EFFICIENT DETECTION OF PHISHING HYPERLINKS USING MACHINE LEARNING

Anshumaan Mishra[1] And Fancy[2]

[1]Department of Computer Science Engineering, SRM Institute of Science and Technology, Kancheepuram,India.

[2] Department of Information Technology, SRM Institute of Science and Technology, Kancheepuram, India.

ABSTRACT

Phishing is a type of Social Engineering cyber-attack, hackers use it to gain access to confidential credentials like bank account credentials details, details of their personal life like debit card details, social media credentials, etc. Phishing website links seem to seem just like the genuine ones and it's a tedious and troublesome task to differentiate among those websites. In this paper, features are extracted from a separate dataset of phishing and benign website URLs and then using the Machine Learning method we determine the phishing websites. We also rank the features based on the contribution of each feature used in determining the outcome of a URL link using built python libraries. Most of the phishing URLs use a large URL length when used for an attack. Hence, we proposed three machine learning models Random Forest, Support Vector Machine (SVM), Decision trees models for the efficient detection of phishing using fake URLs. The performance of the models is also compared among themselves using a confusion matrix to determine the highest performance. The implemented models have shown an accuracy of 84.81 (for Random Forest and SVM),83.96 (Decision tree)

KEYWORDS

Machine Learning, social engineering, Random Forest, SVM, Decision Trees

1. INTRODUCTION

The second-largest type of cyber-attacks is phishing attacks. They are mostly executed using Social Engineering. These attacks often lead to deception and manipulation of the victim leading them to leak their hidden data to the attacker. The usage of websites for tasks like shopping, banking, emailing is used via confidential credentials for different users. The attacker can clone these sites to create and create fake URLs that approximately mimic these types of website links. This link on choosing leads to an attacker-controlled webpage. In most cases these web pages look like professional and authorized ones, requesting individuals for sensitive data. They have many discernable differences when compared to a genuine webpage, which might be invisible to the victim who might not have sufficient technical background. As the second-largest attack in cyberspace, phishing detection or anti-phishing software have been deployed to counter the potential damage that can be done to the victims of this type of attack. There have been many frameworks that have been introduced earlier to combat the problem using machine learning.

D. Angeline, A. Arul et al.(Eds): ICICI-2021
pp. 23-33, 2021. IJCI – 2021 DOI:10.5121/ijci.2021.100204

2. RELATED WORK

This subsection shows previous work related to the topic. To begin with, old techniques like blacklisting are one of the simple ways to identify phishing websites but can't be used to find new phishing websites. It also takes a lot of time to use this method.

Current research in the detection of phishing links is classified into four groups which are the Visual Similarity-based approach, Heuristic Based approach, Fuzzy rule-based approach, Machine Learning approach.

A. Visual Similarity

In this kind of approach, a phishing website is compared with a legitimate website on the grounds of visual appearance this includes analyzing the HTML tags, Images present on the suspected page, javascript version used, etc. Eric et al [1] used a similar approach where they used the signature of a suspected phishing website is obtained and compared with a legitimate website's signature. A signature of a web page was used to capture information encompassing the images and text content present on this web page. To be more specific, It is a set of attributes that represent different aspects of a website. To detect phishing sites, they use three features: text fragments, images embedded in the page, and the overall visual appearance of the web page as made by the browser. Although they receive a negligible false positive rate on the dataset, they used to consist of only 41 phishing pages which are way too small.

B. Heuristics Based Approach

The heuristic-based approach is the second choice. This method combines different features extracted from the target pages to determine if it is a phishing or legitimate web page. The heuristic architecture of suspicious websites fits the feature set that is commonly used in phishing websites in this approach. CANTINA is a mechanism proposed by Zhang et al [2], which has proposed a framework called CANTINA which detects phishing pages by analyzing text content using the TF-IDF algorithm. However, the scheme's limitations are determined by the TF-IDF algorithm and the website's language.

C. Fuzzy Rule-based Approach

Fuzzy logic techniques take advantage of linguistic variables to represent the main phishing characteristic indicators. Maher et al [3] used fuzzy logic to detect phishing websites based on six different criteria. They have made separate layers in their method with each layer containing one or more criteria, they also calculate the phishing rate using a formula. The disadvantage of this method is that it is unable to detect zero phishing pages.

