# Phishing website detection through website traffic using Machine Learning

B. Swetha[1], Kappara Sri Sai Samanvita[2], Nalla Puneeth Krishna Reddy[3], Tummala Revanth Mahindra[4]
[1]*Assistant Professor, Mahatma Gandhi Institute of Technology*
[2,3,4]*UG Student, Mahatma Gandhi Institute of Technology*

***Abstract-*** **Web sites traffic largely encourage the expansion of illegal activities on the Internet and restrict the growth of Web services. Consequently, there's been a great drive for the development of methodical approaches to discourage consumers visiting these kinds of websites. Our suggestion is to use a learning-based method to divide websites into two categories: high and low. Our approach does not access the content of the websites; it just analyzes the Uniform Resource Locator. Consequently, it removes the chance of exposing users to browser-based vulnerabilities and run-time latency. With the help of learning algorithms, our system performs better in terms of generality and coverage with the blacklist service.**

**The website URLs are divided into two classes:**

**High: Secure websites offering standard services**

**Low: Websites try to overwhelm consumers via advertisements or other content, such deceptive surveys.**

***Keywords:*** **Machine Learning Classification, Phishing Detection, Random Forest Algorithm.**

## I. INTRODUCTION

The Internet has given many people access to never-before-seen levels of convenience when it comes to managing their investments and finances, but it has also opened up new avenues for large-scale, low-cost fraud. Since there are now more safeguards against technological penetration than there were in the past, fraudsters can now control users rather than hardware or software systems. Phishing is perhaps extensively practiced Internet frauds. Its primary emphasis is on the stealing of private, confidential information like credit card numbers and passwords.

Attacks using phishing have two kinds.:

- efforts to trick victims into disclosing their secrets by posing as reliable sources has an actual need for this kind of data.
- efforts to acquire information by infecting victims' computers with malware.

The particular malware utilized in phishing attempts is not covered in this thesis and is a topic of study for the malware and virus communities. The study focus of this thesis is on phishing assaults that propagate by tricking users and the term ‘phishing attack’ will become accustomed to refer to this type of attack. In cutting-edge digital age, the proliferation of phishing assaults poses a amazing risk to on line protection, exploiting unsuspecting clients and compromising sensitive data with alarming frequency. Proactive steps that enable consumers to use the internet as it should be might be urgently needed to address this growing issue. One revolutionary approach includes harnessing the strength of gadget mastering to stumble on doubtlessly malicious websites primarily based on their traffic styles. By studying various factors which includes traffic extent, sources, session duration, and geographic distribution, system gaining knowledge of models can determine suspicious behavior indicative of phishing attempts. These models, each work on a variety of data including relevant internet websites and fraud and can detect threats in real time, providing timely warnings and protection as consumers browse. Adding this time within the network through extensions or plugins provides seamless protection, alerting customers to operational risks sooner rather than falling prey to malicious systems. Furthermore, the ceaseless monitoring and maintenance of these instances provides some relief to evolving threats, increasing the resilience of the on-line security ecosystem of the internet Through new technologies and in addition to user training, including acknowledgment campaigns and secure surfing signals, this holistic approach to humans aims to empower to secure virtual identity and provide a safe on-line environment for everyone. In Previous work, An NN model that is not well-structured might not match with the training data set. On the other hand, over-constraining the algorithm to fit every point inside the training set of data can lead to overfitting of the algorithm. One way to prevent is to avoid the problem of overfitting is to rebuild the neural network model by adjusting few parameters, incorporating additional neurons into the hidden layer, or maybe extending the network to a new layer. A neural network with few hidden neurons might not function well enough for modeling the complexity and diversity of data. Conversely, however, networks with an excessive number of concealed neurons can overestimate details. However, at particular point the model cannot be changed further, so procedure ought to be stopped. Therefore, an adequate margin of error ought to be determined when building each NN model, which in alone is regarded as an issue because it’s challenging to figure out an adequate margin of rate in advance. As an illustration, the creator of the model might specify an acceptable margin of error value that cannot be reached, causing the prototype to remain at local minima, or occasionally the model creator might specify an acceptable margin of error value that can be further improved. Here, we encounter issues like the dataset will take some time to load, the procedure is inaccurate and It is going to examine gradually.

