JOURNAL OF OPTOELECTRONICS LASER ISSN:1005-0086

Volume 41 Issue 6, 2022

Phishing Websites Detection Using Machine Learning

Suhani Jain[*1], Dr. Naveen Choudhary[2] & Kalpana Jain[3]

*[*1]M Tech Student, Computer Science Department, College of Technology and Engineering, MPUAT, Udaipur*

[2]HOD of Computer Science Department, College of Technology and Engineering, MPUAT, Udaipur

[3]Professor of Computer Science Department, College of Technology and Engineering, MPUAT, Udaipur

ABSTRACT Phishing is an online crime in which a criminal tries to persuade unsuspecting users to reveal sensitive (and valuable) personal information to the miscreant, such as usernames, passwords, financial account details, personal addresses, SSNs, and social contacts, for harmful purposes. Phishing is usually carried out by impersonating a reliable entity in Internet communication, which is accomplished through a combination of social engineering and technical trickery. Attackers regularly employ spoofing emails and deceptive websites to persuade users to provide personal information. Spoofing emails frequently pretend to be from legitimate companies and direct consumers to fake websites where they can enter important information. Phishing is one of the most common forms of online crime in today's world. Checking URLs against blacklists of known phishing websites, which are generally built based on manual verification, is a frequent countermeasure that is inefficient. As the Internet develops in size, automatic URL recognition becomes more necessary to offer end users with timely protection. This paper explains how to use machine learning to detect dangerous phishing websites, with an emphasis on attributes retrieved just from the URL. It starts with a description of the available data and the feature engineering process, then moves on to choosing acceptable machine learning approaches. It compares algorithm performance and assesses the outcomes obtained.

Keywords: Machine Learning, Classification, Algorithm, Features Extraction.

I. INTRODUCTION

When it comes to cyberattacks, phishing is still one of the most common methods used by attackers with malevolent intent. Phishing is a sort of attack that combines technical deception and social engineering to steal personal or otherwise sensitive information from the target, such as login passwords, credit card information, or trade secrets. Typically, an attack is carried out by sending a spoof e-mail or message that appears to be a real request from a well-known business. A link in the received item usually sends a potential victim to a corrupt website, which is often constructed to look like the assumed entity's legitimate website. The victim is subsequently enticed to enter sensitive information, which the phishing website's operator can then misuse. Today's systems have advanced to the point where they can detect malware with extreme precision. They enabled us to practically eliminate the human aspect from the equation, resulting in a significant reduction in malware-hosting websites. However, because phishing involves social engineering, getting the same results with phishing websites is more challenging. This fact could be the primary explanation for the continued rise in phishing efforts.

II. SIGNIFICANCE OF THE STUDY

Machine learning is a branch of computer science that uses algorithms and mathematical models to tackle various computer science problems without the use of explicit instructions. Instead than using explicit code instructions, the algorithms learn through patterns and inference. We use a more formal description to define how a programme "learns": If a computer program's performance on T, as measured by P, increases with experience E, it is said to learn from experience E with respect to some task T and some performance metric P. A spam filter tool that learns to detect spam e-mail (Task T) by recognising patterns in past e-mails is a good example (experience E).

This paper offers a method for detecting phishing URLs that uses machine learning techniques and attributes taken just from the URL. This may provide a practical advantage in terms of network security. Network monitoring and security can keep an eye on and defend the entire network. This means that phishing detection based on URLs could protect all users on the network. As a result, the suggested method merely uses data collected from the URL, obviating the requirement for any active scans or website content analysis. The experiment's findings show that models trained on the types of features provided are capable of achieving spectacular results. The proposed research work is based on the machine learning techniques for the detection and classification phishing websites dataset available at Phishtank and Kaggle website.

III. REVIEW OF RELATED STUDIES

Yuxiang Guan *et. al.* (2018) Analyze phishing website features and offer two types of web phishing detection features: domain and name-based features. In order to achieve high accuracy and efficiency, many sorts of characteristics have been extracted. The model has been strengthened by the extraction of several features.

Samuel Marchal *et al.* (2015) Used the attributes collected from words that constitute a URL based on query data from Google and Yahoo search engines, one idea of intra-URL relatedness was evaluated. These characteristics are

then employed in a machine-learning-based categorization to identify phishing URLs in a real-world data collection. He has presented a phishing detection machine learning technique that uses solely lexical and domain features.

