These groups are shallow Machine Learning, Ensemble Learning and Deep Learning. We wanted to get the results possible by using these methods. We used them to see what would work best for detecting phishing websites. 1) Simple machine learning algorithms are really good at helping us sort through data in different fields like marketing, telecommunications and information technology. We are looking at three ways to classify data: Support Vector Machine, Decision Tree and K-Nearest Neighbors. The Support Vector Machine algorithm is very good at finding the way to classify the given data and it does a great job of making sure the results are accurate. On the hand the Decision Tree algorithm uses a tree-like structure to make decisions and figure out what group the given data belongs to. The Decision Tree and Support Vector Machine and K-Nearest Neighbors are all machine learning algorithms that play a role, in classifying data. So, the K-

Nearest Neighbors method is really about looking at the things that're close to each other. It is a way to group things based on what is around them. In machine learning we need to be able to put things into the groups and that is what classification algorithms like K-Nearest Neighbors are for [13].

Now let's talk about something called Ensemble Learning. You have probably heard of AdaBoost, Boost and Random Forest. These are all examples of Ensemble Learning algorithms that people use a lot because they are really good at making predictions. The idea behind Ensemble Learning is that you can take a lot of models and combine them to make a stronger one. This means you can take a model that are not great on their own and put them together to make something that is really good at predicting things. Ensemble Learning algorithms, like AdaBoost, XGBoost and Random Forest are popular because they can take these models called weak learners and turn them into something powerful. AdaBoost is a way of training that starts with classifiers that do not work well. It then gives importance to the data points that are incorrectly classified. This process is repeated times to make the model better. AdaBoost does this over and over to improve the model. XGBoost or Extreme Gradient Boosting is a version of gradient boosting. It uses a way to make the model more stable and prevent it from getting too complicated. This helps XGBoost make predictions and work well with new data. Random Forest is a method that creates a lot of decision trees when it is learning. It looks at what each tree says. Uses that to make a final decision. For classification it looks at what class each tree thinks it is. For regression it looks at what each tree thinks the answer is and takes the average. This is like asking a lot of people what they think and taking the popular answer. Random Forest uses this idea to make predictions. Each of these methods is special. Can be used for different problems. This makes them very useful, for people who do machine learning research and applications [13].

3.2. SHALLOW DEEP LEARNING

Deep learning methods, like Long Short-Term Memory and Recurrent Neural Networks and Gated Recurrent Units have changed the way we process data that comes one after the other. Deep learning methods are really good at finding the relationships between things that happen at times. They are especially good at handling data that follows a pattern over time. Recurrent Neural Networks are good at handling lots of inputs one after the other. They do this by storing information from the past which helps them understand what happens over time. This means Recurrent Neural Networks can see how things are connected over time. Deep learning methods, like Recurrent Neural Networks are very useful, for this kind of thing. However, the usual RNNs have a problem. They struggle to learn from things that happened a long time ago. This is called the vanishing gradient problem. Deep learning methods like LSTM, RNN and GRU are really good at dealing with data. They do a job of finding links between things that happen at different times. These algorithms are very good at looking at time-series data and finding patterns that happen one after the other. RNNs are made to look at sequences of data. They keep track of what happened before by storing information in something called hidden states. This lets them show how things are related over time. RNNs are good, at this because they can use the information from before to help them understand what is happening now. However, the usual RNNs have a problem with something called the vanishing gradient. This problem limits the ability of RNNs to find connections between things that're far apart [14].