## II. LITERATURE SURVEY

C-RAN, or Cloud Radio Access Network, is thought of as a facilitator of fifth-generation (5G) cellular networks, as it lowers operating expenses while boosting resource assignment adaptability and agility. However, depending on traffic requirements, its adoption leads to unused resources or dissatisfied users. It is complex and cannot be scalable. Furthermore, it is inflexible since it possesses a central architecture that may not be compatible with changes in websites [1].

In a study by Petluri and Al-Masri, the current time series for website traffic Google's forecasting dataset was used to project future Wikipedia article traffic. Website operators could gain with predicted Traffic on the web in several methods, like figuring out how to load balance cloud-based web sites efficiently, estimating the next few years trends using previous data, and comprehending user behavior. RNN model was utilized in this study, although it has limitations such as complexity and being prone to overfitting [2].

Jain and Gupta presented a ML-based method in order to detect phishing by extracting hyperlink content from HTML source code. The proposed method classifies hyperlink-specific elements to twelve unique classes & is utilized in the training of ML algorithms. While language is independent, this approach may not be sufficient to detect more sophisticated phishing attacks that involve dynamic content inserted by JavaScript or other client-side technologies [3].

In another study, Shelatkar et al. web traffic time series forecasting using ARIMA and LSTM RNN models was discussed. By estimating the amount of network traffic and showing that on a dashboard in the present time, information can efficiently conveyed. However, limitation lies in the assumption of stationarity in forecasting methods like ARIMA and LSTM, It might not apply to actual online traffic data, showing non-stationary patterns [4].

## III. PROPOSED SYSTEM

Based on the observation of Lexical Features that many unlawful websites have different URLs than authorized websites. We can capture the property for categorization purposes by analyzing lexical features. Initially, we take off the hostname and path from the URL, that we take out a bundle of words (strings separated by '?', '/', '.', '=', ' )’, and '-'. In mean time discovered a longer URL, additional levels (separated by dots), additional tokens in the path and domain, and a longer token are preferred by the phishing website. Furthermore, popular brand names can be used as tokens on malware and phishing websites to make them appear trustworthy rather than just second-level domain tokens. Given that malicious and phishing websites can employ IP addresses directly to hide questionable URLs, which is extremely uncommon in benign cases, [5]. The phishing site was also discovered to have many sensitive word tokens. These security-sensitive terms are looked for, and the binary value is added to our features [5]. Malicious websites always seem to be less well-liked than legitimate ones. As a result, one could see website popularity as an important factor. In accordance with the finding that harmful websites are consistently registered with less trustworthy hosting centers, host-specific features have been developed. Here,

1. Every URL in the Dataset has a label.
2. SVM and Random Forest are two algorithms that are supervised. we trained them with the scikit-learn.

## IV. SYSTEM ARCHITECTURE

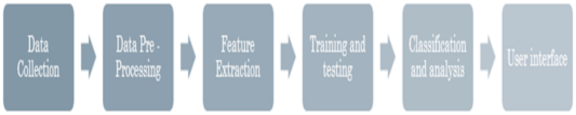


Figure1: System Architecture

Data collection: Collecting the website traffic data.

Data Pre-Processing: Arrange the chosen data by cleaning, formatting, and extracting samples.

Feature Extraction: Extract relevant features from the URLs. These features may include lexical features (length of the URL, presence of certain characters, etc.). Structural features like number of subdomains, path length, etc. Presence of security-sensitive words and other domain-specific features.

Training and testing: A training set of data is used to train those with supervised learning models.

Classification and analysis: Identify the patterns and evaluate the performance.

User interface: Develop a user interface for users to interact with the system. Users can input a URL, and the risk classification will be provided by the system.

