A COMPARATIVE STUDY OF MACHINE LEARNING ALGORITHMS FOR MALICIOUS URL DETECTION

MR. ABHISHEK KUMAR
Research Scholar, CSE Department, Sandip University, Madhubani, BIHAR, INDIA
Email: abhi4613@gmail.com

DR. SHAMBHU KUMAR SINGH
Assistant Professor & Head, CSE Department, Sandip University, Madhubani, BIHAR, INDIA
Email: shambhu.singh@sandipuniversity.edu.in

ABSTRACT

There are a large number of domains that leads the user to phishing websites for tricking the user to steal their sensitive information or inject malware into their system. In this paper, we will discuss about three machine learning algorithms- Support Vector Machine, MLP Neural Network, Random Forest and compare their performance with each other to see which algorithm is giving the best output to help predict that the URL is safe or malicious.

Keywords

URL, Support Vector Machine, SVM, Random Forest, MLP Neural Network, malicious URL, Machine Learning.

1. 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 running on the internet for people to visit, among which a significant amount of websites are phishing domains that tricks user to take control of their system or get their sensitive data. There is always a higher chance of an URL to be malicious. People can enter to phishing websites while surfing on social media or websites on the internet. The significant amount of phishing attacks is executed on the social media platforms, as the users spend averagely more of their time on the social media encountering through a large number of posts, where they run into various advertisements and fun promising sites like, pranks, birthday wishes, new year cards, etc. that could redirect or trick the users to malicious domains.

We have discussed about three of the machine learning algorithms that are support vector machine, random forest, MLP neural network. These classification algorithms used to classify the URL into safe or malicious. The machine learning algorithms are used to train the model with a dataset of 11055 instances of the URLs that consist of 4898 phishing websites and 6157 safe websites. Each instance of URL in the dataset contains 30 attributes/features values 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 of three types generally, which could be stated as Content-Based, Host-Based, and Lexical based features. We will compare the algorithms with the classification report generated after the training and testing of the algorithms for the dataset. This comparison will show the best algorithm to be implemented for the classification of the nature of URLs.

2. 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. The Support Vector Machine and Random Forest algorithm and the extraction of features of URL were discussed and used for the detection of malicious domains.

Paper [3] was focused mainly on the importance of features of URL and how the extracted features could be useful for the classification of malicious domains. It has 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 author has extracted important features from the URL and used it to train the machine learning models such as, Random Forest and Support Vector Machine for the detection of malicious domains. 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.

3. PROPOSED WORK

3.1 URL and its Features

URL stands for Uniform Resource Locator. It can locate the web resources through the internet and get the data available for that particular website on the server. It is a string and is an address of the resource located on the server.

The URL contains some main components to locate the web resource that are Protocol, Host name, Primary domain, Top-level domain, Subdomain and Path.

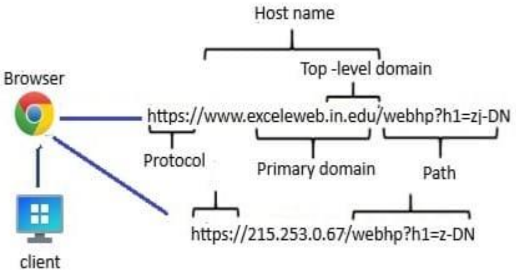

Host name
Browser
Top -level domain
https://www.exceleweb.in.edu/webhp?h1=zj-DN
Protocol
Primary domain
Path
https://215.253.0.67/webhp?h1=z-DN
client

Figure 1. URL components

A malicious URL have harmful contents that are used to inject malware into the user's system or redirect the user to a phishing website. This allows the attacker to get the sensitive information of user by tricking them to enter the information or taking control of their system.

The categorization of URL features is of three main type that are known as content based features, lexical features and host based features. We are using these features of URL to train the machine learning models and classify the URLs into two classes, one is safe and the other is malicious.