Vahid Shahrivari *et al.* (2019) On the phishing website dataset, which contains 6157 legal websites and 4898 phishing websites, multiple classifiers have been implemented and assessed. Logistic Regression, Decision Tree, Support Vector Machine, Ada Boost, and Random Forest are the classifiers that were tested. Different types of classifiers were used, and the results were compared. Random forest emerged as the most efficient classifier after the successful comparison. To evaluate the accuracy, a large data set was used.

Rishikesh Mahajan (2018) uses machine learning technologies to improve the detection process for phishing websites. Using the random forest technique, he achieved 94.14 percent detection accuracy with the lowest false positive rate. In addition, the results demonstrate that classifiers perform better when more data is utilised as training data.

M. Karabatak and T. Mustafa (2018) proposed feature selection techniques to reduce dataset components for higher order execution It also compared the outcomes of different data mining classification techniques. The phishing website dataset was obtained from the UCI machine learning library. He also proposes the use of a c4.5 decision tree approach to detect phishing URL websites. This method calculates heuristic values by extracting features from the sites. The c4.5 decision tree algorithm was given these values to determine if the site was phishing or not.

GoldPhish (2010) extracts page information from a displayed page using optical character recognition. It then employs search engines to determine whether the content of the page is consistent with its domain, so identifying phishing sites. He combines URLs, HTML DOM (Document object model), third-party services, and search engines to create the 15 features. To detect phishing assaults, it trains these attributes using SVM (support vector machine).

IV. PROPOSED PLAN OF WORK

The proposed work is defined in various steps which are as follows:

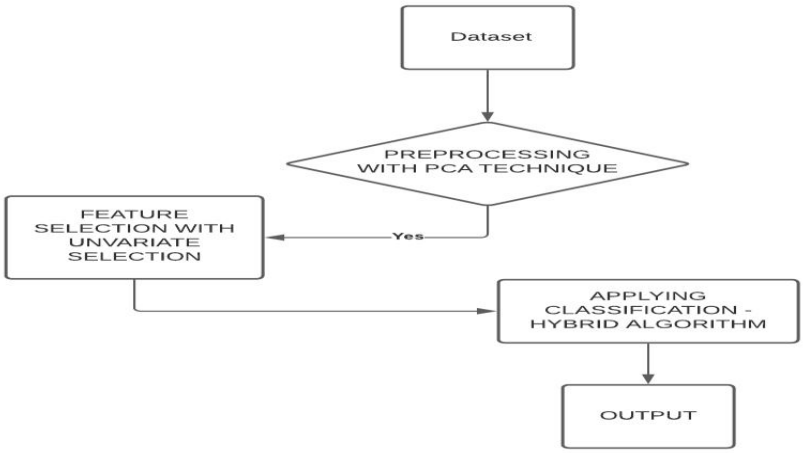

Fig 1: Flow chart of the proposed Methodology

Step 1: Data Collection: Collect the dataset from the authorized websites.

Step 2: Data Pre-Processing Technique: After selecting the data convert all the null values into 0, and other characters into numeric values.

Step 3: Standardization of The Data
Standardization is all about scaling your data in such a way that all the variables and their values lie within a similar range. It will calculated by:

$$Z = \frac{Variable\ value - mean}{Standard\ deviation}$$

Step 4: PCA - Principal Component Analysis:
Principal components analysis (PCA) is a dimensionality reduction approach that allows you to find correlations and patterns in a data collection so that it can be turned into a data set with much fewer dimensions while retaining all of

the critical information. PCA is an unsupervised statistical technique for reducing the size of a dataset's dimensions. In exploratory data analysis and machine learning for predictive models, PCA is the most extensively used tool. PCA is frequently used to remove noise and unmeasured values in data.

Step 5: Computing the Covariance Matrix
PCA can be used to find correlations and dependencies between features in a data set. The correlation between the different variables in the data set is expressed by a covariance matrix. Heavyly dependent variables must be identified since they include biassed and redundant information, lowering the model's overall performance.

Step 6: Calculating the Eigenvectors and Eigenvalues
Eigenvectors and eigenvalues are the mathematical constructs that must be computed from the covariance matrix in order to determine the principal components of the data set.