D. Machine learning-based approach

Machine learning is one of the latest approaches used by researchers to find out whether a website is a phishing site. Ankit et al [4] proposed an anti-phishing framework that uses hyperlink-specific characteristics of various machine learning algorithms. The fact that the features are built on the source code is a drawback of this strategy as the source code of the website is subject to change for malicious purposes. This could lead to an increase in false prediction.

3. PROPOSED ARCHITECTURE

In this section, we describe the proposed architecture of this paper

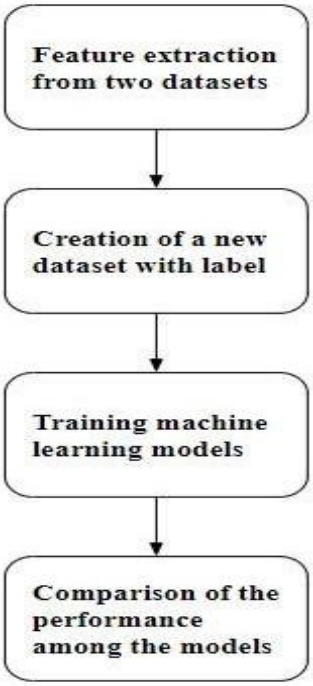

Fig-a. Showing the proposed architecture

A. Methodology

In this paper, we used two different datasets at the beginning, one dataset is obtained from Phish Tank [5] and the other is obtained from the Canadian Institute of Cybersecurity [6] which contains benign URLs. The main objective of our work is to find out malicious URLs using three different methods and compare their performance in predicting the same. We first extract eight features from phishing and benignURLs and later create a new dataset with those features as the column names and a label that shows 0 for genuine and 1 for phishing. We have used 5500 samples from each dataset to avoid imbalance in the new dataset we created. The features extracted are address bar-based.

B. Features Selected

- IP Address in URL:

  IP Address in the URL Checks for the presence of IP address in the URL. URLs may have IP addresses instead of the domain name. If an IP address is used as an alternative to the domain name in the URL, we can be sure that someone is trying to steal personal information with this URL. If the domain part of the URL has an IP address, the value assigned to this feature is 1 (phishing) or else 0 (legitimate).

- "@" Symbol in URL:

"@" Symbol in URL Checks for the presence of the '@' symbol in the URL. Using the "@" symbol in the URL leads the browser to ignore everything preceding the "@" symbol and the real address often follows the "@" symbol.
If the URL has the '@' symbol, the value assigned to this feature is 1 (phishing) or else 0 (legitimate).

- Length of URL:

  This feature helps in finding the length of a given URL. Phishers can use a long URL to hide the skeptical part in the address bar. In this project, a URL is labeled as phishing if it is longer than or equal to 54 characters, which is the threshold length. If the URL duration is greater than 54, the value assigned to this function is 1 (phishing) otherwise 0 (legitimate).

- Depth of URL:

  Depth of URL Computes the depth of the URL. This feature calculates the number of sub-pages in the given URL based on the '/'. The value of the feature is numerical based on the URL. A given URL can have a lot of depth.

- Redirection "//" in URL:

  This feature looks for the "//" symbol in a URL. If the URL route contains the character "//," the user will be redirected to another website. The "//" in the URL's position is discovered. We discovered that if the URL starts with "HTTP," the "//" should be placed at the sixth spot. If the URL uses "HTTPS," however, the "//" should be in the seventh position. If the "//" appears somewhere in the URL other than after the protocol, which is extremely rare, the value assigned to this attribute is 1 (phishing) or 0 (no phishing) (legitimate).

- "http/https" in Domain name:

  HTTPS is the latest and the most secured version of HTTP. HTTPS is used on many modern-day websites. However, attackers might try to use this protocol to their advantage by tricking the user into believing that the attacker-controlled website is legitimate. If 'http/https' is present at the beginning of a URL then the number assigned to this feature is 1 as it is considered Phishy else it is assigned 0.

- Using URL Shortening Services “TinyURL”

  Some services are present which shorten the length of the URL. These services are present on the 'World Wide Web. Using URL Shortening Services helps the attacker craft a URL to make the victim believe that the crafted URL is genuine. This is accomplished through an "HTTP Redirect" on a brief name, which links to the webpage that features a long URL. If the URL is using Shortening Services, the worth assigned to the present feature is 1 (phishing) alternatively 0 (legitimate).