The features of the URL that is being considered for the training of our machine learning models for classification of malicious URL are: - SSL final state, Domain registration length, Favicon, Port, HTTPS token, DNS record, Web traffic, Page rank, Google index, Links pointing to page, Request URL, Having IP address, URL length, Shortening service, Having '@' symbol, Redirecting URL, Prefix suffix, Having subdomain, URL of anchor, Links in tags, SFH, Submitting to email, Abnormal URL, Redirect, On mouse over, Right click, Popup window, Iframe, Age of domain, 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.

3.2 Dataset

We are using the dataset of 11055 URL instances that contains 30 features of each URLs. All the URL instances in the dataset are unique and is consist 6157 safe and 4898 malicious URLs. The dataset has 30 columns of features of URLs and each rows has the feature value as 1 for malicious and -1 for safe in all the columns. The target values are in the different data file that will be passed together with the feature dataset to the machine learning model for the training and testing of the model.

3.3 SVM Model

SVM (Support Vector Machine) is a supervised machine learning algorithm that is used for both classification and regression problems. We are using the SVM model as a classification algorithm for the prediction of malicious URLs.

The SVM algorithm works by finding the hyperplane best suited for the classification of two classes of the data points. The best hyperplane is picked from the infinite hyperplanes separating two different classes by consideration of the maximum distance of the nearest data points of different classes, the distance must be equal from the hyperplane to both such data points. The linear SVM model separates the data points with a straight line to indicate the boundary of the class of the data point.

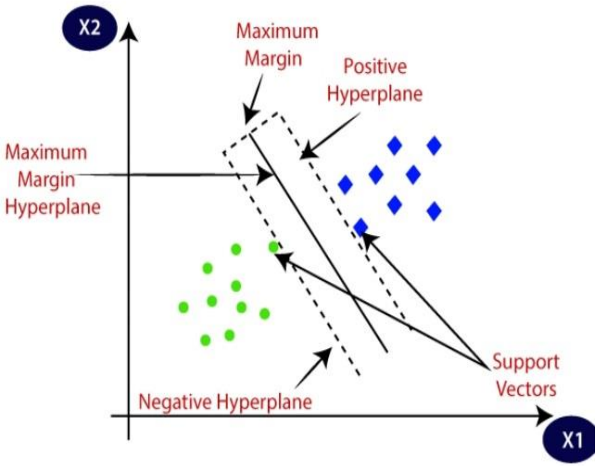


Figure 2. SVM linear hyperplane

In figure 2 we can observe the best hyperplane is a straight line that separates two classes of the data points in a 2D plane. Here the positive/right side of the hyperplane is considered as 1 which is malicious URL and negative/left side of the hyperplane is considered as -1 which is safe URL**.**

3.3.1 Training and testing SVM

The dataset will get split in the ratio of 80:20 Using train_test_split from scikit-learn. This data then will be passed to Fit it into the model for the training and testing of the support vector machine model.

The split ratio of dataset indicates that the SVM model will be trained with 80% of the dataset and the 20% of the dataset will be used for the testing of the model accuracy.

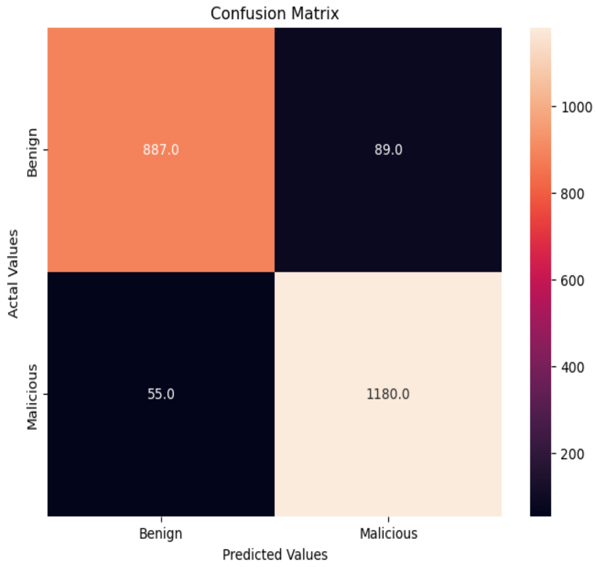


Figure 3. Confusion Matrix SVM

In figure 3, we can observe the result of the testing of model presented by the confusion matrix has passed 2211 URLs in total to the model for the prediction of malicious URL. In which there were 976 safe URLs among which 887 was predicted correctly as safe and 89 was predicted wrongly as malicious and among 1235 malicious URLs, 1180 were predicted correctly as malicious and 55 were predicted wrongly as safe URLs.