Step 7: Computing the Principal Components
All we have to do now is sort the Eigenvectors and Eigenvalues in decreasing order, with the eigenvector with the highest eigenvalue being the most significant and therefore forming the first principal component. In order to minimise the dimensionality of the data, the primary components of lesser relevance can be deleted.

STEP 8: Reducing the Dimensions Of The Data Set
The final stage in PCA is to reorganise the original data using the final principle components, which represent the data set's maximum and most significant information. The newly created Principal Components were used to replace the original data axis.

V. FEATURE EXTRACTION PROCESS

Step 1: Feature Selection: When creating a predictive model, feature selection is the process of minimising the number of input variables. The number of input variables should be reduced to lower the computational cost of modelling and, in some situations, to increase the model's performance.
Wrapper methods (forward, backward, and stepwise selection), Filter methods (ANOVA, Pearson correlation, variance thresholding), and Embedded methods are the three forms of feature selection (Lasso, Ridge, Decision Tree). The filter approach is used here.

1. **Filter Methods**: Filter methods test the usefulness of a subset of features by actually training a model on it, whereas filter methods measure the importance of features by their correlation with the dependent variable. Filter techniques are much faster than wrapper methods since they do not require the models to be trained.

Univariate Analysis
Statistical tests can be used to select those features that have the strongest relationship with the output variable. The scikit-learn library provides the SelectKBest class that can be used with a suite of different statistical tests to select a specific number of features.

The following category of features are selected:
• Address Bar based Features
• Domain based Features
• HTML & JavaScript based Feature

1) IP Address: If the URL contains an IP address rather than a domain name, such as http://217.102.24.235/sample.html.
2) URL Length: Phishers can hide the suspicious element of a URL in the address bar by using a long URL.
3) **Shortening Service**: Provides links to websites with long URLs. The URL http://sharif.hud.ac.uk/, for example, can be reduced to bit.ly/1sSEGTB.
4) Using the @ sign in the URL causes the browser to ignore everything before the @ symbol, and the real address frequently comes after the @ symbol.
5) Redirection with a Double Slash: The presence of / in the URL indicates that the user will be redirected to another website.
6) Prefix Suffix: Phishers often add prefixes or suffixes to domain names, separated by (-), to make consumers believe they are dealing with a reputable website. For instance, http://www.Paypal-Confirme.com.
7) Subdomain: Having a subdomain in the URL.
8) SSL State: Shows that website use SSL

9) Domain Registration Length: Because a phishing website only exists for a limited time, this factor is important.
10) Favicon: A favicon is a little visual image (icon) that is connected with a website. If the favicon is loaded from a domain other than the one displayed in the address bar, the webpage is most certainly a Phishing attempt.
11) Using Non-Standard Ports: It is much better to only open ports that you need to regulate incursions. Several firewalls, proxy servers, and Network Address Translation (NAT) servers will, by default, block all or most of the ports and only allow access to those that have been selected.
12) HTTPS token: URL with a deceptive https token. For instance, http://www.mellat-phish.ir
13) Request URL: Request URL checks whether external elements such as photos, videos, and music on a webpage are loaded from another domain.
14) Anchor's URL: The a > tag defines an anchor as an element. This feature is handled in the same way as Request URL.
15) Meta Tags: Legitimate websites frequently utilise Meta tags to provide metadata about the HTML text.