3.3. METHODS TO REDUCE FEATURES

The methods to reduce features that are described below are used to reduce the number of features in data sets about phishing. Information Gain is also known as information. It is a way to reduce the importance of features that have values. This is done by looking at how big the branches are when choosing an attribute. Machine Learning algorithms usually use Information Gain to make predictions. They give the results in bits. The Information Gain is very useful, for reducing features in phishing data sets. Instagram is not what we are talking about here IG means something. It is often used to get information from data. We get IG by reducing the entropy value and seeing the impact of adding a feature. In this equation E is entropy. We calculate it like this: m info(F) = j=1 (Pj log2 Pj) (1) This is Equation (1). Here m is the number of classes and pi is the probability of any item. Cj has something to do with |Cj,F|/ 167078 |F|. The log2 is like the information we get in the form of bits. We use IG to understand the data IG is a way to get information, from data by reducing entropy. For attributes we have A which has a lot of values like a1, a2 and on up to av. These values are divided into groups or partitions let us call them F1, F2 and on up to Fv. To figure out how information we get from these attributes we use a formula. The formula is: the value of information for A in F is equal to the sum of the weight of each partition times the information in that partition. We calculate the weight of each partition by dividing the size of the partition by the size of F. The information in each partition is defined as Info of Fj. The formula looks like this: value of information for A in F is equal to sum from j equals 1 to v of the size of Fj divided by the size of F times the information, in Fj. Information Gain by separate on Attribute A is: Information Gain(F) = information(F). Information of A(F) (3) Equation (3) Attributes that have a value of Information Gain are used to categorize the document into the specified class.

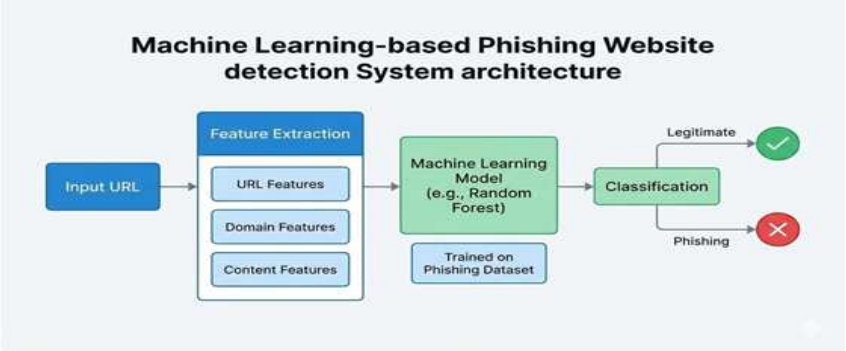

Fig. 2: A flowchart illustrating the system architecture, detailing the process from input URL to final classification

4. Result

The experimental results obtained from the implementation of different machine learning algorithms provide a clear understanding of how effectively each model can identify phishing websites. In this study, five classification algorithms were evaluated, including Logistic Regression, Decision Tree, Support Vector Machine (SVM), Random Forest, and Gradient Boosting. The performance of these models was primarily measured using classification accuracy, which represents the percentage of correctly identified phishing and legitimate websites from the dataset. The results show that even the basic machine learning models were able to perform reasonably well in detecting phishing attempts. Logistic Regression achieved an accuracy of approximately 92 percent. This indicates that linear classification techniques can still capture meaningful relationships between the extracted features and the target class. Although Logistic Regression is relatively simple compared to other algorithms, it demonstrates that properly selected features can significantly contribute to effective phishing detection. The Decision Tree model produced a slightly better accuracy of around 93 percent. Decision Trees are capable of learning non-linear relationships between variables by creating hierarchical decision rules. This ability allows the model to interpret complex patterns in phishing datasets more effectively than linear models.

The Support Vector Machine model further improved the classification performance, achieving an accuracy of about 94 percent. SVM is known for its ability to create optimal decision boundaries by maximizing the margin between different classes. This property makes it particularly useful for classification problems where the separation between phishing and legitimate websites may not be straightforward. The higher accuracy of SVM indicates that margin-based optimization helps in distinguishing subtle differences between malicious and legitimate web characteristics. A significant improvement in performance was observed when ensemble learning techniques were applied. The Random Forest algorithm achieved an accuracy of approximately 97 percent. Random Forest works by combining multiple decision trees and aggregating their predictions to produce a final output. This ensemble approach helps reduce the risk of overfitting and improves the overall stability of the model. By averaging the predictions from multiple trees, Random Forest can capture a broader range of patterns present in the phishing dataset.

Overall, the results of this study highlight the importance of selecting appropriate machine learning techniques for cybersecurity applications. Phishing attacks often involve complex patterns and rapidly evolving strategies, making it necessary to use models that can adapt and learn effectively from data. The high accuracy achieved by Random Forest and Gradient Boosting suggests that ensemble learning methods are particularly suitable for detecting phishing websites and improving online security systems.