- Prefix or Suffix "-" in Domain:

  Prefix or Suffix "-" in Domain Checking the presence of '-' in the domain part of URL. The dash symbol is rarely used in legitimate URLs. Phishers tend to add prefixes or suffixes separated by (-) to the domain name so that users feel that they are dealing with a legitimate website. If the URL has the '-' symbol within the domain as a part of the

URL, the worth assigned to the present feature is 1 (phishing) alternatively 0 (legitimate).

3. RESULTS AND CONCLUSION

To evaluate our overall performance of our models we have compared them based on confusion matrix, Feature Ranking, and Accuracy

A. Confusion Matrix

One of the best ways to identify the performance of a classifier is through a confusion matrix. To compare the performance of the models that were used we have used a confusion matrix to identify the
True Positive (TP): The number of URLs that have been marked as Phishing URLs.
False Negative (FN): The number of URLs that have been incorrectly determined to be Legitimate URLs.
True Negative (TN): The number of URLs that have been determined to be Legitimate URLs.
False Positive (FP): The number of URLs that have been mistakenly marked as Phishing URLs

Table 1. Confusion matrix

		Actual Values	
		Positive	Negative
Predicted Values	Positive	True Positive	False Positives
	Negative	False Negative	True Negative

Fig. b. Example of a confusion matrix

Precision indicates how many of the cases that were correctly expected turned out to be positive.

It is calculated using this formula

$$\text{Precision} = \frac{TP}{TP+FP}$$

Recall indicates how many of the real positive cases our model was able to correctly predict.

It is calculated using this formula.

$$\text{Recall} = \frac{TP}{FN+TP}$$

F1-Measure or F score

F-Measure is a method for combining precision and recall into a single measure that captures both characteristics. The formula is as follows.

$$\text{F - Measure} = \frac{2 \times \text{Precision} \times \text{Recall}}{Precision+Recall}$$