	count	mean	std	min	25%	50%	75%	max
UsingIP	11054.0	0.313914	0.949495	-1.0	-1.0	1.0	1.0	1.0
LongURL	11054.0	-0.633345	0.765973	-1.0	-1.0	-1.0	-1.0	1.0
ShortURL	11054.0	0.738737	0.674024	-1.0	1.0	1.0	1.0	1.0
Symbol@	11054.0	0.700561	0.713625	-1.0	1.0	1.0	1.0	1.0
Redirecting//	11054.0	0.741632	0.670837	-1.0	1.0	1.0	1.0	1.0
PrefixSuffix-	11054.0	-0.734938	0.678165	-1.0	-1.0	-1.0	-1.0	1.0
SubDomains	11054.0	0.064049	0.817492	-1.0	-1.0	0.0	1.0	1.0
HTTPS	11054.0	0.251040	0.911856	-1.0	-1.0	1.0	1.0	1.0
DomainRegLen	11054.0	-0.336711	0.941651	-1.0	-1.0	-1.0	1.0	1.0
Favicon	11054.0	0.628551	0.777804	-1.0	1.0	1.0	1.0	1.0
NonStdPort	11054.0	0.728243	0.685350	-1.0	1.0	1.0	1.0	1.0
HTTPSDomainURL	11054.0	0.675231	0.737640	-1.0	1.0	1.0	1.0	1.0
RequestURL	11054.0	0.186720	0.982458	-1.0	-1.0	1.0	1.0	1.0
AnchorURL	11054.0	-0.076443	0.715116	-1.0	-1.0	0.0	0.0	1.0
LinksInScriptTags	11054.0	-0.118238	0.763933	-1.0	-1.0	0.0	0.0	1.0
ServerFormHandler	11054.0	-0.595712	0.759168	-1.0	-1.0	-1.0	-1.0	1.0
InfoEmail	11054.0	0.635788	0.771899	-1.0	1.0	1.0	1.0	1.0
AbnormalURL	11054.0	0.705446	0.708796	-1.0	1.0	1.0	1.0	1.0
WebsiteForwarding	11054.0	0.115705	0.319885	0.0	0.0	0.0	0.0	1.0
StatusBarCust	11054.0	0.762077	0.647516	-1.0	1.0	1.0	1.0	1.0

Fig 2 : List of all Features

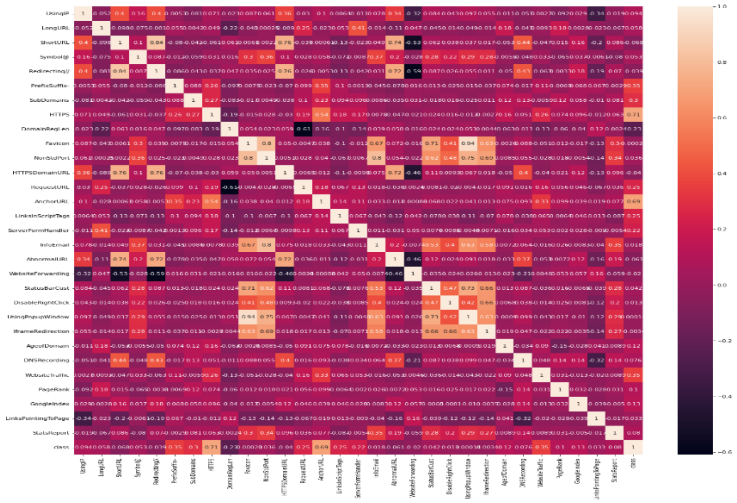

Fig : 3 Heat Map of all the Feature

VI. APPLYING ALGORITHM

STEP 1: Classification Hybrid Model: This step is the major step in machine learning. This is when the cleaned and pre-processed data is sent into the intelligent algorithms for classification. Here we will choose the SVM and RANDOM FOREST hybrid algorithms that are suitable for discovering patterns in the data. This hybrid algorithms will provide better accuracy in terms of knowledge discovery than others.

Algorithm Used

1. **Random Forest**: A random forest is a machine learning technique for solving classification and regression problems. The (random forest) algorithm determines the outcome based on decision tree predictions. It

forecasts by averaging or averaging the output of various trees. To generate a more accurate and reliable prediction, random forest creates many decision trees and blends them together. A decision tree or a bagging classifier have approximately the same hyperparameters as a random forest.

2. **Support Vector Machine (Svm):** SVM is a supervised machine learning algorithm which can be used for classification or regression problems. It uses a technique called the kernel trick to transform your data and then based on these transformations it finds an optimal boundary between the possible outputs.

Step 2: INTERPRETATION : In this step, the results were compared and analyzed in a human viewable form so as to determine which algorithm performs well and how better the results have been obtained.

VII. PERFORMANCE EVALUATION

Performance of the proposed research work will be carried out using certain evaluation parameters, namely; Accuracy, Recall, Precision, False Negative Rate, False Positive Rate, True Positive Rate and True Negative Rate.