## V. MODULES

Data Collection Module: In this module we collect detailed data about website traffic from various sources. This includes accessing logs, web server records, and other related data stores for storing all kinds of data. The data collected includes details like a page loads, unique visits, first visits and URLs. Through the integration of different datasets, we ensure an in-depth understanding of network traffic patterns and behaviors.

Data Classification Module: After the data collection process is completed, the collection of dataset is basically divided into two sub-modules: the training set and the testing set. Using the training set, to train supervised learning models, while testing set is reserved for evaluating these models' performance. This module uses numerous segmentation techniques to segment web traffic data into groups based on pre-defined characters. By classifying data systematically, we enable models to identify patterns more efficiently.

Data preprocessing modules: This includes steps to ensure efficiency and accuracy before loading data into classification models This includes steps such as organizing the data to standardized format, cleaning it to remove inconsistencies or abnormalities eliminated, additionally a balanced sample representation for the dataset, Normalization techniques can be used to standardize set of features.

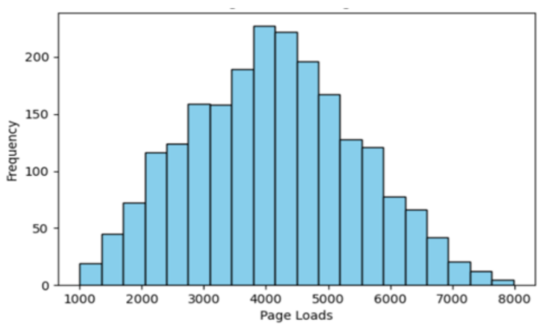
Feature extraction module: In this module we analyze the complex characteristics of URLs related to website-traffic data. By studying lexical capabilities such as URL length, keyword availability, and URL path structure. Also, our suggested system is utilized in the subsequent manner. The system can be applied to detect and prevent phishing attacks by analyzing the lexical and structural features of URLs. This application is vital for protecting individuals and organizations from financial loss, identity theft, and other forms of cybercrime. In environments where internet access needs regulation or restriction, the system can aid in content filtering. By classifying websites based on their traffic levels, administrators can implement policies to block access to low-traffic sites that have a higher probability of containing malicious or inappropriate content. This application is relevant for educational institutions, workplaces, and public Wi-Fi networks. The system can enhance targeting and campaign optimization efforts in digital marketing and advertising. Marketers can use it to identify high-traffic platforms for advertising placements, maximizing the reach and effectiveness of their campaigns. Additionally, the system can help advertisers avoid low-traffic or potentially fraudulent websites, ensuring efficient allocation of marketing budgets.

## VI. ALGORITHMS

We use machine learning algorithms to identify misleading URLs and better classify them. Each algorithm has a significant part in the overall process, contributing its unique strengths to increase the precision and reliability of our system. First, One of the fundamental algorithms for issues involving binary categorization, logistic regression used in it. Based on type it models probability of a URL being deceptive, using a logistic function to map the input variables to a binary outcome. Because relationships between input attributes and class labels are observed, logistic regression offers a straightforward but efficient framework for distinguishing between legitimate and malicious URLs. Furthermore, utilized are random forest classifiers, for their robustness and scalability when working with substantial datasets. Random forests by creating clusters of decision trees and combining their predictions can better capture complex relationships in the data and reduce unnecessary correlations. Our classification and generalization performance of the unseen data is enhanced by this cluster-based technique. Support vector machines (SVMs) offer another powerful technique for binary classification, particularly well-suited for datasets with high-dimensional feature spaces. The goal of this approach is to identify the ideal hyperplane it increases the space across data points from distinct classes. This margin-based approach enables SVMs to achieve high classification precision as well as resilience to noise, making them valuable assets in our URL detection framework. Finally, AdaBoost classifiers use the concept of continuity to combine multiple weak learners into one strong classifier. This one concentrates on model robustness, improving overall classification accuracy by retraining models on the same data and adjusting their weights based on how successfully they execute. This policy of increasing variability enforces strengthening the robustness and predictive power of our classifiers, especially in unbalanced or noisy environments. By exploiting the strengths of these algorithms, we are able to design a system that is comprehensive and scalable in fake URL detection classification. The unique features of the algorithm work together to enable the system to detect malicious URLs accurately and reduces online cybersecurity risks We do this by carefully selecting and combining algorithms.

