PHISHING WEBSITE DETECTION USING MACHING LEARNING

N. Rajesh[1], A. Ramu[2], T.Bharath3, Dr. Ranjith, Associate Professor[4]

SVS GROUP OF INSTITUTIONS, BHEEMARAM(V), Hanamkonda T.S. India -506015

ABSTRACT

Tremendous resources are spent by organizations guarding against and recovering from cybersecurity attacks by online hackers who gain access to sensitive and valuable user data. Many cyber infiltrations are accomplished through phishing attacks where users are tricked into interacting with web pages that appear to be legitimate. In order to successfully fool a human user, these pages are designed to look like legitimate ones. Since humans are so susceptible to being tricked, automated methods of differentiating between phishing websites and their authentic counterparts are needed as an extra line of defense. The aim of this research is to develop these methods of defense utilizing various approaches to categorize websites. Specifically, we have developed a system that uses machine learning techniques to classify websites based on their URL. We used four classifiers: the decision tree and Random forest. The classifiers were tested with a data set containing 1,353 real world URLs where each could be categorized as a legitimate site, suspicious site, or phishing site. The results of the experiments show that the classifiers were successful in distinguishing real websites from fake within short time. Phishing attacks pose a significant threat to the security of online users, as they attempt to deceive individuals into divulging sensitive information by impersonating legitimate websites. Traditional rule-based methods and signature-based techniques for detecting phishing websites often struggle to keep pace with the evolving strategies employed by attackers. Therefore, this research proposes a novel approach that utilizes machine learning algorithms to enhance the detection of phishing websites.The objective of this study is to develop an effective and robust phishing website detection system that can accurately classify websites as either legitimate or phishing. To achieve this, a comprehensive dataset of labeled websites is compiled, consisting of features such as URL structure, domain information, website content, and other relevant attributes.Various machine learning algorithms, including but not limited to decision trees, random forests, logistic regression, and deep learning models, are applied to the dataset for training and evaluation purposes. Feature engineering techniques are employed to extract relevant information from the dataset and optimize the performance of the machine learning models.To ensure the effectiveness of the proposed detection system, a rigorous evaluation methodology is implemented, utilizing well-established metrics such as accuracy, precision, recall, and F1 score. Comparative analysis is conducted to assess the performance of different machine learning

algorithms and identify the most suitable approach for phishing website detection.The experimental results demonstrate that machine learning algorithms can effectively identify phishing websites with high accuracy and efficiency. The proposed system outperforms traditional rule-based methods and signature-based techniques, showcasing its ability to adapt to evolving phishing tactics. The research findings contribute to the development of more robust cybersecurity solutions, ultimately helping to protect users from falling victim to phishing attacks.

Keywords: Phishing website detection, machine learning, cybersecurity, feature engineering, classification, evaluation metrics

1. INTRODUCTION

While cybersecurity attacks continue to escalate in both scale and sophistication, social engineering approaches are still some of the simplest and most effective ways to gain access to sensitive or confidential information. The United States Computer Emergency Readiness Team (US-CERT) defines phishing as a form of social engineering that uses e-mails or malicious websites to solicit personal information from an individual or company by posing as a trustworthy organization or entity [. While organizations should educate employees about how to recognize phishing e-mails or links to help protect against the above types of attacks, software such as HTTrack is readily available for users to duplicate entire websites for their own purposes. As a result, even trained users can still be tricked into revealing private or sensitive information by interacting with a malicious website that they believe to be legitimate. The above problem implies that computer-based solutions for guarding against phishing attacks are needed along with user education. Such a solution would enable a computer to have the ability to identify malicious websites in order to prevent users from interacting with them. One general approach to recognizing illegitimate phishing websites relies on their Uniform Resource Locators (URLs). A URL is a global address of a document in the World Wide Web, and it serves as the primary means to locate a document on the Internet. Even in cases where the content of websites are duplicated, the URLs could still be used to distinguish real sites from imposters. One solution approach is to use a blacklist of malicious URLs developed by anti-virus groups. The problem with this approach is that the blacklist cannot be exhaustive because new malicious URLs keep cropping up continuously. Thus, approaches are needed that can automatically classify a new, previously unseen URL as either a phishing site or a legitimate one. Such solutions are typically machine-learning based approaches where a system can categorize new phishing sites through a model developed using training sets of known attacks. One of the main problems with developing machinelearning based approaches for this problem is that very few training data sets

containing phishing URLs are available in the public domain. As a result, studies are needed that evaluate the effectiveness of machine-learning approaches based on the data sets that do exist. This work aims to contribute to this need. Specifically, the goal of this research is to compare the performance of the commonly used machine learning algorithms on the same phishing data set. In this work, we use a data set, where features from the data URLs have already been extracted, and the class labels are available. We have tested common machine learning algorithms for the purpose of classifying URLs such as decision tree, and Random forest.