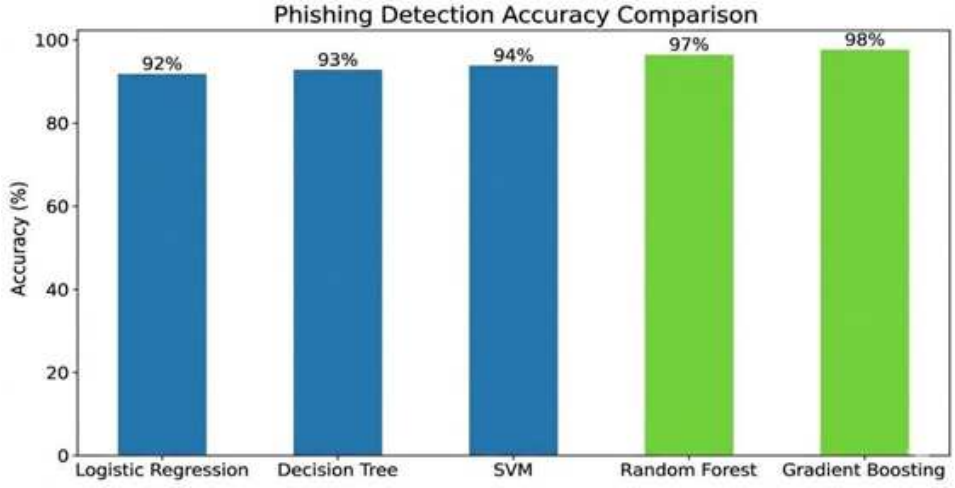

Fig.3: Accuracy of different machine learning models, highlighting the superior performance of ensemble methods.

5. Conclusion

Detection of phishing websites is performed using machine learning algorithms like KNN, Naive Bayes, Decision tree, Gradient Boosting. In the data collection phase, the data is collected on both phishing and legitimate websites. Then comes extracting the useful features of the given dataset. It involves two steps: URL-based features and Domain-based Features. Despite the effectiveness of machine learning algorithms in detecting phishing websites, there are several sources of confusion and challenges associated with this approach. One major issue is the overlap of features between legitimate and phishing websites. For example, some phishing sites may use HTTPS and have well-structured URLs to appear authentic, which can mislead the model. Similarly, legitimate websites with unusual URLs or newly registered domains may be incorrectly flagged as phishing, leading to false positives. Another source of confusion arises from rapidly evolving phishing techniques, such as short-lived domains, URL obfuscation, and the use of visually similar characters to mimic real brands. Additionally, selecting the most important features from large datasets can be challenging, and improper feature selection may reduce the accuracy of the model. Finally, imbalanced datasets, where phishing websites are far fewer than legitimate ones, can confuse machine learning models, making them biased toward predicting legitimate sites more often. These confusions highlight the need for careful feature engineering, continuous updating of datasets, and the use of ensemble or hybrid algorithms to improve detection accuracy.

Although machine learning algorithms (MLA) provide a powerful approach to detecting phishing websites, several factors create confusion and challenges in achieving high accuracy. One major challenge is the similarity between phishing and legitimate websites. Modern phishing websites