## VII. RESULTS

Figure 2: Page Loads vs Frequency



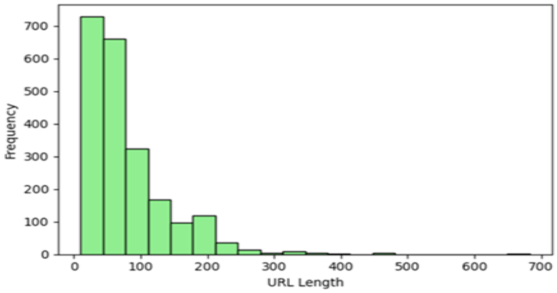


Figure 3: URL Length vs Frequency

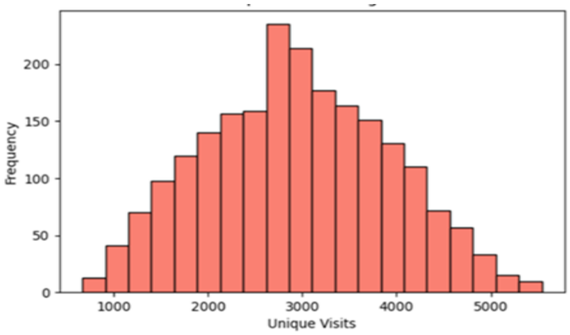


Figure 4: Unique Visits vs Frequency

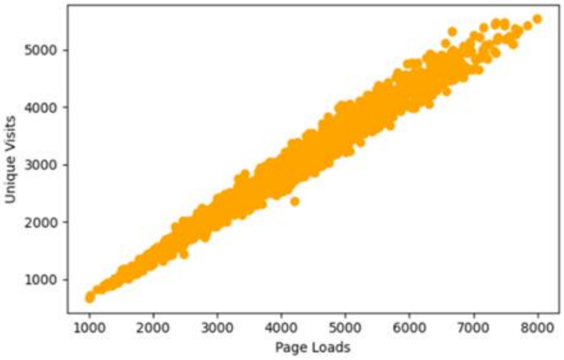


Figure 5: Page Loads vs Unique Visit

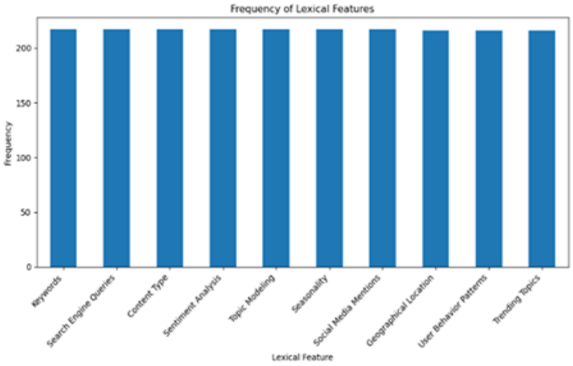


Figure 6: Lexical Features vs Frequency

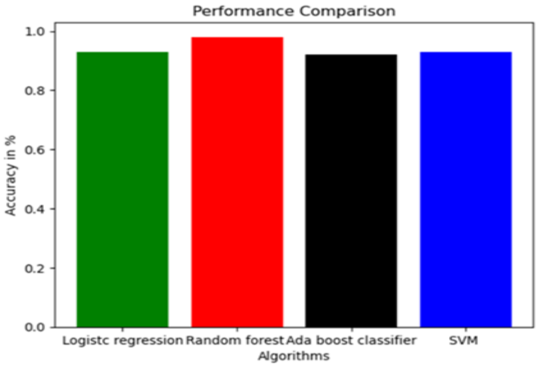


Figure 7: Algorithms vs Accuracy

The study classified URLs as phishing or legitimate using a number of machine learning methods. Logistic Regression, Support Vector Machine (SVM), AdaBoost Classifier, and Random Forest are the techniques employed. After extensive training and evaluation, the following results were obtained: Of all the algorithms, Random Forest had the best accuracy, achieving 98.5% accuracy. Logistic regression showed strong performance with 93.375% accuracy. Support Vector Machine (SVM)) showed competitive accuracy and achieved 92 0.44% accuracy. AdaBoost Classifier gave 92% accuracy. These results highlight the effectiveness of machine learning algorithms in detecting phishing URLs.