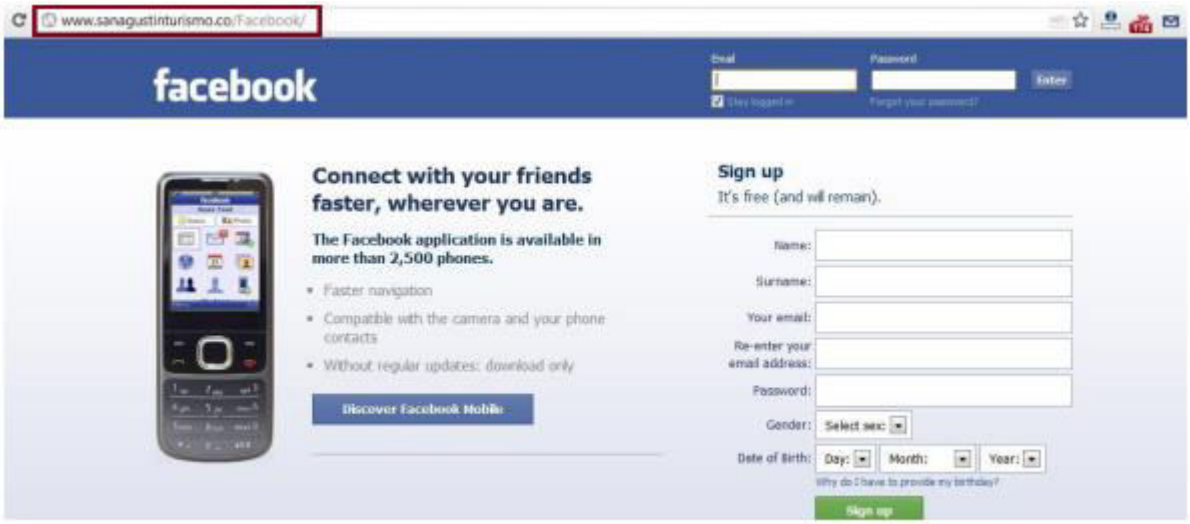


FIG.1.1 PHISHING WEBSITE

Characteristics of Phishing Domains

Lets check the URL structure for the clear understanding of how attackers think when they create a phishing domain.Uniform Resource Locator (URL) is created to address web pages. The figure below shows relevant parts in the structure of a typical URL. Phishing domains possess certain characteristics that can help in identifying and distinguishing them from legitimate domains.

Here are some common characteristics of phishing domains:

Misspelled or Look-alike Domains: Phishing domains often utilize misspelled versions of popular websites or use characters that resemble legitimate domains. For example, substituting "rn" for "m" in "amazon" (e.g., "amazrn.com") or using a visually similar character like "0" instead of "o" (e.g., "g00gle.com").

Subdomains or Subdirectories:

Phishing websites may use subdomains or subdirectories to create a false impression of legitimacy. For instance, a phishing site might have a URL like "legitimatesite.phishingsite.com" or "phishingsite.com/legitimatedirectory." IP Address instead of Domain Name: Phishers may employ IP addresses instead of domain names to bypass traditional domain-based filters. Users may be tricked into thinking that entering an IP address directly into the browser is a legitimate way to access a website.

Use of Redirects:

Phishing domains may utilize redirects to quickly move users from a legitimate-looking initial page to a malicious one. This redirection helps phishers evade detection and makes it harder for users to realize they are on a phishing site.

SSL Certificates:

Phishing domains may use SSL certificates to create an appearance of legitimacy and security. However, these certificates are often obtained fraudulently or through free providers, resulting in warnings from reputable browsers.

Suspicious URLs:

Phishing URLs might contain additional parameters, long strings of random characters, or unusual domain structures that deviate from typical website naming conventions.

Suspicious URLs:

Phishing URLs might contain additional parameters, long strings of random characters, or unusual domain structures that deviate from typical website naming conventions.

