Applying Machine Learning for Identifying Fraud Sites

Renuka Reddy
Department of MCA
AMC Engineering College
Bengaluru
renukareddy2000ragu@gmail.com

Prof. Gunasekaran K
Asst. Professor
Department Of MCA
AMC Engineering College
Bengaluru

Abstract

Offenders looking for sensitive information create illicit clones of legitimate websites and e-mail accounts. The email will contain actual company logos and phrases. When a user clicks on one of these hackers' links, the hackers obtain access to all of the user's sensitive information, including bank account information, personal login passwords, and photos. Random Forest Decision Tree methods and are employed often in current systems, and their accuracy must be improved. The current models have a low latency. Existing systems lack a specialised user interface.

Not all algorithms are compared in the present system. When consumers read the e-mails or links given, they are sent to a phoney website that looks to be from the legitimate firm. The models to identify fraudulent websites based on URL importance factors and to discover and execute the best machine learning model. The The comparison of machine learning methods includes logistic regression, multinomial Naive Bayes, and XG Boost. Logistic Regression beats the other two algorithms.

Phishing attack, Machine learning.

I. INTRODUCTION

presently spoofing is a crucial worry for security researchers since Not at all straightforward to create a phoney website that appears to be a real website. Experts can recognise bogus websites, but not all users can, and as a result, they become victims of phishing attacks. The attacker's primary goal is to steal bank account details. Businesses in the United States lose $2 billion every year as a outcome of their clientele falling victim to phishing.

Consequently, the third Index for Microsoft Computing Safety Report, published in February 2014, the yearly global effect of phishing might a maximum of $5 billion [2]. Because of a missing users knowledge, phishing assaults are becoming more successful. Because phishing attacks exploit user vulnerabilities, they are difficult to stop, yet it is critical to improve phishing detection systems.

Phishing is a widely utilised technique to mislead unsuspecting people into providing personal information by utilising phoney websites. Phishing website URLs are intended to steal individual data, such as user names, passwords, and online financial transactions. Phishers use websites that are visually and linguistically similar to legitimate websites. To avoid the rapid evolution of phishing techniques as a result of growing technology, anti-phishing approaches must be used to spot phishing. Machine learning is an effective method for preventing phishing attacks. Hackers typically use phishing because it is easier to trick a victim into opening a malicious link that appears to be legitimate than it is to try to circumvent a computer's security mechanisms. The malicious links inside the message body are designed to seem to lead to the faked firm by using its logos and other authentic information. Machine learning is applied in the method provided to build a breakthrough way for detecting phishing websites.

The Gradient Boosting Classifier model was used in our proposed technique to identify phishing websites based on URL importance. The recommended technique employs gradient boosting classifier to identify retrieving and analysing phishing URLs comparing distinct properties between authentic phishing URLs, etc. The outcomes of the experiments show that the proposed technique successfully distinguishes authentic websites from fake ones in real time.

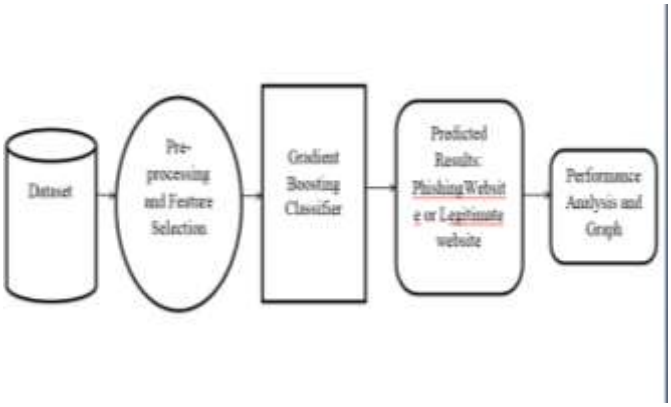

Fig. 1. Proposed Architecture

H. Huang et al., (2009) developed methods for distinguishing phishing by using using a page section analogy to dissect URT tokens and provide forecasts precision phishing pages often retain their CSS style similar to their aim pages.

This approach was introduced by S. Marchal et al., (2017) to distinguish The analysis of genuine site server log information is required for phishing websites. An off-the-shelf programme or the identification of a phishing website. Free, demonstrates a number of exceptional qualities such as high precision, total autonomy, and great language-freedom, speed of choosing, flexibility to dynamic phish, and flexibility to develop in phishing methods.

Mustafa Aydin et al. suggested a classification technique for detecting website phishing by collecting URL characteristics from webpages and analysing subset-based feature selection methods. It uses feature extraction and selection approaches to recognise phishing websites. Alpha-numeric Character Analysis, Keyword Analysis, Security Analysis, Domain Identity Analysis, and Rank Based Analysis are the five various analyses of the extracted characteristics about the links to the sites and the assembled feature matrix. The majority of these elements are textual aspects the links to itself, while some are dependent on third-party services.

Logistic Regression, Multinomial Naive Bayes, and XG Boost are the techniques to machine learning that are compared in the present system. Logistic Regression beats the other two algorithms. In the proposed system, the model is preprocessed, the words are tokenized, and stemming is conducted. The process of transforming or encoding data for simple machine transport is known as data processing. Logistic Regression has an accuracy of 96.63 percent, and the entire comparison is shown.

The current models have minimal latency.

Existing systems lack a specialised user interface.

The current system model is incapable of predicting a continuous result. It only works if the dependent or outcome variable is binary.

If the sample size is too little, the existing system model may be inaccurate.

The current situation may result in an overfitting issue.

 We created our project with a website serving as a platform for all users. This is a responsive, interactive website that will be used to determine whether a website is true or fraudulent. This website was created utilising a variety of web design languages, including HTML, CSS, Javascript, and the Flask framework in Python. HTML is used to create the website's fundamental structure. CSS is used to enhance the appearance and usability of a website by adding effects. It should be mentioned that the website is designed for all users, thus it must be simple to use and no user should have any trouble using it.

