Date: 12/03/2021

Phishing URL Detection by Machine Learning Classifiers

A major Project Report Submitted in Fulfilment

of the Requirements of the Course

Engineering Analytics

INDE6360

for

Semester 1 (Fall 2021)

by

SAI SRI HARSHA AMBATI - 2090997

harshaambati2020@gmail.com

SRIKANTH REDDY NIMMALA - 2100527

srikanth.2378@gmail.com

SAIVARDHAN REDDY NOMULA - 1799472

saivardhan.kf48@gmail.com

Table of Contents

1. Abstract

URL phishing is a growing threat where cybercriminals create fake website and URL which looks similar to genuine website to obtain sensitive information for malicious use, such as usernames, passwords, or banking details. Spotting these phishing websites is typically a challenging task because phishing is mainly a semantics-based attack, that mainly focuses on human vulnerabilities, not network or software vulnerabilities. This research paper aims at classifying phishing and legitimate URLs using machine learning classifiers by various feature selection methods. The dataset used in this project is collected from Phish tank website which consists of 5000 phishing URL's, while another contain 5000 legitimate URL's contain Alexa websites provided by University of New Brunswick. The dataset comprises 5000 phishing URLs and 5000 legitimate URLs. Address bar based, Domain based, and HTML-JavaScript based features are extracted from the URL and assigned values 0 to phishing, 1 to legitimate. The classification algorithms logistic regression, K-Nearest Neighbors, Support Vector Machines, and tree-based boosting algorithms like decision tree, Randomforest, XGBoost, CatBoost are tuned with hyperparameter values and to select the best models that achieve exceptional performance with high accuracy, ROC, F1 score and precision. The observational results have shown that the optimized CatBoost achieves the highest accuracy, precision and ROC.

2. Introduction

Phishing is an attempt by scammers to obtain subtle information such as credit card numbers, login credentials, and passwords from targeted individuals. Time and again, the attacker uses a fake domain address, designs a website that replicates the original website of an organization carefully, sending fraudulent communications to the people to deceive them. These types of communications apply different kinds of threats, scare users to take some actions that the scammer wants, which redirects the users to a web page designed to imitate the login page of an actual website. Phishing messages are designed to look genuine, and often copy the format used by the organization the scammer is pretending to represent, including their branding and logo.

Alternatively, you might be told that a large purchase has been made in a foreign country and asked if you authorized the payment. If you reply that you didn't, the scammer will ask you to confirm your credit card or bank details so the 'bank' can investigate. In some cases, the scammer may already have your credit card number and ask you to confirm your identity by quoting the 3 or 4-digit security code printed on the card.

There are a few methods used by attackers such as content injection inserting malicious content into the legitimate site and other methods include Cybersquatting and Typosquatting. Cybersquatting is the process of URL hijacking. The attacker buys the domain name of an already established company that does not have a website related to the domain name. Typosquatting refers to buying a website URL similar to a legitimate website but containing a typographical error. An example of this is google.com and goggle.com. Often, internet users make typing errors while entering the website URL which is exploited by attackers. Besides these, the attacker may also choose to manipulate the URL by altering the sub-domain names, query lengths, adding redirect requests, or making the URL excessively long. Since

phishing data is easily available in phishing databases such as PhishTank and OpenPhish, once a website is suspected of being related to phishing, the attacker can easily modify the website URL by altering the sub-domain names to make a new website. Therefore, there is a need for an intelligent method for identifying phishing URLs and reduce phishing attacks. Data mining techniques can help in the classification of website URLs into phishing and legitimate URLs.

3. Literature Survey

Now, different journals, conferences have different studies and research for the detection of phishing websites one of the approaches in [1] implemented using the Random Forest technique they had obtained high accuracy for this model. Multiple classification algorithms implemented in the [2] which includes SVM, AdaBoost, and Naive Bayes. These algorithms are divided into three stages using 21 fixed yet unique features. Then a two-step procedure takes place with the help of another classification algorithm but the problem here is the time consumed and the complexity involved, overheads involved and the performance issues and hence this isn't an optimal method. The approach that was proposed in [3] is the classification algorithm they have selected and implemented using K-nearest neighbors and Support vector machine and random forest algorithm and the author also proposes the development of a Chrome Extension for identifying phishing websites and the reported accuracy is 93 percent.