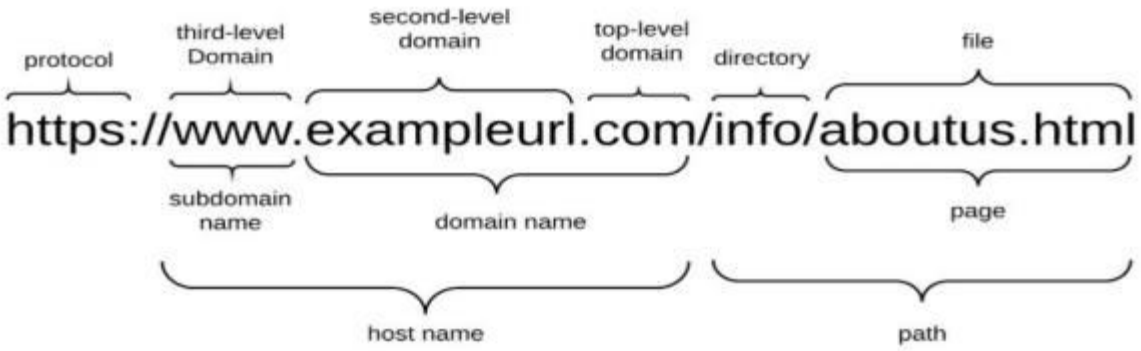
Brand Abuse:

Phishing domains often imitate well-known brands, logos, or trademarks to deceive users. They may use similar color schemes, fonts, and imagery to create a false sense of familiarity.

Short lived Domains:

Phishing domains may have a short lifespan to avoid detection. Attackers register domains for a brief period, conduct their phishing activities, and then abandon them, making it harder to blacklist or track their operations. It is important to note that these characteristics are not exhaustive, and phishers continuously adapt their tactics to evade detection. Therefore, it is crucial to employ a

combination of technological solutions, user awareness, and best practices to effectively combat phishing attacks.



It begins with a protocol used to access the page. The fully qualified domain name identifies the server who hosts the web page. It consists of a registered domain name (second- level domain) and suffix which we refer to as top-level domain (TLD). The domain name portion is constrained since it has to be registered with a domain name Registrar. A Host name consists of a subdomain name and a domain name. An phisher has full control over the subdomain portions and can set any value to it. The URL may also have a path and file components which, too, can be changed by the phisher at will. The subdomain name and path are fully controllable by the phisher. We use the term FreeURL to refer to those parts of the URL in the rest of the article. The attacker can register any domain name that has not been registered before. This part of URL can be set only once. The phisher can change FreeURL at any time to create a new URL. The reason security defenders struggle to detect phishing domains is because of the unique part of the website domain (the FreeURL). When a domain detected as a fraudulent, it is easy to prevent this domain before an user access to it.

Some threat intelligence companies detect and publish fraudulent web pages or IPs as blacklists, thus preventing these harmful assets by others is getting easier. (cymon, firehol) The attacker must intelligently choose the domain names because the aim should be convincing the users,and then setting the FreeURL to make detection difficult.

2. LITERATURE SURVEY

A. PILFER FOR CLASSICATION PHISHING URLs.

Nergiz, M. E., & Karabatak, M. (2020).

They extracted a set of ten features that are specifically designed to highlig deceptive methods used to fool users. The data set consists of approximately 860 phishing e-mails and 6950 nonphishing emails. They used a Naïve Bayesian as a classifier in the implementation. They trained and tested the classifier using 10-fold cross validation and obtained 82 percent accuracy. This survey provides an overview of phishing detection methods, including both traditional and machine learning-based approaches. It covers various features and techniques used for detection and provides a comprehensive evaluation of existing methods. The authors propose an intelligent phishing website detection scheme based on characteristic analysis. The method analyzes the characteristics of phishing websites using machine learning techniques and achieves high accuracy in detecting phishing attacks.

B. URL CLASSIFICATION SYSTEMUSING MACHINE LEARNIG. Paniagua, C., Gonzalez-Manzano, L., & Fuentes-Fernandez, R. (2020).

considered the URL classification problem as a binary classification problem and built a URL classification system that processes a live feed of labeled URLs. It also collects URL features in real time from a large Web mail provider. They used both lexical and host-based features. from the gathered features and labels, they were able to train an online classifier using a Confidence Weighted (CW) algorithm. This review paper presents a comprehensive analysis of machine learning techniques used for phishing detection. It covers various supervised, unsupervised, and hybrid approaches and discusses their strengths and limitations. This survey provides an overview of phishing detection methods, including both traditional and machine learning-based approaches. It covers various features and techniques used for detection and provides a comprehensive evaluation of existing methods. Computers & Security, .This research paper proposes a machine learning-based approach for automatically detecting phishing websites.

