INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X
Impact Factor: 6.078
(Volume 10, Issue 1 - V10I1-1252)
Available online at: https://www.ijariit.com

Chrome extension to detect malicious URLs using support vector machine algorithm

Dev Kumar
devkr644@gmail.com
Excel Engineering College, Komarapalayam, Tamil Nadu

E. Deepan Kumar
edeepankumar.eec@excelcolleges.com
Excel Engineering College, Komarapalayam, Tamil Nadu

Aarya D Roy
royaaryaprince@gmail.com
Excel Engineering College, Komarapalayam, Tamil Nadu

Aftab Alam
raushankumar@gmail. com
Excel Engineering College, Komarapalayam, Tamil Nadu

Harsh Vardhan
harshkvardhan4@gmail.com
Excel Engineering College, Komarapalayam, Tamil Nadu

ABSTRACT

There are a vast number of domains that leads the user to phishing websites for tricking the user to steal their sensitive information or inject malware into the system. In this paper, we will discuss about the support vector machine algorithm and how we have used the weights associated with each attribute of the trained SVM model in a chrome extension. This tool will identify the nature of the URL and avert the users from becoming a victim of phishing attack by notifying the user about the malicious URL on each page load via an alert box as safe or unsafe URL.

Keywords: *URL, Support Vector Machine, SVM, Chrome Extension, malicious URL, Machine Learning, Feature extraction, JavaScript.*

I. INTRODUCTION

The internet has become necessity for the people to do their work and to make life more joyful and convenient. There are over a billion websites on the internet, among which there are a significant amount of phishing domains that tricks the user to get their sensitive data. There is always a high chance of a user to enter a phishing website while surfing on the internet. Most of the phishing attacks are done on the social media platforms, as the users spend most of their time on the social media surfing through a large number of posts, where they encounter various advertisements and fun promising sites like, birthday wish, new year wish, etc. that could redirect the users to malicious domains.

We have discussed about the support vector machine algorithm and how we have used it in chrome extension tool for detecting malicious URL on each and every page load in the browser. The SVM is used to train the model with a dataset of 11055 instances of the URLs that are obtained from 4898 phishing websites and 6157 benign websites. Each instance of URL contains value of 30 attributes/features of the URL as 1 for malicious, -1 for safe or 0 for maybe and the target label for each URL instance as 1 (unsafe) or -1 (safe). The features of URL are categorized in three types generally, which are Content-Based, Host-Based, and Lexical features. We are extracting these features from the URL using JavaScript in our chrome extension tool and use the linear SVM

decision function to get the accurate prediction of the URL as safe or unsafe. This approach will make the tool light weighted and fast for malicious URL detection.

II. RELATED WORKS

Paper [1] has demonstration about the methods like, wrapper type that can be used to select features by modification of SVM and general feature selection that is independent of SVM. With the strategies of various feature selection, it discussed about the performance of combined SVM.

The URL features extraction in [2] were done with focusing mainly in lexical, host based and site popularity features. It discussed about the Support Vector Machine and Random Forest algorithm and how the extracted features of URL were used for the detection of malicious domains.

Paper [3] mainly focused towards the importance of feature extraction of URL and how useful the extracted features are, for the classification of malicious URLs. It discussed about 18 features of the URL to be extracted.
In paper [4], it is discussed that the spammers use twitter platform for spams and phishing attacks and how to filter the spams using sender-receiver relationship on the twitter. It uses the relation feature for the detection of spammers as it is difficult to manipulate relation feature. This paper covered about the detection of spams happening on the twitter.

In paper [5] linear and nonlinear space transformation methods are discussed for the detection of malicious URLs via feature engineering and machine learning model. It demonstrated and compared the performance of five machine learning models for precise detection of malicious URLs.

In [6] the various features where extracted from the URL and is used to train the machine learning models- Random Forest and Support Vector Machine for the detection of malicious URLs. The demonstration shows that the random forest algorithm works best with 100 trees and SVM uses all the features of URL for detecting phishing domains.