According to the Institute of Research Engineers, phishing detection techniques are separated into blacklist-based and heuristic-based approaches [4]. The blacklist-based approach maintains a database list of addresses (URLs) of those sites that are classified as malicious. If a user requests a site that is included in this list, the connection is blocked. The blacklist-based approach has the advantages of easy implementation and a low false-positive rate [4]; however, it has a flaw in that it cannot detect phishing sites that are not listed in the database, including temporary sites. With a heuristic approach [5][6], a signature database of known attacks will be built and used by antiviral systems or intrusion detection systems to scan a web page. The websites will be considered phishing websites if their heuristic patterns match signatures in the database. But, the main drawback of this approach is that signatures can be easily hacked by attackers, and hence, the heuristics fail to detect novel attacks. Besides that, the updating rate of the signature database is usually slower than the pace at which attackers overwhelm victims with novel attacks, resulting in many exploits.

4. Proposed Methodology

The objective is to build a classifier that uses a series of URL’s, which is sent to feature extractor to acquire important features for training the models. Following the feature extraction, seven classifiers are used to train the models. Hyperparameter tuning is done on the base models and best parameters are chosen to get optimized model.

4.1 Data Collection

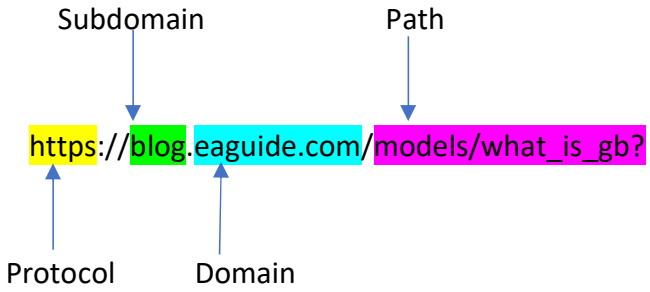
Phish Tank provides an open API to pull phishing URL data and the data is updated in PhishTank on an hourly basis. Dataset provided by the University of New Brunswick has 35000 legitimate URLs. We have considered margin value of 10000 URL's and randomly picked 5000 phishing and 5000 benign samples and combined dataset is used for feature extraction and modelling.

The below table shows the datasets utilized for this project.

Type	Source	URL
Phishing	Phish tank	https://www.phishtank.com/developer_info.php
Benign	University of New Brunswick	https://www.unb.ca/cic/datasets/url-2016.html.\

5. Feature Selection

Uniform Resource Locator (URL) is an address for a webpage where the information is displayed to the requestor. Each webpage has its own unique URL. Original webpages are forged and designed to steal user's confidential information. A typical URL is composed of different parts which is shown below.



To differentiate phishing URL from legitimate URL we extracted some features from the URL that helps in classifying the URL. We selected some of the features which play a crucial role in predicting phishing URLs.

5.1 Address Bar Based Features

- **Domain:** The domain part of the URL is extracted which helps in extracting other features and this feature is dropped while training the model.
- **IP address in URL:** If the URL contains IP address instead of domain name then we can consider it's a phishing webpage which is used to steal confidential information.
- **Having @ in URL:** If the URL contains @ character in URL then the legitimate website at the left will be dropped and browser considers right part, and user will be directed to the phishing webpage.

- **URL Length:** Phishers can take advantage of long URL to hide the phishing webpage address. We calculated length of all the URLs in dataset and considered average length to classify phishing and legitimate URL.
    - URL length >= 54 → Phishing
    - URL length < 54 → Legitimate
- **URL Depth:** This is to count number of subpages in URL with the help of / in URL.
- **Redirection:** If a URL contains // then it is redirected to some other webpage. If a URL begins with http and https then // has to be in 6^{th} and 7^{th} position respectively other than this if // exists anywhere in the URL part, then it can be considered as phishing URL.
- **http/https in domain:** If the http/https is present in domain or path then the URL can be considered as phishing
- **Tiny URL:** URL Shortener shortens long address to tiny/short which are readably and easy to remember. If someone clicks on short URL it may redirect to phishing webpage.
- **Prefix or Suffix separated by "-" to Domain:** "-"character is rarely used in legitimate URL's domain part. Phishers add "– "in domain part so that users feel its legitimate website.