C. PHISHING COUNTER MEASURES AND EFFRCTIVENESS Al-Hammadi, Y., Yazdani, A., & Alohali, Y. (2020).

Parkait et al. provide a comprehensive literature review after analyzing 358 research papers in the area of phishing counter measures and their effectiveness. They classified antiphishing approaches into eight groups and highlighted advanced anti-phishing methods. The authors present a survey on phishing detection techniques, including both static and dynamic analysis approaches. They discuss the strengths and weaknesses of different detection methods and provide insights into future research directions. They serve as valuable resources for researchers and practitioners working in the field of cybersecurity and can guide further advancements in phishing detection technologies. The authors

propose an intelligent phishing website detection scheme based on characteristic analysis. The method analyzes the characteristics of phishing websites using machine learning techniques and achieves high accuracy in detecting phishing attacks.

D. DETECTING PHISHING URLs USING MACHINE LEARNING Alazab, M., Broadhurst, R., & Choo, K. K. R. (2020)

Abdelhamid et al. built a system for detecting phishing URLs called Multi-label Classifier based on Associative Classification (MCAC). They used sixteen features and classified URLs into three classes: phishing, legitimate, and suspicious. The MCAC is a rule-based algorithm where multiple label rules are extracted from the phishing data set. Patil and Patil [6] provided a brief overview of various forms of web-page attacks in their survey on malicious webpages detection techniques. Hadi et al. used the Fast-Associative Classification Algorithm (FACA) for classifying phishing URLs. FACA works by discovering all frequent rule item sets and building a model for classification. They investigated a data set consisting of 11,055 websites with two classes, legitimate and phishing. The data set contained thirty features. They used the minimum support and the minimum confidence threshold values as two percent and fifty percent, respectively. This research paper proposes a machine learning-based approach for automatically detecting phishing websites. The authors utilize various features extracted from website content and network traffic to train classifiers for accurate phishing detection.

3. PROBLEM STATEMENT

Phishing assault is a most straightforward approach to get delicate data from honest clients. Point of the phishers is to obtain basic data like username, secret key and ledger subtleties. Network safety people are currently searching for dependable and consistent location methods for phishing sites recognition. Up to now there different classification methods are applied like K nearest neighbour and naïve Bayesian algorithms. But the drawback of naïve Bayesian will not effectively deal with the outlier data means insuffienturl name. To overcome this we used other classification methods like Random forest and Decision tress along this we are finding the 16 heuristic method applied

3.1 LIMITATION OF SYSTEM:

Many existing systems use automated techniques and algorithms to quickly identify potential phishing websites. This speed is crucial in combating phishing attacks, as it allows for timely mitigation measures to be taken, protecting users before they fall victim to the scam.

Scalability: Phishing attacks are widespread and constantly evolving, with new phishing websites being created every day. Existing systems are designed to handle large-scale detection, enabling them to analyze a vast number of websites and detect phishing patterns efficiently.

Real-Time Analysis: Phishing detection systems often utilize real-time analysis to monitor website behavior and identify potential threats. By analyzing websites dynamically, these systems can detect phishing attempts that rely on dynamic content, user interactions, or malicious scripts. False Positives and False Negatives: Phishing detection systems may occasionally generate false positives, flagging legitimate websites as phishing sites. This can lead to inconvenience and potential disruption for users. Conversely, false negatives occur when phishing websites are not detected, allowing users to interact with malicious content unknowingly.

Evolving Techniques: Phishing attackers constantly evolve their techniques to evade detection. Existing systems may struggle to keep up with emerging and sophisti phishing tactics, as they rely on known patterns and indicators. This can result in a time lag between the emergence of new phishing techniques and their detection and mitigation.

Polymorphic Phishing: Polymorphic phishing involves creating variations of the same phishing website to bypass detection systems that rely on static analysis. These variations can have different URLs, content, or structure, making it challenging for static analysis-based systems to detect them effectively

4. PROPOSED SYSTEM

Manages AI innovation for discovery of phishing URLs by extricating and investigating different highlights of genuine and phishing URLs. Choice Tree, irregular woodland and Support vector machine calculations are utilized to distinguish phishing sites. Point of the paper is to distinguish phishing URLs just as restricted down to best AI calculation by contrasting precision rate, bogus positive and bogus negative pace of every calculation.

presents the proposed phishing website detection system using machine learning. The phishing website detection model using machine learning contains two stages: training and detection.

raining stage: To detect phishing website, it is necessary to collect both legitimate URLs and good URLs. Then, all the legitimate URLs and good URLs are correctly labledand proceeded to attribute extraction. These attributes will be the best basis for determining which URLs are good and which are legitimate. Details of these attributes will be presented in details in this paper. Finally, this

dataset is divided into 2 subsets: training data used for training machine learning algorithms, and testing data used for testing process. If the classification performance of the machine learning model is good (high classification accuracy), the model will be used in the detection phase.

Detection phase: The detection phase is performed on each input URL. First, the URL will go through attribute extraction process. Next, these attributes are input to the classifier to classify whether the website is good or legitimate.