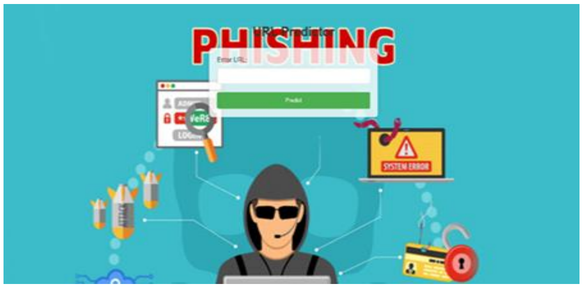
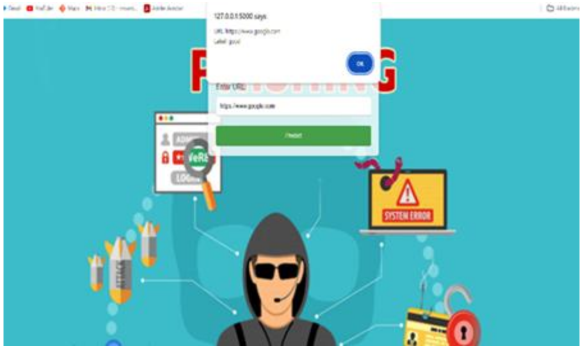


Figure 8: User Interface



127.0.0.1:5000 says

URL: https://www.google.com

Label: good

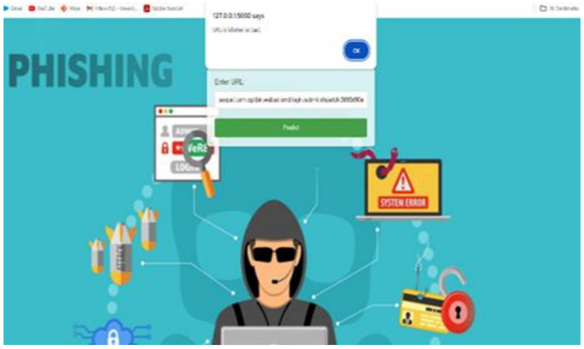
OK

Enter URL:

https://www.google.com

Predict

Figure 9: Good Case refers to positive case i.e. legitimate website.



127.0.0.1:5000 says

URL is labeled as bad

OK

Enter URL:

[illegible]

Predict

Figure 10: Bad Case refers to negative case i.e. phishing site.

## VIII. CONCLUSION

This study investigated the efficacy of machine learning algorithms in detecting fraudulent URLs, a critical task in contemporary threat detection. The primary objective was to create a robust detection system capable of accurately distinguishing between legitimate and deceptive URLs, thereby mitigating the risks associated with malicious online activities. Through rigorous experimentation and analysis, the performance of several ML models, like Logistic Regression, Support Vector Machine, AdaBoost Classifier and Random Forest, was evaluated. Each algorithm underwent meticulous training and evaluation on a diverse dataset comprising both genuine and deceptive URLs. Findings underscored the exceptional performance of Random Forest, which emerged as the most effective model having an accuracy percentage of 98.5%. Its robustness in detecting fraudulent URLs highlights its potential like a foundation in modern threat detection frameworks. Furthermore, Logistic Regression exhibited commendable performance, achieving an accuracy of 93.375%. Its interpretability and simplicity make it a useful instrument for identifying subtle patterns in URL data. SVM and AdaBoost Classifier also demonstrated respectable accuracy rates of 92.44% and 92%, respectively. Their ability to discern complex patterns and adapt to evolving threats makes them viable candidates for inclusion in detection systems. Overall, this study contributes to the growing body of research in threat detection by providing insights into the effectiveness of ML algorithms in detecting deceptive URLs. As online threats continue to evolve, the development of robust detection systems remains paramount in safeguarding individuals and organizations against malicious activities.