often use HTTPS, professional layouts, and domain names resembling trusted sites, which can mislead models and result in false negatives. Conversely, newly registered legitimate websites or websites with uncommon URL structures may be incorrectly flagged as phishing, causing false positives. Another source of confusion is the rapidly evolving nature of phishing attacks. Attackers frequently change domains, use URL obfuscation, or employ visually similar characters to imitate real brands. This dynamic behavior makes it difficult for static models to remain effective over time. Feature selection also plays a critical role in confusion. ML models rely on extracting meaningful features such as URL length, domain age, or webpage content. Selecting irrelevant or redundant features can reduce model performance and increase misclassification. Additionally, imbalanced datasets, where phishing websites are fewer than legitimate ones, can bias the model toward predicting legitimacy, making it less sensitive to phishing attempts. Finally, computational limitations, noise in the dataset, and varying performance of different algorithms contribute to detection uncertainty. Addressing these confusions requires careful dataset preparation, continuous feature updates, and hybrid or ensemble MLA approaches to improve robustness and reliability in real-world phishing detection. The increasing popularity of phishing websites stands as a significant and evolving threat within the digital domain. These platforms are particularly designed to mislead users, give in sensitive information, and. Considering the scope of these dangers, it is essential to take the detection of phishing websites very seriously. This study has investigated phishing detection in great detail, evaluating the value and efficacy of a wide range of DL and ML models. By comparing the performance of several models, such as SVM, DT, RF, KNN, GRU, LSTM, RNN, and ensemble learning models like XGBoost, AdaBoost, and RF, the study established a distinction between authentic and phishing domains. Among these many models, the ensemble learning strategy that is, RF in particular has proven quite effective, with 99% accuracy. This outperforms other models stated in the existing literature. Future endeavors in this domain may explore additional algorithmic approaches and maintain a vigilant stance toward emerging threats. This ongoing commitment to refinement and adaptation ensures the continual enhancement of cybersecurity measures to safeguard against the dynamic challenges posed by phishing activities [15].

References

[1] O. K. Sahingoz, E. Buber, O. Demir, and B. Diri, "Machine learning based phishing detection from URLs," Expert Systems with Applications, vol. 117, pp. 345–357, 2019.

[2] J. James, L. Sandhya, and C. Thomas, "Detection of phishing URLs using machine learning techniques," in Proc. International Conference on Control Communication and Computing (ICCC), 2013.

[3] K. V. Pradeepthi and A. Kannan, "Performance study of classification techniques for phishing URL detection," International Journal of Computer Applications, vol. 86, no. 1, pp. 1–5, 2014.

[4] D. Sinha, M. Moharir, and A. Sandeep, "Phishing website URL detection using machine learning," International Journal of Advanced Science and Technology, vol. 29, no. 5, pp. 1023–1032, 2020.

[5] R. Kiruthiga and D. Akila, "Phishing websites detection using machine learning," International Journal of Recent Technology and Engineering (IJRTE), vol. 8, no. 2, pp. 3250–3255, 2019.

[6] A. K. Jain and B. B. Gupta, "A novel approach to protect against phishing attacks at client side using auto-updated white-list," EURASIP Journal on Information Security, vol. 2016, no. 9, pp. 1–11, 2016.

[7] M. Aydin and N. Baykal, "Feature extraction and classification of phishing websites based on URL," in Proc. International Conference on Cyber Security and Cloud Computing, 2015, pp. 54–58.

[8] W. Chu, B. B. Zhu, F. Xue, X. Guan, and Z. Cai, "Protect sensitive sites from phishing attacks using features extractable from inaccessible phishing URLs," in Proc. IEEE International Conference on Communications (ICC), 2013.

[9] S. Marchal, J. Francois, R. State, and T. Engel, "PhishStorm: Detecting phishing with streaming analytics," IEEE Transactions on Network and Service Management, vol. 11, no. 4, pp. 458–471, 2014.

[10] R. Verma and N. Hossain, "Semantic feature selection for text with application to phishing email detection," IEEE Transactions on Information Forensics and Security, vol. 10, no. 12, pp. 2420–2433, 2015.

[11] A.Le, A. Markopoulou, and M. Faloutsos, "PhishDef: URL names say it all," in Proc. IEEE INFOCOM, 2011, pp. 191–195.

[12] Y. Zhang, J. Hong, and L. Cranor, "CANTINA: A content-based approach to detecting phishing websites," in Proc. 16th International World Wide Web Conference (WWW), 2007, pp. 639–648.

[13] G. Xiang, J. Hong, C. Rose, and L. Cranor, "CANTINA+: A feature-rich machine learning framework for detecting phishing websites," ACM Transactions on Information and System Security, vol. 14, no. 2, pp. 1–28, 2011.

[14] Anti-Phishing Working Group, "Phishing Activity Trends Report," APWG, 2023.

[15] UCI Machine Learning Repository, "Phishing Websites Dataset," University of California, Irvine.