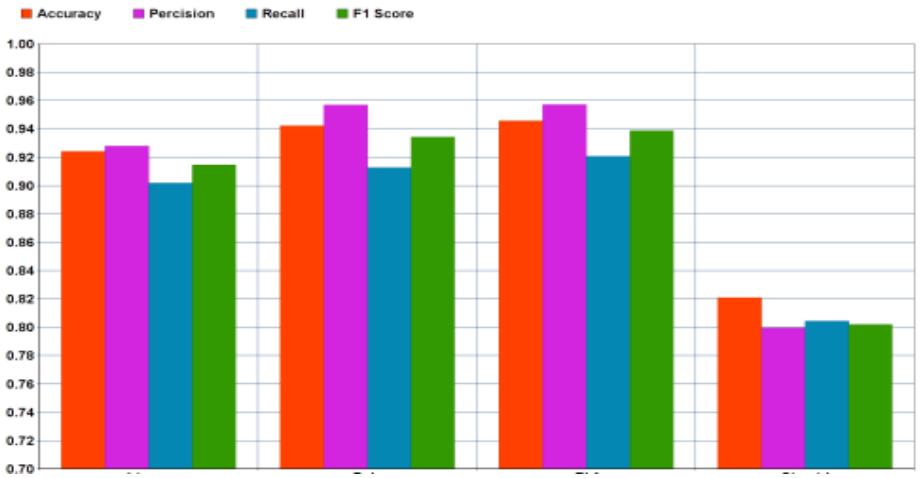

Fig 4 : Output and comparison of the proposed and recent work

VIII. RESULT AND OUTPUT

Comparison of the proposed methodology to the previous work done till now.

METHODS	ACCURACY	PRECISION	SENSITIVITY	F-SCORE
PROPOSED METHODOLOGY	96.8	96.8	97.6	97.2
METHODOLOGY WITHOUT HYBRID CLASSIFICATION	95.6	95.3	95.0	94.3
METHODOLOGY WITHOUT FEATURE EXTRACTION WITH FILTER METHOD	93.2	92.0	92.5	93.3
METHODOLOGY WITHOUT PCA TECHNIQUE	90.2	90.1	90.7	92.6
METHODOLOGY WITH SMALL DATA SETS.	91.6	92.5	90.4	91.0

IX. SOFTWARE

The tools and library which can be required for the implementation of proposed work are as follows:
Jupyter Notebook: The Jupyter Notebook is a free web programme that lets you create and share documents with live code, equations, visualisations, and narrative text. Data cleansing and transformation, numerical simulation, statistical modelling, data visualisation, machine learning, and many other applications are all possible.

Scikit-Learn: Scikit-learn (previously scikits.learn and sklearn) is open-source software. Scikit-learn is a financially supported NumFOCUS project. In Python, Scikit-learn (Sklearn) is the most usable and robust machine learning library. It uses a Python consistency interface to deliver a set of efficient machine learning and statistical modelling capabilities, such as classification, regression, clustering, and dimensionality reduction.
Python: Python is a high-level, interpreted programming language for general-purpose applications. Python has a design philosophy that prioritises code readability, which includes a lot of whitespace. It has structures that allow for clear programming at both small and large sizes.

X. CONCLUSION FUTURE WORK

Machine learning has made significant progress in the field of phishing website classification and detection. In this thesis, the two most common classification algorithms (Random Forest and SVM) will be combined into a hybrid model to discover which is more effective in detecting phishing websites. In data preparation, feature selection will be combined with the PCA technique, and the best results will be determined by overall detection accuracy. Machine learning is a data analysis and scientific research of algorithms that has recently proven results in combating phishing pages when compared to visualisation, legal remedies, such as awareness workshops, and traditional anti-phishing tactics.

When machine learning techniques are utilised in research, they can extract a variety of useful data. As a result, machine learning is being more widely used in sectors such as biomedicine, information technology, communication engineering, agriculture, and medicine. Machine learning algorithms are used to classify data into previously defined categories. They employ a machine learning approach, in which the system learns about classification rules and behaves accordingly with new or previously unseen data, resulting in the determination of their labels/classes. Previously, machine learning/data mining techniques for classification were used in a range of sectors, including security, healthcare, agriculture, and management. Since a decade, classification algorithms such as Nave Bayes, neural networks, decision tree induction, and SVM have been used in various application domains where classification is needed. These machine learning techniques, when implemented in the field of online security, an automatic detection of phishing websites detection will be developed.