## IX. FUTURE SCOPE

The study has showcased the effectiveness of ML algorithms in discerning deceptive URLs, providing valuable observations into realm of cybersecurity. However, this exploration merely scratches the surface of a vast landscape ripe for further investigation. Future endeavours could delve into the realm of deep learning techniques, like RNNs and CNNs, to capture intricate patterns and relationships within URLs. Additionally, integrating behavioural analysis methodologies, like user browsing behaviour and clickstream analysis, could bolster detection capabilities by identifying anomalous interactions and adapting to emerging threats. Furthermore, enhancing the Machine learning models' interpretability and explainability, in addition to fortifying their resilience against adversarial attacks, stands as pivotal objectives to engender trust and reliability in detection systems. Moreover, cross-domain generalization and transfer learning methods offer avenues to broaden the applicability and versatility of detection models across diverse environments. By embarking on these paths of inquiry, the trajectory of cybersecurity research can continue its evolution towards more sophisticated and robust detection systems, safeguarding online users and organizations from malicious URLs.

## REFERENCE

[1] R. Guerra-Gómez, S. R. Boqué, M. García-Lozano and J. O. Bonafé, "Machine-Learning based Traffic Forecasting for Resource Management in C-RAN," 2020 European Conference on Networks and Communications (EuCNC), Dubrovnik, Croatia, 2020, pp. 200-204, doi: 10.1109/ EuCNC 48522. 2020. 9200958.

[2] N. Petluri and E. Al-Masri, "Web Traffic Prediction of Wikipedia Pages," 2018 IEEE International Conference on Big Data (Big Data), Seattle, WA, USA, 2018, pp. 5427-5429,doi: 10.1109/ BigData. 2018.8622207.

[3] Jain, A.K., Gupta, B.B. A machine learning based approach for phishing detection using hyperlinks information. J Ambient Intell Human Comput 10, 2015–2028 (2019). https://doi.org/10.1007/s12652-018-0798-z

[4] Shelatkar, Tejas & Tondale, Stephen & Yadav, Swaraj & Ahir, Sheetal. (2020). Web Traffic Time Series Forecasting using ARIMA and LSTM RNN. ITM Web of Conferences. 32. 03017.10.1051/itmconf/20203203017.

[5] A. R. Mohammed, S. A. Mohammed and S. Shirmohammadi, "Machine Learning and Deep Learning Based Traffic Classification and Prediction in Software Defined Networking," 2019 IEEE International Symposium on Measurements & Networking (M&N), Catania, Italy, 2019, pp. 1-6, doi: 10.1109/IWMN.2019.8805044.

[6] Akash Mahanand, Prathibha Prakash, Anjuna Devaraj. "Deep Learning-based Hybrid Technique for Forecasting Web Traffic", 2023 Fifth International Conference on Electrical, Computer and Communication Technologies (ICECCT), 2023

[7] Liu Haotian, Xiang Pan, and Zhengyang Qu. "Learning-based Malicious Web Sites Detection using Suspicious URLs." Department of Electrical Engineering and Computer Science, Northwestern University, IL, USA.

[8] Subashini, A & K, Sandhiya & Saranya, S & Harsha, U. (2019). Forecasting Website Traffic Using Prophet Time Series Model. International Research Journal of Multidisciplinary Technovation. 1. 56-63. 10.34256/irjmt1917.

[9] D. Sikka and C. N. S. Vinoth Kumar, "Website Traffic Time Series Forecasting Using Regression Machine Learning," 2023 IEEE 12th International Conference on Communication Systems and Network Technologies (CSNT), Bhopal, India, 2023, pp. 246-250,doi: 10.1109/CSNT57126.2023.10134631.