In survey [7] the performance of Neural Network and Machine Learning methods are discussed for the detection of malicious URLs, the neural network methods include Generative Adversarial Network, Neural Network Architect, Recurrent Neural Network and Convolutional Neural Network. The Support Vector Machine, Decision Trees, Random Forest, XGBoost, Gradient Boosting, AdaBoost and K Nearest Neighbor are the machine learning methods discussed in the paper.

In paper [8] 30 features of URL were discussed and how it contributes in accurate prediction of phishing domain by training the machine learning model with these features extracted from the URL. The demonstration of Support Vector Machine, Artificial Neural Networks classification and Extreme Machine Learning Algorithm were done and the comparison of these models showed the highest precision performance of Extreme Machine Learning Algorithm.

Paper [9] discussed about the structure of URL and training the logistic regression model using only the structural features of phishing URLs (without consideration of any page data) for the classification of phishing and safe domains.

In [10] author has discussed about the potential of twitter posts containing malicious URL and it mainly focused on redirecting chains of URL, shared resources and the correlations of multiple redirecting chains that shares redirection servers.

III. PROPOSED WORK

URL and its Features

URL stands for Uniform Resource Locator. it is an address of a resource on the internet. It is a string, that is used to locate and retrieve available resources on the web.
The main components of URL are Protocol, Host name, Primary domain, Top-level domain, Subdomain and Path.

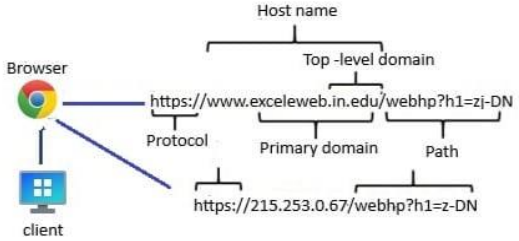

Figure 1. URL components

A malicious URL contains harmful content that can inject malware into the user's system or can redirect the user to a phishing website with the aim of getting sensitive information of the user by tricking them or taking control of the system.

The features of URL are categorized in three main types, that are content based features, lexical features and host based features. We will extract all these types of features for the prediction of the phishing domains with the help of decision function of SVM. The features that we will be extracting and are used for classification of malicious URL are: - Having IP address, URL length, Shortening service, Having '@' symbol, Redirecting URL, Prefix suffix, Having subdomain, SSL final state, Domain registration length, Favicon, Port, HTTPS token, Request URL, URL of anchor, Links in tags, SFH, Submitting to email, Abnormal URL, Redirect, On mouse over, Right click, Popup window, Iframe, Age of domain, DNS record, Web traffic, Page rank, Google index, Links pointing to page, Statistical report. All these features will be the attributes for the SVM model and will have possible value of 1 for malicious, -1 for safe or 0 for maybe.

SVM Model

SVM stands for Support Vector Machine. It is a supervised machine learning algorithm which can be used for both classification and regression problems. We have used SVM model as a classifier for our chrome extension tool.

It finds the best hyperplane to separate two classes of the data points. There could be infinite hyperplanes that separate two different classes but the hyperplane which has the maximum distance from both the classes is the best for the classification. The linear SVM model will have a straight line hyperplane in 2D plane separating the data points. The weights associated with all the attributes of our trained SVM model will play the key role as per our methodology to use it in the decision function of SVM for the prediction of malicious URL.

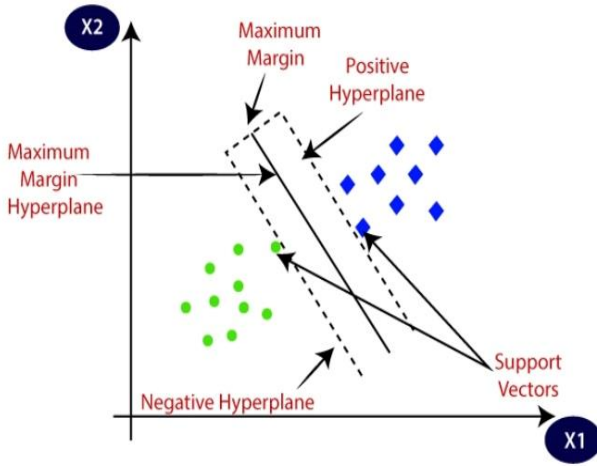

Figure 2. SVM linear hyperplane

The figure 2 shows the best hyperplane separating two classes of data points in 2D plane. For our model the positive/right side of the hyperplane will have 1 as malicious URL and negative/left side of the hyperplane will have -1 as safe URL.