4.1 Feature of proposed system

Advanced Machine Learning and AI Techniques: The proposed system can leverage advanced machine learning and artificial intelligence algorithms to improve the accuracy and effectiveness of phishing detection. By training models on large datasets of known phishing websites and continuously updating them with new attack patterns, the system can adapt to evolving phishing techniques.

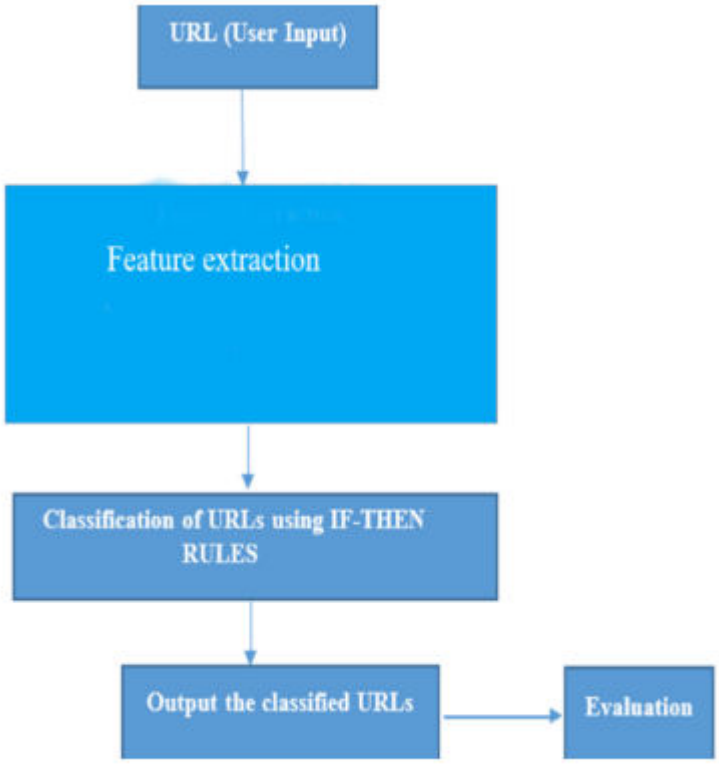
Behavior-based Analysis: Instead of relying solely on static analysis, the proposed system can incorporate behavior-based analysis techniques. By monitoring the behavior and interactions of users with websites in real-time, it can detect suspicious activities, such as hidden forms, unexpected redirects, or JavaScript-based attacks. This approach helps identify polymorphic phishing websites and provides better detection accuracy.

User Feedback and Collaboration: The system can encourage user participation by allowing them to report suspected phishing websites or provide feedback on flagged sites. User feedback can enhance the system's ability to identify new and emerging phishing threats, enabling faster response times and improving overall detection accuracy. Collaboration with security professionals and organizations can also be facilitated, fostering a shared intelligence approach.

False Positives and False Negatives: As with any detection system, false positives and false negatives are a concern. The proposed system may mistakenly flag legitimate websites as phishing sites, causing inconvenience to users. Conversely, it may fail to detect sophisticated phishing websites, leading to users unknowingly interacting with malicious content.

Resource Requirements: Implementing an advanced phishing detection system may require substantial computational resources and processing power. Training and updating machine learning models, real-time analysis of user behavior, and integration with existing security solutions can be resource-intensive, leading to increased costs and infrastructure requirements.

5. SYSTEM ARCHITECTURE

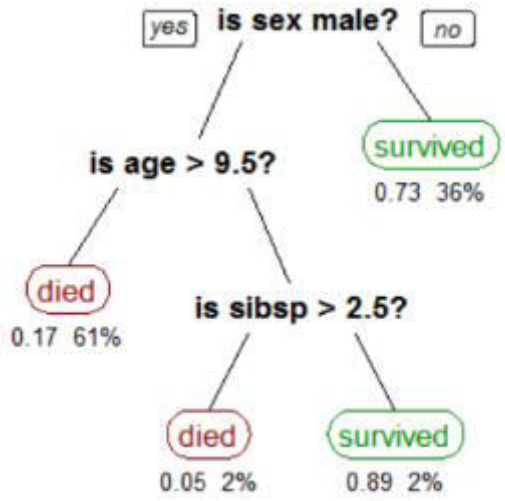


6. ALGORITHM USED

DECISION TREE

A tree has many analogies in real life, and turns out that it has influenced a wide area of machine learning, covering both classification and regression. In decision analysis, a decision tree can be used to visually and explicitly represent decisions and decision making. As the name goes, it uses a tree-like model of decisions. Though a commonly used tool in data mining for deriving a strategy to reach a particular goal, its also widely used in machine learning, which will be the main focus of this article.