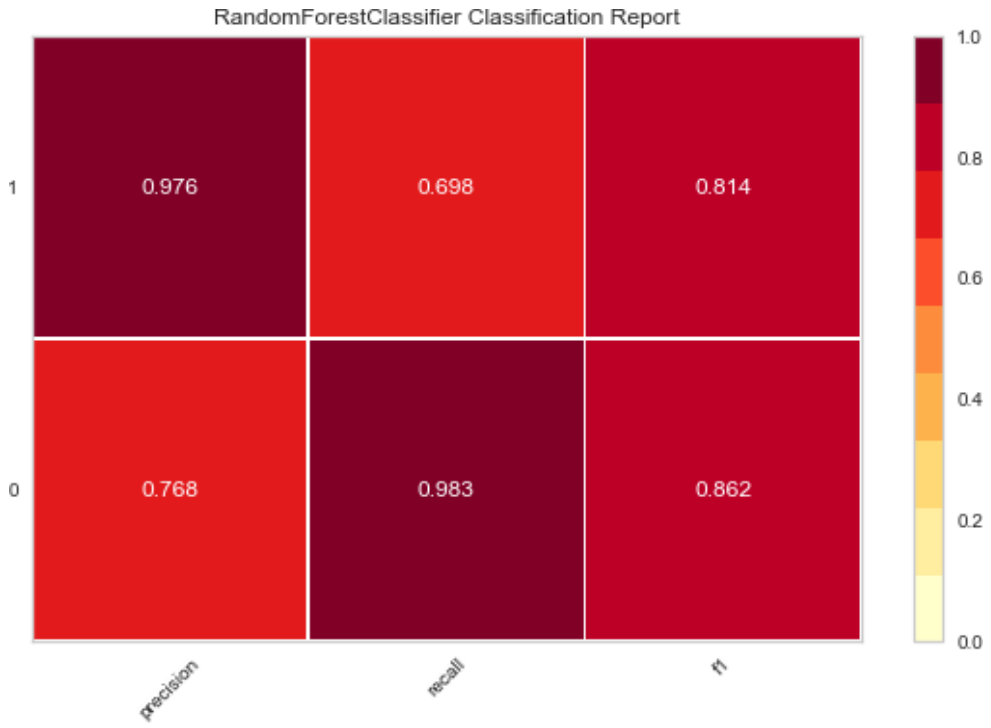

Fig-1 Random Forest Confusion Matrix

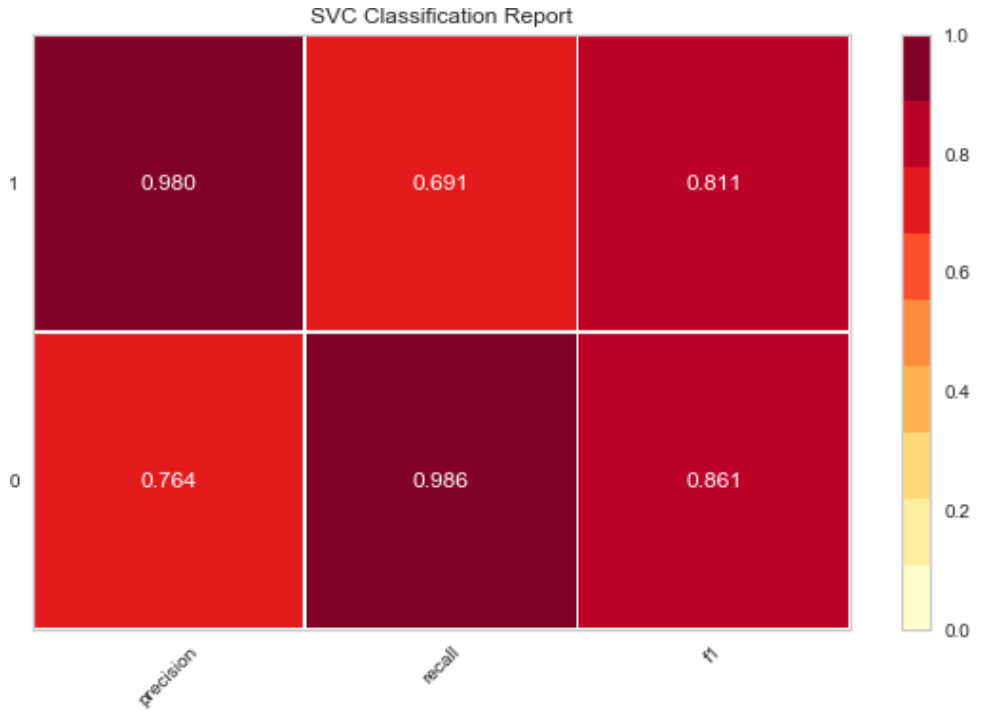

Fig-2 Support Vector Machine Confusion Matrix

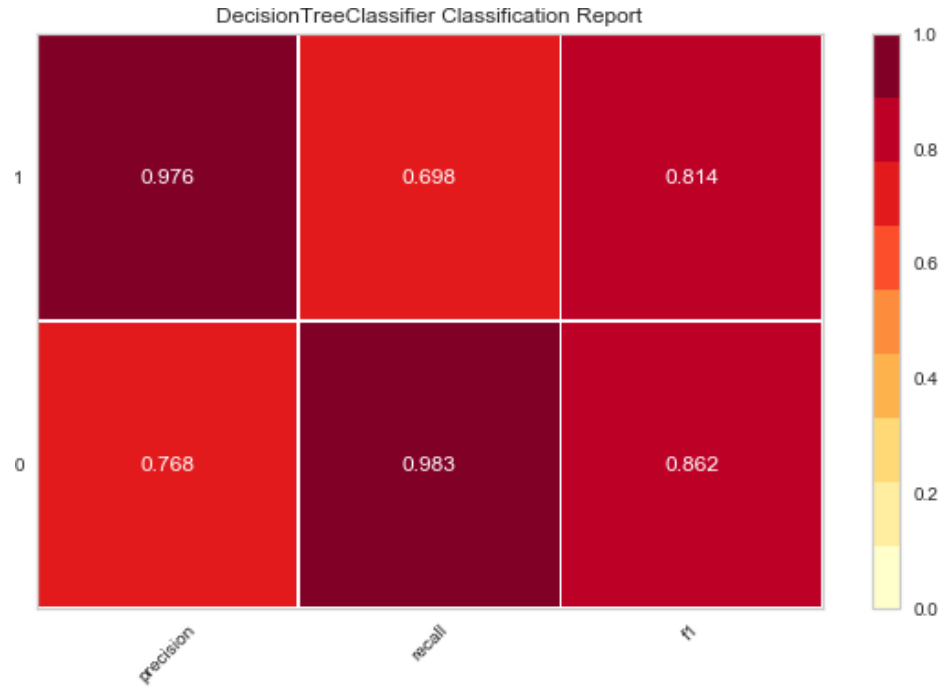

Fig-3 Decision Confusion Matrix

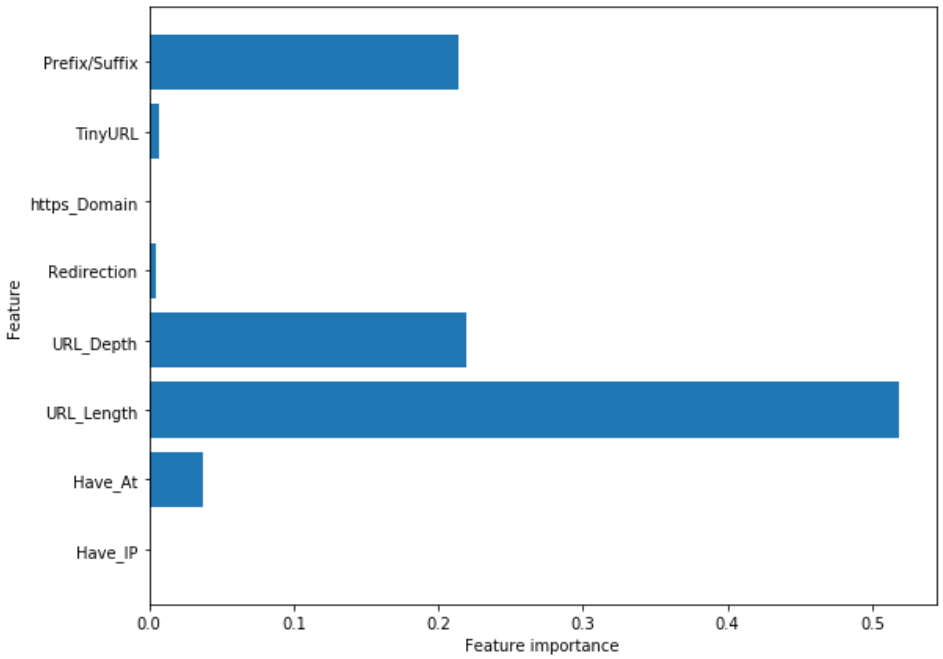

Fig – 4. Random Forest Feature ranking Graph

B. Feature Ranking

We have also ranked the eight features that were extracted to see the feature that is present in every phishing URL. This helps us understand the features that have contributed the most and the ones that contributed the least when a certain URL was tested.