5.2 Domain based Features

- **DNS Record:** WHOIS database helps to lookup for the DNS record. If the URL domain is not found in the WHOIS then we can consider it as phishing URL.
- **Web Traffic:** Frequently and large number of user visit increases web traffic and whereas phishing websites are for a short period of time in the network and Alexa database may not recognize them. When a webpage traffic is very low and not found in Alexa database can be suspected and can be considered as phishing.
- **Domain Age:** Domain age is difference between Domain creation and expiration date. Most of the legitimate domain's age is more than 12 months. As phishing URL are for short period of time and the domain age is less than 12 months can be considered for phishing

5.3 HTML and Java Script based features

- **Iframe:** iframe is an HTML based feature used inside a webpage to load another HTML document (another webpage) inside it. Phishers can use iframe to hide the frame border. The frameborder causes the browser to render a visual delineation. If the iframe is blank or there is no response, then we can consider it as phishing or legitimate.
- **Right Click:** Right click function is disabled by using java script to restrict users to user to view webpage source code. we will search for event "event. button==2" in the webpage source code and check if the right click is disabled. If there right click is disabled, then we can consider it as phishing.
- **Web Forward:** webpage forwarding is another feature that helps in differentiating phishing and legitimate URL. From the dataset we found most of the legitimate URL's are redirected once whereas phishing URL redirected more than 4 times. If the website redirected more than 4 times can be considered as phishing URL.

We developed a python script to import a live dataset from PhishTank and Extracted 15 features from a combined dataset of 10,000 URLs that includes 5000 phishing URLs and 5000 legitimate URLs in the form of 0's and 1's except Domain and URL's Depth.

6. Implementation

6.1 Data Preprocessing

Domain feature was dropped from the dataset as it is a object type and does not have any significance towards the model training. The data is clean and is in numerical format with no null values. No encoding is performed here. This leaves us with 14 features and a target column. In the feature extraction, the features are just concatenated without any shuffling lead to model overfitting. So, we shuffled the data evenly before the splitting of training and test sets for modeling.

6.2 Hyperparameter Tuning

In machine learning, a model is represented by model parameters. Parameters are learned during the process of training a model whereas hyperparameters are set manually to help guide the learning process before training the machine learning model and these hyperparameters are external to the model as they cannot be changed during the training and also hyperparameters control the learning process and determine the value of model parameters that a machine learning algorithm ends up in learning. These are input parameters to the model and hyperparameter tuning avoids the manual process of running the model with each set of parameters this helps in selecting the best hyperparameters to obtain the best performance which has the best accuracy and this entire process we are not aware of the optimal parameter that the model selected to obtain the best performance. The process of choosing the best hyperparameters for the model is called hyperparameter tuning. Two of the simplest and common optimization algorithms that can be used are Random search and Grid search. Here we preferred random search over grid search as it is best for discovery and getting hyperparameter combinations, faster execution, and training a wide range of values. Random search optimal hyperparameter selection helps us in a better model with minimal error and high accuracy.

6.3 Machine learning algorithms used for classification

We trained below listed seven different machine learning algorithms

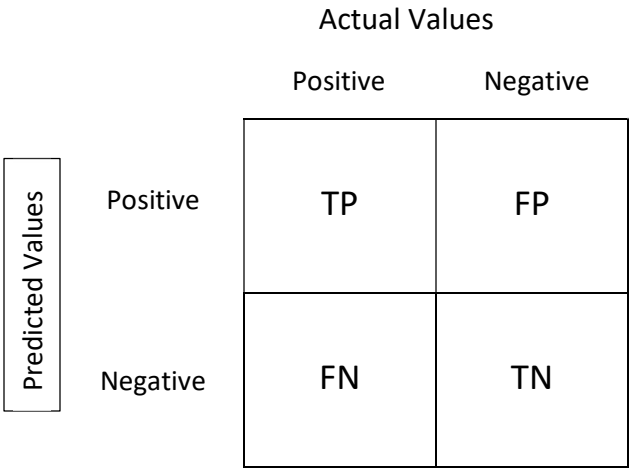
- Logistic Regression
- K- nearest neighbors
- Support Vector Machine
- Decision Tree
- Random Forest
- XG Boost
- Cat Boost

CatBoost (Categorical &Boosting) is recent open-sourced machine learning from Yandex. This library works well with multiple categories of data such as audio, text, image including historical data. We can use CatBoost without any explicit pre-processing to convert categories into numbers as it converts categorical values into numerical values internally and reduces chances of overfitting.