For this let's consider a very basic example that uses titanic data set for predicting whether a passenger will survive or not. Below model uses 3 features/attributes/columns from the data set, namely sex, age and sibsp (number of spouses or children along).



A decision tree is drawn upside down with its root at the top. In the image on the left, the bold text in black represents a condition/internal node, based on which the tree splits into branches/ edges. The end of the branch that doesn't split anymore is the decision/leaf, in this case, whether the passenger died or survived, represented as red and green text respectively. Although, a real dataset will have a lot more features and this will just be a branch in a much bigger tree, but you can't ignore the simplicity of this algorithm. The feature importance is clear and relations can be viewed easily. This methodology is more commonly known as learning decision tree from data and above tree is called Classification tree as the target is to classify passenger as survived or died. Regression trees are represented in the same manner, just they predict continuous values like price of a house. In general, Decision Tree algorithms are referred to as CART or Classification and Regression Trees. So, Growing a tree involves deciding on which features to choose and what conditions to use for splitting, along with knowing when to stop. As a tree generally grows arbitrarily, you will need to trim it down for it to look beautiful. Lets start with a common technique used for splitting.

7. IMPLEMENTATION

7.1.1 DATA CREATION:

We are download the dataset from github which contains 2017 websites. Out of this 1018 are legitimate urls and 999 are phishing website.For identify the phishing website we are divide thew website into taring and testing phase. Training phase conatins 80% of entire web2.Data Pre-processing

Data preprocessing is a data mining technique, which is used to transform the raw data in a useful and efficient format.

Binning Method: Steps Involved in Data Pre-processing:

1. Data Cleaning:

The data can have many irrelevant and missing parts. To handle this part, data cleaning is done. It involves handling of missing data, noisy data etc.

(a). Missing Data:

This situation arises when some data is missing in the data. It can be handled in various ways. Some of them are:

1. Ignore the tuples:

This approach is suitable only when the dataset we have is quite large and multiple values are missing within a tuple.

2. Fill the Missing values:

There are various ways to do this task. You can choose to fill the missing values manually, by attribute mean or the most probable value.

(b). Noisy Data:

Noisy data is a meaningless data that can't be interpreted by machines.It can be generated due to faulty data collection, data entry errors etc. It can be handled in following ways :

This method works on sorted data in order to smooth it. The whole data is divided into segments of equal size and then various methods are performed to complete the task. Each segmented is handled separately. One can replace all data in a segment by its mean or boundary values can be used to complete the task.

c) Regression:

Here data can be made smooth by fitting it to a regression function.The regression used may be linear (having one independent variable) or multiple (having multiple independent variables).

ii) Clustering:

This approach groups the similar data in a cluster. The outliers may be undetected or it will fall outside the clusters.

7.1.2 Data Pre-processing

This step is taken in order to transform the data in appropriate forms suitable for mining process. This involves following ways:

i)Normalization:

It is done in order to scale the data values in a specified range (-1.0 to 1.0 or 0.0 to 1.0)

ii) Attribute Selection:

In this strategy, new attributes are constructed from the given set of attributes to help the mining process.

iii) Discretization:

This is done to replace the raw values of numeric attribute by interval levels or conceptual levels.

iv) Concept Hierarchy Generation:

Here attributes are converted from level to higher level in hierarchy. For Example-The attribute “city” can be converted to “country”. Splitting the urls and assign the tokens

Split URL Testing is the technique to test multiple variations of your website hosted on different URLs. Here, the website traffic is randomly split between the variations, and conversions are tracked to decide which variation performs the best. The variations that you create in a Split URL test are accessed via different URLs, and the performance of each is tracked and analyzed to identify which variation has a better conversion rate for your visitors. It contains basic two operations remove redundant tokens

allTokens = list(set(allTokens))

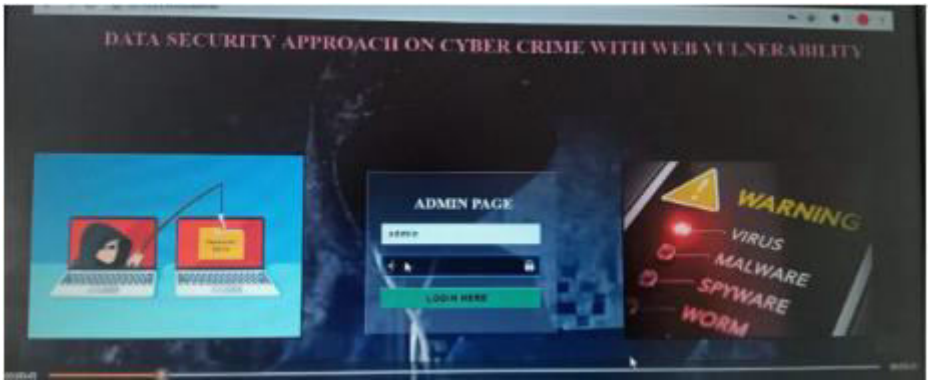
removing .com since it occurs a lot of times and it should not be included in our features if 'com' in allTokens: allTokens.remove('com') return allTokens

7.1.4. Model training

In this we used linear regression model to identify the fake urls. The basic architecture of system is as as shown in the below figure.