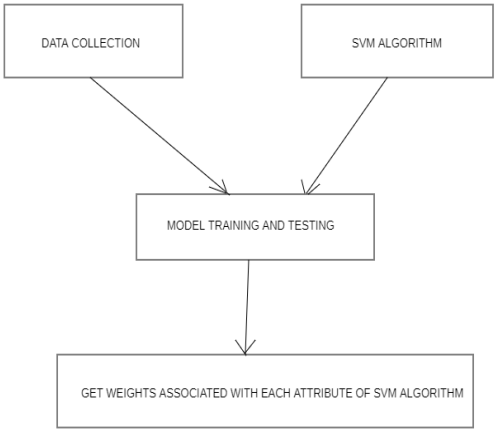

Figure 3. SVM model block diagram

Dataset

The dataset consists of 11055 unique URL instances. These URL instances are obtained from 4898 phishing websites and 6157 benign websites. Each row of the dataset has values of 30 features of an URL in 30 columns, the values are in 1 for malicious, -1 for safe or 0 for maybe. Each column of the dataset represents unique feature of URL and one column has results for supervised learning of the model.

Train and test the model

Using train_test_split from scikit-learn split the dataset in 80:20 and Fit the train test data into the model to train and test the Support Vector Machine model.

The SVM model will be trained with 80% of the dataset and 20% of the dataset will be passed to the model for the testing of model accuracy.

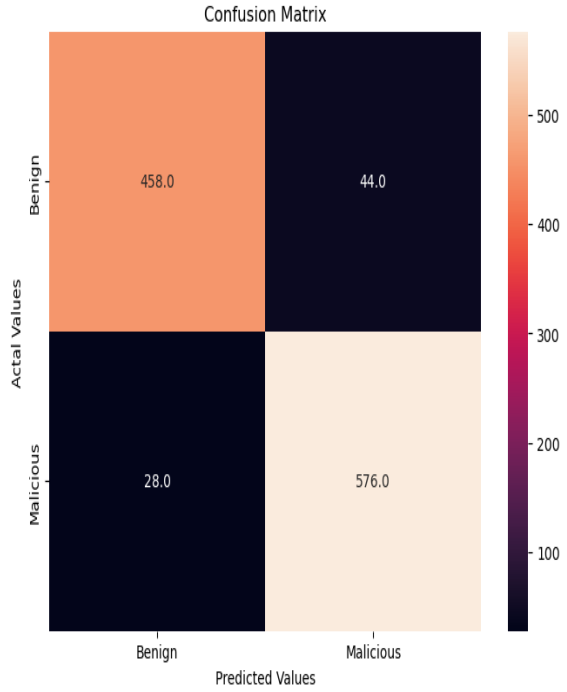

Figure 4. Confusion Matrix

In figure 4, the confusion matrix clearly shows that 1106 URL instances were tested, among which 458 benign URLs were predicted correctly as benign, 576 malicious URLs were predicted correctly as malicious, 44 benign URLs were predicted wrong as malicious and 28 malicious URLs were predicted wrong as benign.

Weights associated with each attribute of the trained SVM model

The trained linear SVM model has weight coefficient associated with each attribute/feature of the URL that is used to train the model.

We can get all the weight coefficient by the attribute coef_

```
Weights: [[ 0.38291098 -0.04234045 -0.47116119  0.199629    0.06440828  1.4845371
   0.28120139  1.11003478  0.02998025  0.0712307   0.25612659 -0.2441992
   0.15440562  1.76653726  0.34791423  0.3784021  -0.24151145 -0.08305214
  -0.73450425  0.03619849  0.03503546 -0.19415755 -0.06260429  0.04262898
   0.28930519  0.32006612  0.0682417   0.37768314  0.51666743  0.17396747]]
```

We will use these weights associated with each attribute/feature of the URL in the decision function for correct and fast predictions.