6.4 Evaluation Metrics

Metrics help us understand how a classifier performs and the Classification metrics are calculated from true positives (TP), False positive (FP), False negative (FN) and True negative (TN). These four outcomes are plotted on confusion matrix for better visualization of the performance of the model.

6.5 Confusion Matrix



The confusion matrix display the actual and predicted values Confusion matrix here in our classification is 2x2 matrix.

True Positive (TP): correctly predicted phishing class as phishing

True Negative (TN): correctly predicted legitimate class as legitimate

False Positive (FP): Incorrectly predicted legitimate class as phishing

False Negative (FN): Incorrectly predicted phishing class as legitimate

Accuracy

Accuracy is one of the metrics which gives the fraction of correct predictions for the test data. It can be calculated by dividing the number of correct predictions by the number of total predictions.

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

Precision

Precision gives the fraction of correctly identified phishing URL out of all predicted as phishing

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

Recall (Sensitivity or True positive rate)

Recall gives the fraction we correctly identified as phishing out of all phishing

-

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

F1 Score

It is defined as the harmonic mean of the model's precision and recall, calculated as below

$$\text{F1 Score} = 2 * \frac{Precision * Recall}{Precision + Recall}$$

ROC/AUC Curve

ROC curve shows the performance of the classification model. Its another metric that we are calculating the "Area Under the Curve (AUC) of "Receiver Characteristic Operator" (ROC). ROC curve plots the sensitivity and specificity for a model.

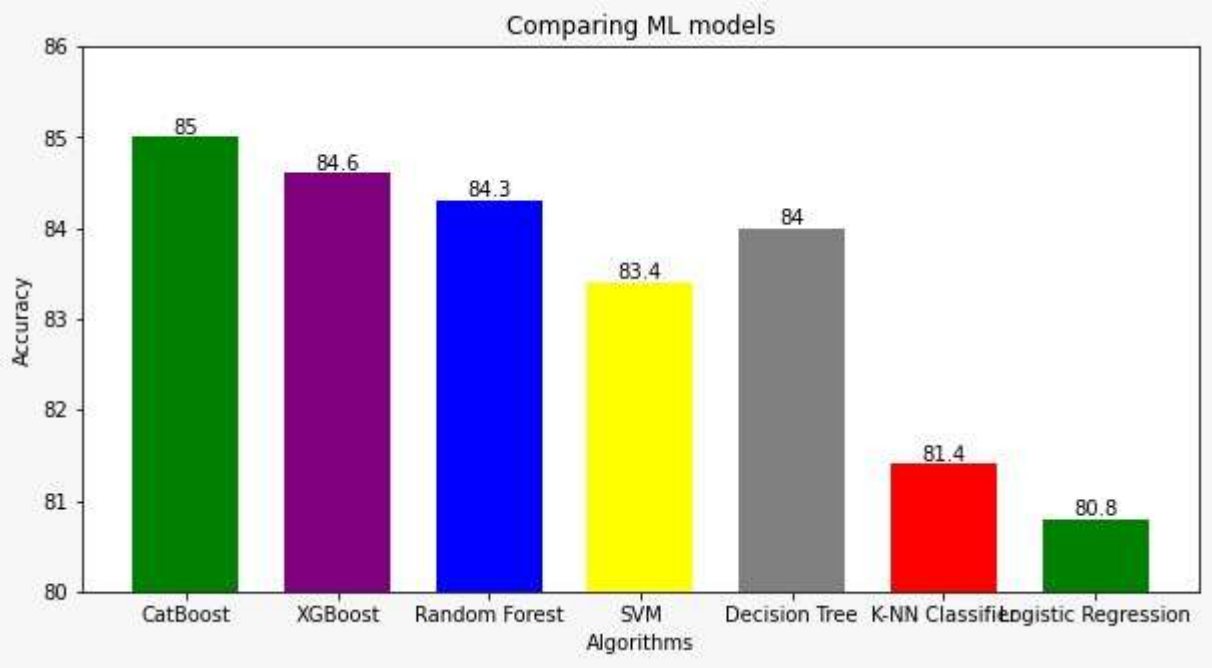
7. Model Evaluation

Models are trained with a split of data where 80% of data is utilized for training and 20% of data for test accuracy.