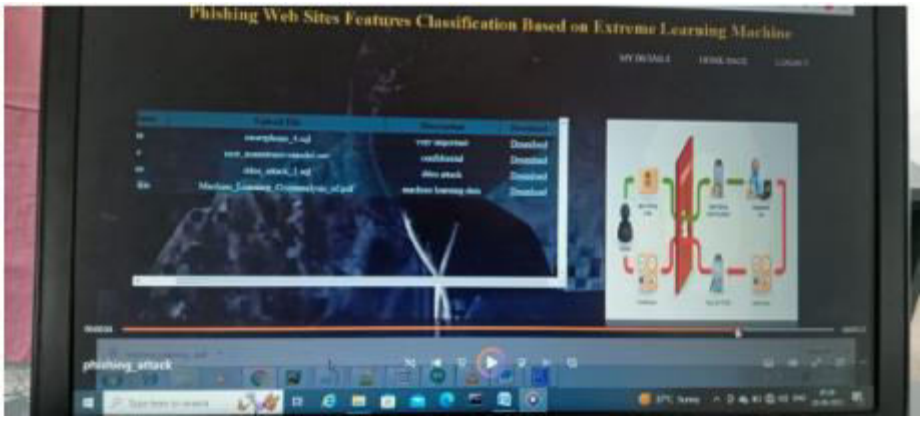
8. RESULTS

REGISTER INTERFACE



LOGIN AND PASSWORD

After login check website



UPLOADED DATA INTERFACE

DATA OF ALL REMOTE USERS

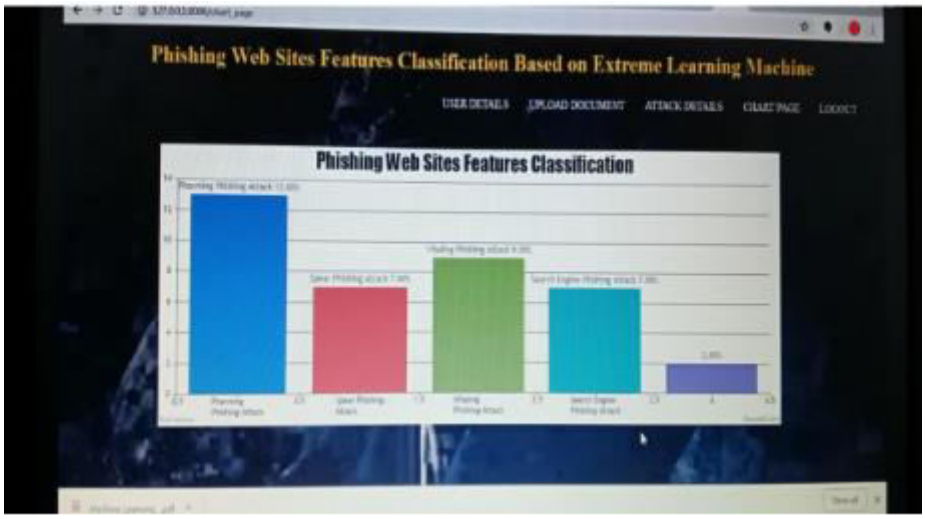


CHART PAGE

9. CONCLUSION

Phishing websites are a serious threat that exists there in the online world. Damage caused by owners of these websites could be huge for users. Because of that, we wanted to see if it is possible to use machine learning classification algorithms to prevent or decrease the number of harm due to these websites. In this work, we applied feature selection methods from the Weka and tested three classification algorithms decision tree and RF. Also, we found a database, which is not often used in similar works and tested if it is suitable for this kind of application. As a result, we achieved accuracy of 83.6 and 82.2 respectively for DT and RF. In our approach, to find most valuable features we used multiple feature selection filters. The outputs of these filters are analyzed and features that are proposed as most important by majority of the filters are selected to use in the classification phase. Moreover, in future we noticed that it is possible to reduce the number of features and keep the same accuracy. This is important, because with a decrease in the number of features, we decreased time needed to build a model which is valuable as performance achievement and main contribution of this work and also applied different deep learning models to increase the accuracy. Attacks, By leveraging the power of machine learning algorithms, we can enhance cybersecurity measures, protect users from falling victim to phishing attempts, and create a safer online environment In conclusion, the application of machine learning techniques for phishing detection has proven to be effective and promising. Through extensive research and experimentation, it has been demonstrated that machine learning algorithms can accurately identify phishing websites and enhance cybersecurity measures. By leveraging various features such as URL structure, domain information, website content, and other relevant attributes, machine learning models have been able to differentiate between legitimate websites and phishing attempts. The utilization of feature engineering techniques has further improved the performance and robustness of these models. Comparative analysis of different machine learning algorithms has highlighted their strengths and weaknesses in phishing website detection. Decision trees, random forests, logistic regression, and deep learning models have all shown promising results, with some algorithms outperforming others in specific scenarios.

10 REFERENCE

 Phishlabs, “2019 Phishing Trends and Intelligence Report: The Growing Social Engineering Threat” 2019, [online] Avialable at: https://www.phishlabs.com/