Chrome Extension Tool

Chrome extension is a web based application used to add functionality to the existing chrome browser. They are built using HTML, CSS, JavaScript and Manifest. The manifest.json file gives the basic information about the extension and is used to manage the scripts and getting permissions for the extension to work properly. The HTML and CSS are used to structure and style the popup page, frontend of the extension. The JavaScript is used to implement the logic and make backend of the extension for proper functionality and working.

Working of chrome extension tool

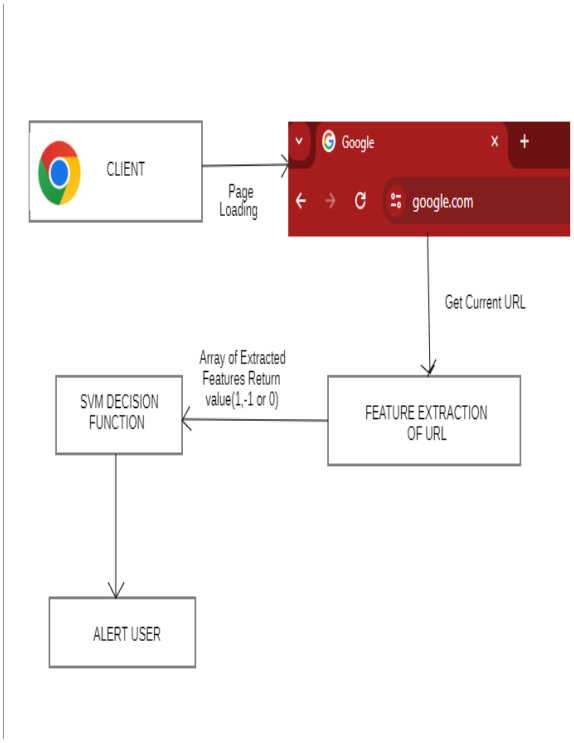

Figure 5. overview of chrome extension working

The chrome extension tool will catch the current URL of the loading webpage from the browser tab, then the features of URL will be extracted and the return values (1, -1 or 0) of each feature of URL will be stored in an array. This array will then be used as the input for the SVM decision function for the prediction of the nature of the URL as the result. After the successful prediction, the user will be informed with an alert box stating the nature of URL as safe or malicious. The tool will do it every time the user loads any webpage.

Feature Extraction from the URL

The feature extraction of URL will be done in the background by using JavaScript. The JavaScript file will be executed every time the user open any link on the browser. The features that will be extracted are: - Having IP address, URL length, Shortening service, Having '@' symbol, Redirecting URL, Prefix suffix, Having subdomain, SSL final state, Domain registration length, Favicon, Port, HTTPS token, Request URL, URL of anchor, Links in tags, SFH, Submitting to email, Abnormal URL, Redirect, On mouse over, Right click, Popup window, Iframe, Age of domain, DNS record, Web traffic, Page rank, Google index, Links pointing to page, Statistical report.

Every feature extraction function will return 1, -1, or 0. The returned value of extracted features will be stored in an array that will be passed to the decision function for the prediction of malicious URL.

Decision function and prediction

The SVM decision function is used to get the position of new data points from the hyperplane, based on that we can classify the data points as negative or positive.

The decision function for single data point uses the dot product of weight associated with the attribute of SVM and the value of new data point attribute that is, f(x) = x . w + b , where, x is the value of new data point attribute, w is the weight associated with that attribute and b is the bias term. The b = - c , where, c is the distance between the origin and the hyperplane. When f(x) >=0 , the data point lies on the positive side of hyperplane and when f(x) <0 , the data point lies on the negative side of hyperplane.

The decision function for array input of attribute values. It calculates the dot product for each attribute value and add all of them. The result value then compared with 0 for the prediction of the class it belongs to. The decision function is:

$$f(x) = \sum_{i=1}^{n}(w_i \cdot x_i) + b$$

Where, Wi are the weight associated with each attribute/ feature, Xi are the value of the attributes and b is the bias term.