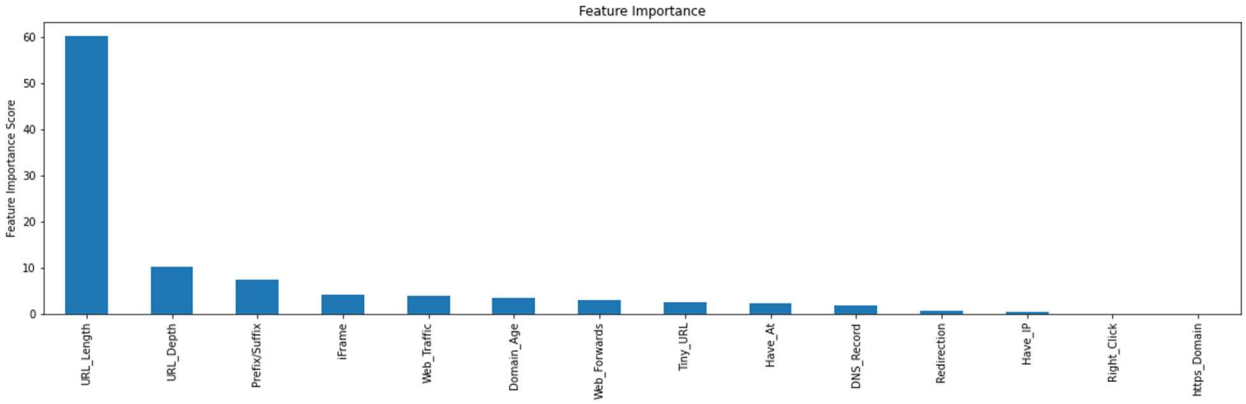
80:20 Ratio					
ML Classifiers	**Accuracy**	**Precision**	**F1_Score**	**Recall**	**ROC**
Logistic Regression	80.8	92.96	77.53	66.49	80.76
KNN	81.4	48	43	38.93	81.47
SVM	83.4	96.8	80.4	68.71	83.26
Decision Tree	84	94.4	82.35	73.03	84.1
Random Forest	84.3	93.23	83.03	74.84	84.3
XGBoost	84.6	91.94	83.06	75.75	84.3
CatBoost	**85**	**93.25**	**83.23**	**75.15**	**85**

The table above shows the accuracies, F1 score, recall, Precision and Roc values of the different models

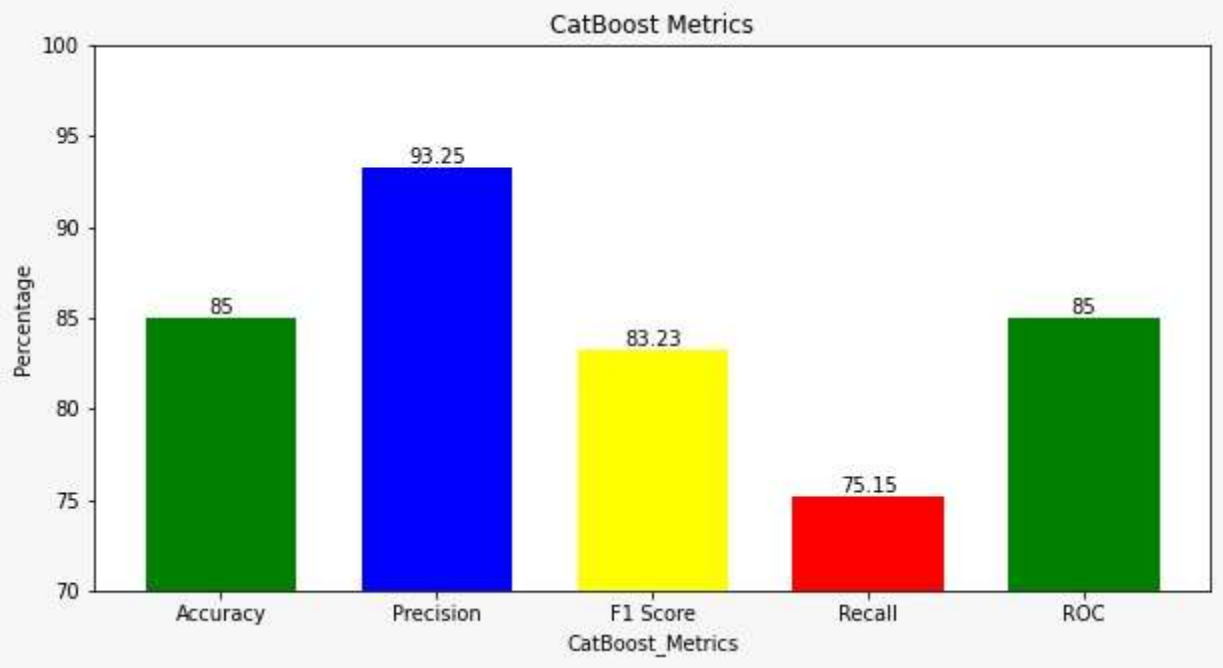
Models are trained with 80% of training data and 20% of data is utilized for testing. CatBoost has highest accuracy of 85% preceded by Random Forest (84.3%) and XGBoost (84.6%). Although, the accuracies are varied by minimal differences, we are considering other metrices for model evaluation such as F1 Score, Precision and ROC. XGboost has higher precision value than CatBoost and but has less ROC value than CatBoost. Random Forest has accuracy equal to ROC measures the performance of classification between classes and CatBoost outperforms in accuracy and ROC which has probability of 85% to classify phishing and legitimate URL's and this is considered as best model.