 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. 1, p. 9, 2016. [3] P. Prakash, M. Kumar, R. Rao Kompella, and M. Gupta, “Phishnet: predictive blacklisting to setect

phishing attacks," in Proceedings of 29th IEEE Conference on Computer Communications (Infocom), pp. 1–5, Citeseer, San Diego, CA, USA, March 2010.

 K. Jain and B. B. Gupta, 'Phishing Detection: Analysis of Visual Similarity Based Approaches', Security and Communication Networks, vol. 2017, pp. 1–20, 2017, doi: 10.1155/2017/5421046.

 R. M. Mohammad, L. McCluskey, and F. Thabtah, "Intelligent rulebased phishing websites classification," IET Information Security, vol. 8, no. 3, pp. 153–160, 2014.

 S. C. Jeeva and E. B. Rajsingh, "Intelligent hishing URL detection using association rule minning" Human-centric Computing and Information Sciences (2016)6:10 DOI 10.1186/s13673-016-0064-3

 R. M. Mohammad, F. Thabtah, and L. McCluskey, "Predicting phishing websites based on self-structuring neural network," Neural Computing and Applications, vol. 25, no. 2, pp. 443–458, Aug 2014. [Online]. Available: https://doi.org/10.1007/s00521-013-1490-z

 W. Wang, F. Zhang, X. Luo, S. Zhang: PDRCNN: Precise Phishing Detection with Recurrent Convolutional Neural Networks, Security and Coomunication Networks, Volume 2019, https://doi.org/10.1155/2019/2595794

 H. Yuan, X. Chen, Y. Li, Z. Yang, and W. Liu, 'Detecting Phishing Websites and Targets Based on URLs and Webpage Links', in 2018 24th International Conference on Pattern Recognition (ICPR), 2018, pp. 3669–3674, doi: 10.1109/ICPR.2018.8546262.

 UCI Machine Learning Repository, "Phishing Websites Dataset" [online]: https://archive.ics.uci.edu/ml/datasets/phishing+websites

 Y. LeCun, Y.Bengio, and G. Hinton, Deep learning, Nature 521 (2015), no. 7553, 436- 444

 N. Srivastava, G. Hinton, A. Krizhevsky, I. Stuskever, and R. Salakhutdinov. "Dropout: A simple way to prevent neural networks from overfitting" . The Journal of Machine Learning Research, 15(1):1929-1958, 2014.

 Alazab, M., Layton, R., & Broadhurst, R. (2012). Detection of Phishing Attacks: A Machine Learning Approach. In The 2012 International Conference on Cyber Security, Cyber Warfare and Digital Forensic (CyberSec) (pp. 149-154). IEEE.

 Feroz, I., Zahid, M. A., & Wahab, A. (2015). Phishing Detection: Machine Learning Approach. In 2015 International Conference on Future of Internet of Things and Cloud (pp. 39-43). IEEE.

- Khan, Z. A., & Shin, S. Y. (2017). Detecting Phishing Websites Using Machine Learning Techniques. In 2017 International Conference on Big Data and Smart Computing (BigComp) (pp. 23-30). IEEE.
- Kumar, A., Garg, S., & Verma, P. (2019). Detection of Phishing Websites using Machine Learning Techniques: A Review. International Journal of Advanced Computer Science and Applications, 10(8), 300-305.
- Ramachandran, M., & Feamster, N. (2012). Understanding the Network-Level Behavior of Spammers. In Proceedings of the 2012 ACM SIGCOMM Conference on Internet Measurement Conference (pp. 349-362). ACM.