When f(x) >= 0 then it is 1(indicates malicious URL), when f(x) < 0 then it is -1 (indicates safe URL)

IV. RESULTS AND DISCUSSION

The Support Vector Machine model is giving 93.49% accuracy in the prediction of malicious URLs.

	precision	recall	f1-score	support
Benign	0.94	0.91	0.93	502
Malicious	0.93	0.95	0.94	604
accuracy			0.93	1106
macro avg	0.94	0.93	0.93	1106
weighted avg	0.94	0.93	0.93	1106

accuracy = 93.49%

Figure 6. Performance of SVM model

In figure 6, we can see the f1-score, precision and recall for benign and malicious URL prediction. The accuracy mentioned in the figure 6 is percentage of correct predictions.
The formula for calculating the accuracy, precision and recall are:
Accuracy = (TP + TN /(TP+TN+FP+FN))*100

Precision = (TP/(TP+FP))*100
Recall = (TP/(TP+FN))*100
Where, TP (True Positive): - number of prediction that was positive and is actually positive.
TN (True Negative): - number of prediction that was negative and is actually negative.
FP (False Positive): - number of prediction that was positive and is actually negative.
FN (False Negative): - number of prediction that was negative and is actually positive.
The SVM model has high f1-score and is a good classification model. The formula of f1-score is:
f1-score = (2*(precision*recall))/(precision + recall)
The chrome extension tool is light weighted and is fast to predict the nature of the URL on all page load.

V. CONCLUSION

we have discussed about the Support Vector Machine algorithm and built an SVM model that can predict the nature of URL. We have demonstrated how to use the weights associated to the attributes of trained SVM model. These weights are used in the decision function as discussed in this paper and built a chrome extension tool for the detection of malicious URLs on each page load. The tool works with good accuracy of 93.49%. We have also seen the importance of feature extraction and how to use its return values in the decision function for the accurate prediction of the nature of the URL. This chrome extension tool is able to alert the user on each page load about the nature of the website they are visiting as safe or malicious URL. This helps the user to avoid the malicious URL and stay safe while surfing on the internet.

VI. REFERENCES

[1] Y.-W. Chen and C.-J. Lin. Combining SVMs with various feature selection strategies. In Feature Extraction, volume 207 of Studies in Fuzziness and Soft Computing, pages 315– 324. 2006.
[2] Ripon Patgiri, Hemanth Katari, Ronit Kumar and Dheeraj Sharma, (2020) “Empirical Study on Malicious URL Detection Using Machine Learning”, International Conference, ICDICT.
[3] Immadisetti Naga Venkata Durga Naveen, Manamohana K, Rohit Verma, (2019) “Detection of Malicious URLs using Machine Learning Techniques”, International Journal of Innovative Technology and Exploring Engineering (IJITEE).
[4] Jonghyuk Song, Sangho Lee, and Jong Kim. Spam filtering in twitter using sender-receiver relationship. In International workshop on recent advances in intrusion detection, pages 301–317. Springer, 2011.
[5] Tie Li, Gang Kou, Yi Peng (2020) “Improving Malicious URLs Detection via Feature Engineering: Linear and nonlinear Space Transformation Methods”, Information Systems (Elsevier).
[6] Cho Do Xuan,Hoa Dinh Nguyen, Tisenko Victor Nikolaevich,(2020) “Malicious URL Detection based on Machine Learning”, International Journal of Advanced Computer Science and Applications.
[7] Eint Sandi Aung, Hayato Yamana, (2020) “Malicious URL Detection: A Survey”, Department ofomputer Science and Communication Engineering, Graduate School of Fundamental Science and Engineering.
[8] Yasin Sonmez, Turker Tuncer, Huseyin Gokal, Engin Avci (2018) “Phishing Web Sites Features Classification Based on Extreme Learning Machine”, 6th International Symposium on Digital Forensic and Security (ISDFS).
[9] Sujata Garera, Niels Provos, Monica Chew, and Aviel D Rubin. A framework for detection and measurement of phishing attacks. In Proceedings of the 2007 ACM workshop on Recurring malcode, pages 1–8. ACM, 2007.
[10] Sangho Lee and Jong Kim. Warningbird: Detecting suspicious urls in twitter stream.