The above image shows the comparison of accuracies between different algorithms. CatBoost has highest accuracy preceded by XGBoost and Random forest



The image above shows the feature importance in CatBoost for classifying the data as phishing or legitimate.



The above image depicts the metrics Accuracy, F1 Score, Precision, Recall and ROC for CatBoost.

8. CONCLUSION

This study provides a comparison of performance different algorithms to classify phishing and legitimate URL's Phishing has become a severe threat in stealing personal data in various forms like spear phishing, whaling, smishing. To reduce the phishing attacks there were several strategies and solutions addressed by the professionals, research institutes. Of all the methods, machine learning algorithms are proved as best approach to identify malicious URL's. In our study, we implemented traditional algorithms like logistic regression, KNN, SVC, Decision trees, Randomforest and received lower rate of detection. Hyperparameter tuning approach is utilized in this project to obtain best optimal parameters to train the model.

We have considered real-time data, performed hyperparameter tuning, and compared the predictive accuracy of eight classifiers on the phishing dataset. The classifiers included decision trees, k-nearest neighbors, support vector machines, logistic regression, random forest, XGBoost, CatBoost.

Out of which, CatBoost performs exceptionally well, with an accuracy of 88.4%. Hence, the CatBoost is the best classifier for detecting whether or not a website is phishing. Also, the model performs decently well on unseen data.

9.REFERENCES

[1] Y. Sönmez, T. Tuncer, H. Gökal, and E. Avci, "Phishing web sites features classification based on extreme learning machine," 6th Int. Symp. Digit. Forensic Secur. ISDFS 2018 - Proceeding, vol. 2018-Janua, pp. 1–5, 2018, doi: 10.1109/ISDFS.2018.8355342.

[2] M. Zabihimayvan and D. Doran, “Fuzzy Rough Set Feature Selection to Enhance Phishing Attack

Detection,” IEEE Int. Conf. Fuzzy Syst., vol. 2019-June, 2019, doi: 10.1109/FUZZ-IEEE.2019.8858884.

[3] S. Parekh, D. Parikh, S. Kotak and S. Sankhe, "A New Method for Detection of Phishing Websites: URL Detection," 2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT), 2018.

[4] International Journal of Advanced Computer Technology (IJACT), “A Review of Various Techniques for Detection and Prevention for Phishing Attack”.

[5] A. T. Nguyen, B. L. To, H. K. Nguyen and M. H. Nguyen, "Detecting phishing web sites: A heuristic URL-based approach," 2013 International Conference on Advanced Technologies for Communications (ATC 2013), 2013, pp. 597-602, doi: 10.1109/ATC.2013.6698185.

[6] A. Desai, J. Jatakia, R. Naik and N. Raul, "Malicious web content detection using machine leaning," 2017 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), 2017, pp. 1432-1436, doi: 10.1109/RTEICT.2017.8256834.

[7] C. Seifert, I. Welch, and P. Komisarczuk, “Identication of malicious web pages with static heuristics,” in The Australasian Telecommunication Networks and Applications Conference, 2008

[8] N. Chou, R. Ledesma, Y. Teraguchi, and J. Mitchell, “Client-side defense against web-based identity theft,” in The 11th Annual Network and Distributed System Security Symposium, 2004.