 The suggested system is trained using a dataset comprised of several attributes; however, the dataset contains none of the website URLs. The dataset contains several criteria which must be considered while deciding if a website URL is real or fraudulent. The Gradient Boosting Classifier is used to create the suggested system. After The system has undergone training using the dataset, the classifier identifies the provided URL based on the prepared information; if the site is phishing, it alerts the that the website's user phished; if the site is real, it alerts

the that the website's user authentic. We discovered phishing sites that use a 97% accuracy using Gradient Boosting Classifier.

A user interface is given, and the model is trained using a variety of characteristics.

High level of precision

The suggested method is typically more accurate than previous modes, and it can train quicker, especially on bigger datasets.

Most of the suggested systems enable the processing of categorical characteristics, and several of them handle missing values natively.

The basic approach of checking websites for phishing by updating banned URLs and Internet Protocol (IP) addresses in the antivirus database, commonly known as the "blacklist" method. To avoid blacklists, attackers employ inventive tactics to deceive consumers by altering the URL to appear genuine using obfuscation and many more simple techniques like: fast-flux, in which proxies are created automatically to host the website; automated creation of new URLs; and so on. The main disadvantage adopting this tactic is that it cannot identify zero-hour phishing attacks.

Among the many often used algorithms in machine learning. It is easy to understand the decision tree algorithm, and implement. The root of the decision tree is chosen by the decision tree as the best splitter among the categorization options provided. Up to the leaf node, the algorithm keeps growing the tree. A training model is created by the decision tree and used to forecast the target value or class. Each internal node in a tree representation is node of the tree represents Each leaf node represents a class label and an attribute. The decision tree algorithm's gini index and data gain approaches are used to determine these nodes.

The random forest algorithm, which is based on the notion of the One of the most potent algorithms is the decision tree algorithm in machine learning technology. The random forest method generates a forest with a large number decisions using trees. There are a lot of trees results in excellent detection accuracy. The bootstrap technique utilised to create trees. The bootstrap strategy uses a random selection of dataset characteristics and samples with replacement to build a single tree. Random forest algorithm, like decision tree algorithm, will identify the best splitter from among randomly selected characteristics. Random forest algorithm also employs strategies to increase information using the Gini index determine the best splitter. This technique will be repeated until the random forest produces n trees.

Another effective technique in machine learning technology is the support vector machine. Each data item is displayed as a point in n-dimensional space in the support vector machine algorithm, and the support vector machine method generates a separating line for classification of two classes; this separating line is commonly known as a hyperplane. The support vector machine searches for the nearest points, known as support vectors, and then constructs a line linking them. The support vector machine then creates a separation line that is perpendicular to and bisects the connecting line. The margin should be as wide as possible in order to accurately categorise data. The margin in this case is the distance between the hyperplane and the support vectors.

CONCLUSIONS

Using machine learning technologies, this article tries to improve detection methods for phishing websites. Using the random forest approach, We managed to recognise targets with a 97.14% accuracy rate and the lowest false positive rate. Also, the results demonstrate classifiers function more effectively when more data is utilised as practise data. In the future, The use of hybrid technologies utilised to detect phishing websites more precisely, with the using the machine learning technology's random forest technique, the blacklist approach being applied.

REFERENCES

[1] Gunter Ollmann, “The Phishing Guide Understanding & Preventing Phishing Attacks”, IBMInternet Security Systems, 2007.

[2] https://resources.infosecinstitute.com/category/enterprise/phishing/the-phishing-landscape/phishing-data-attack-statistics/#gref

[3] Mahmoud Khonji, Youssef Iraqi, "Phishing Detection: A Literature Survey IEEE, and Andrew Jones, 2013
[4] Mohammad R., Thabtah F. McCluskey L., (2015) Phishing websites dataset. Available: https://archive.ics.uci.edu/ml/datasets/Phishing+Websites Accessed January 2016
[5] http://dataaspirant.com/2017/01/30/how-decision-tree-algorithm-works/
[6] http://dataaspirant.com/2017/05/22/random-forest-algorithm-machine-learing/
[7] https://www.kdnuggets.com/2016/07/support-vector-machines-simple-explanation.html
[8] www.alexa.com
[9] www.phishtank.com
[10] L. Tang and Q. H. Mahmoud, "A Deep Learning-Based Framework for Phishing Website Detection", *IEEE Access*, vol. 10, pp. 1509-1521, 2022.
[11] P. Yang, G. Zhao and P. Zeng, "Phishing Website Detection Based on Multidimensional Features Driven by Deep Learning", *IEEE Access*, vol. 7, pp. 15196-15209, 2019.
[12] W. Ali and S. Malebary, "Particle Swarm Optimization-Based Feature Weighting for Improving Intelligent Phishing Website Detection", *IEEE Access*, vol. 8, pp. 116766-116780, 2020.
[13] C. Pham, L. A. T. Nguyen, N. H. Tran, E. -N. Huh and C. S. Hong, "Phishing-Aware: A Neuro-Fuzzy Approach for Anti-Phishing on Fog Networks", *IEEE Transactions on Network and Service Management*, vol. 15, no. 3, pp. 1076-1089, Sept. 2018.
[14] O. Abdullateef et al., "Improving the phishing website detection using empirical analysis of Function Tree and its variants", *Heliyon*, vol. 7, no. 7,2021.
[15] Dong-Jie Liu, Guang-Gang Geng, Xiao-Bo Jin and Wei Wang, "An efficient multistage phishing website detection model based on the CASE feature framework: Aiming at the real web environment", *Computers Security*, vol. 110, pp. 102421, 2021.