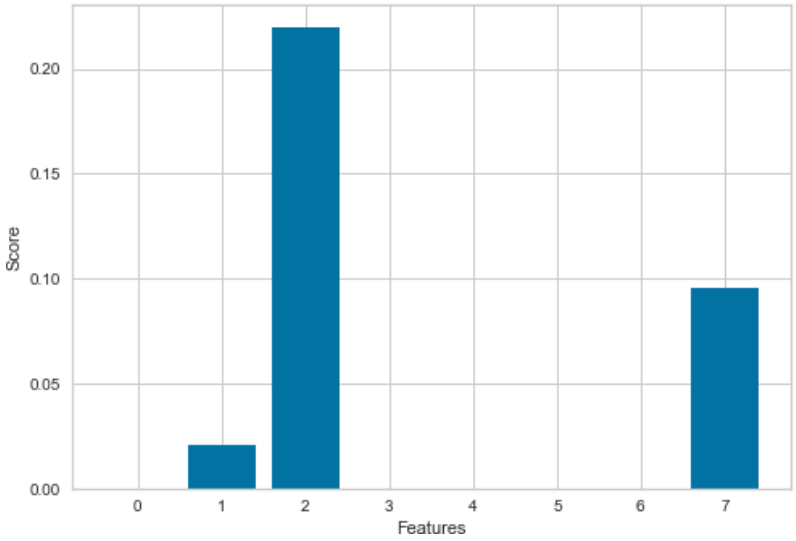

Fig – 5. SVM Feature ranking Graph

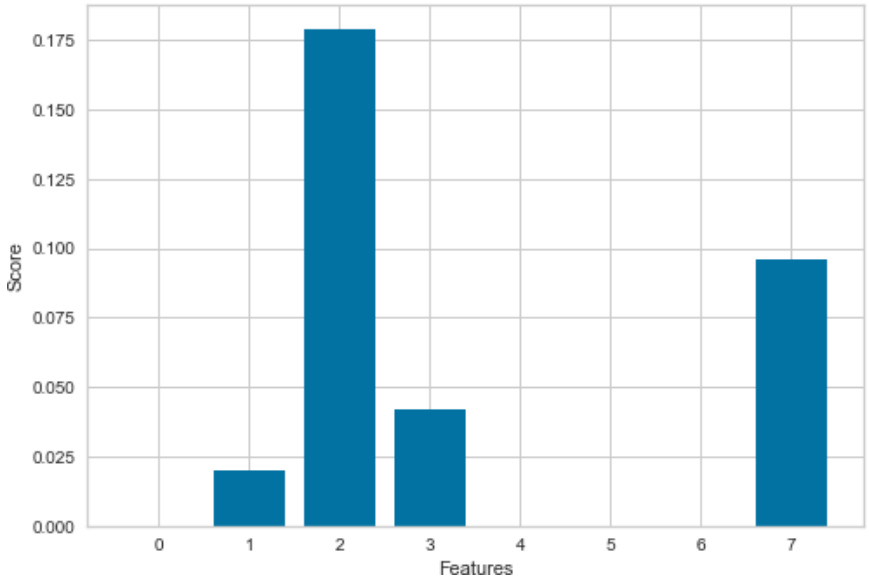

Fig – 6. Decision Trees Feature Ranking Graph

In all the graph figures a score is present for each feature, this score helps in the identification of contribution provided by a certain feature used in evaluating the outcome, Higher the score larger the contribution, and a low score means low contribution. In Fig.4 the URL length feature has a very high score, the HTTP domain feature along with the IP address feature has the least score which tells us about their importance while determining the outcome of a URL. Fig. 5 and Fig. 6 show that feature 2 (URL length) followed by feature 7(Prefix-suffix) has provided an important contribution compared to other features. The score of features 1(presence of '@' symbol) is approximately similar in Fig 5 and 6, other features have not provided any contribution in Fig.5, however in Fig.6 feature 3 (URL Depth) has shown some contribution. Fig. 4 shows the importance of many features used in the model Features 5(http_domain) and 0(Have_IP) have shown the least importance in all three figures.

C. Accuracy

The accuracy score is used to provide the True positive percentage for test data. The accuracy for the three models was determined using this formula

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+TN+FN}$$

Accuracy	**Models**		
	Random Forest	**SVM**	**Decision Trees**
	0.841	0.839	0.841

Fig – 7. Table for the comparison of model performance

Random forest and Decision trees show similar performance while SVM shows the least performance.

In this paper, we have prepared a method to evaluate a set of malicious and benign URLs using feature extraction and three machine learning models. There are only two outcomes of this approach, phishing and benign. We extracted eight features based on the address bar of a given URL. These features have helped us in creating a dataset containing both phishing and benign URLs, we then used machine learning algorithms to determine whether a URL is malicious or benign. Furthermore, we discovered the most contributing features used in the identification process. The Random Forest model contributed nearly all the features unlike SVM, Decision Trees which have shown the contribution of 3 and 4 features. The URL length feature was the only feature providing the highest contribution in all three models. The accuracy table shows that SVM has the lowest performance, Random Forest and Decision trees have performed quite similarly.

REFERENCES

[1] Medvet, Eric, Engin Kirda, and Christopher Kruegel. "Visual-similarity-based phishing detection." Proceedings of the 4th international conference on Security and privacy in communication networks. 2008.

[2] Zhang, Yue, Jason I. Hong, and Lorrie F. Cranor. "Cantina: a content-based approach to detecting phishing web sites." Proceedings of the 16th international conference on World Wide Web. 2007.

[3] Aburrous, Maher, et al. "Intelligent phishing website detection system using fuzzy techniques." 2008 3rd International Conference on Information and Communication Technologies: From Theory to Applications. IEEE, 2008.

[4] Jain, Ankit Kumar, and Brij B. Gupta. "A machine learning-based approach for phishing detection using hyperlinks information." Journal of Ambient Intelligence and Humanized Computing 10.5 (2019): 2015-2028.

[5] Phishing URLs Dataset available at https://www.phishtank.com

[6] Dataset available at:https://www.unb.ca/cic/datasets/url-2016.htm

AUTHORS

Mrs. Fancy is working as Assistant Professor in the Department of Computer Science Engineering, SRM Institute of Science and Technology

Anshumaan Mishra is doing CSE in the Department of Computer Science Engineering, SRM Institute of Science and Technology