Although URL lexical features alone have been demonstrated to have a high level of accuracy (97 percent), phishers have learnt how to make predicting a URL destination difficult by meticulously modifying the URL. As a result, the most successful way is to combine these properties with others, such as host. We plan to create the phishing detection system as a scalable web service with online learning so that new phishing assault patterns can be easily learned and our models' accuracy can be improved with better feature extraction in the future.

REFERENCES

[1] Abu-Nimeh, S., Nappa, D., Wang, X. and Nair, S. 2007. A comparison of machine learning techniques for phishing detection. Anti-Phishing Working Groups Ecrime Researchers Summit, pp. 60–69.
[2] Almseidin, M., Zuraiq. A.A., Al-kasassbeh, M. and Alnidami, N. 2019. Phishing detection based on machine learning and feature selection methods. International journal of interactive mobile technology, 13 : 171–183.
[3] Cao, J.C., Qiang, L., Yuede, Ji., Yukun, He. and Dong, Guo. 2014. Detection of Forwarding-Based Malicious URLs in Online Social Networks. International Journal of Parallel Programming, pp. 1–18
[4] Chou, N., Ledesma, R., Teraguchi, Y., Boneh, D. and Mitchell, J.C. 2004. Client-side defense against web-based identity theft.Proceedings of the 11th Annual Network and Distributed System Security Symposium (NDSS), pp. 76-82.
[5] Dunlop, M., Groat, S. and Shelly, D. May 2010. Goldphish: Using images for content-based phishing analysis. Internet Monitoring and Protection (ICIMP), 5th International Conference on. IEEE, Barcelona, pp. 123–128.
[6] Fette, I., Sadeh, N. and Tomasic, A. May 2007. Learning to detect phishing emails. Proceedings of the International World Wide Web Conference (WWW), pp. 649-656.
[7] Harinahalli, G.L. and BoreGowda, G. 2020 Phishing website detection based on effective machine learning approach. Journal of Cyber Security Technology, pp. 1-14.

[8] Hassan, Y.A. and Abdelfettah, B. 2017. Using case- based reasoning for phishing detection. Procedia Computer Science. 109 : 281–288.

[9] Jain, A. K. and Gupta, B.B. 2018. PHISH-SAFE: URL Features-Based Phishing Detection System Using Machine Learning. Cyber Security. Advances in Intelligent Systems and Computing. 729 : 147-152.

[10] Kumar, J., Santhanavijayan, A., Janet, B., Rajendran, B. and Bindhumadhava, B.S. 2020. Phishing Website Classification and Detection Using Machine Learning. International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, pp. 1–6.

[11] Lee, L.H., Lee, K.C., Juan, Y.C., Chen, H.H, and Tseng, Y.H. 2014. Users Behavioral Prediction for Phishing Detection. Proceedings of the 23rd International Conference on World Wide Web. No. 1, pp. 337–338.

[12] Mao, J., Tian, W., Li, P., Wei, T. and Liang, Z. 2017. Phishing website detection based on effective css features of web pages. 12th International Conference on Wireless Algorithms, Systems, and Applications, pp. 804–815.

[13] Medvet, E., Kirda, E. and Kruegel, C. September 2008. Visual-similarity-based phishing detection. Proceedings of Secure Comm ACM. 129 : pp.224-230

[14] Nourian, A., Ishtiaq, S. and Maheswaran, M. 2009. Castle: A social framework for collaborative anti-phishing databases. ACM Transactions on Internet Technology, pp.

[15] Pan, Y. and Ding, X. 2006. Anomaly based web phishing page detection. Computer Security Applications Conference, pp. 381–392.

[16] R, Mahajan. 2018. Phishing Website Detection using Machine Learning Algorithms. International Journal of Computer Applications. 123: pp. 45-47.

[17] Srinivasa, R.R, and Pais, A. R. 2017. Detecting phishing websites using automation of human behavior. Proceedings of the 3rd ACM workshop on cyber-physical system security, pp 33–42.

[18] Tan, C.L., Chiew, K.L, and Wong, K. 2016. Phish WHO: phishing webpage detection via identity keywords extraction and target domain name finder. Decision Support Systems. 88 : 18–27.

[19] Wu, C.Y., Kuo, C.C, and Yang, C.S. 2019. A phishing detection system based on machine learning. International Conference on Intelligent Computing and its Emerging Applications (ICEA), pp 28–32.