3.4 MLP Neural Network

MLP stands for Multilayer Perceptron and is an artificial neural network that is capable of handling complex classification problems.

MLP works with the weights and biases associated with the neurons of the network to classify the data point. It sets the random weights and biases to the neuron at the initial stage of training and changes it according to the results on each layers of MLP. It uses forward propagation for the prediction and at wrong predictions it uses backpropagation to change the weights and biases of the neurons for minimization of error and get correct predictions.

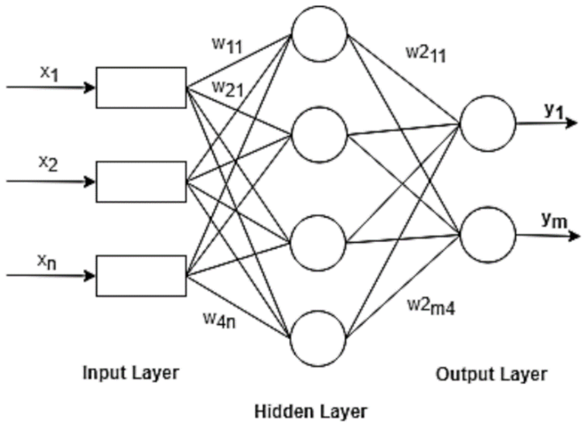


Figure 4. MLP classifier

3.4.1 Training and testing MLP Neural Network

The dataset will get split in the ratio of 80:20 Using train_test_split from scikit-learn. This data then will be passed to Fit it into the model for the training and testing of the MLP Neural Network model.

The split ratio of dataset indicates that the MLP model will be trained with 80% of the dataset and the 20% of the dataset will be used for the testing of the model accuracy.

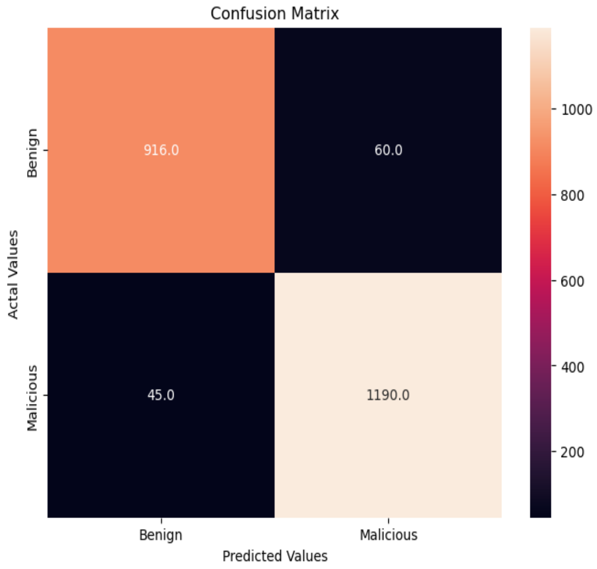


Figure 5. Confusion Matrix MLP

In figure 5, we can observe the result of the testing of model presented by the confusion matrix has passed 2211 URLs in total to the model for the prediction of malicious URL. In which there were 976 safe URLs among which 916 was predicted correctly as safe and 60 was predicted wrongly as malicious and among 1235 malicious URLs, 1190 were predicted correctly as malicious and 45 were predicted wrongly as safe URLs.

3.5 Random Forest

Random Forest is a supervised machine learning algorithm that can be used for both regression and classification problems. It can maintain a high accuracy of prediction with minimized overfitting. It uses the random subsets of dataset to make different decision trees for the training of the model.

It works by creating a set of decision trees with the help of subsets of the dataset which are selected randomly. The final prediction is obtained by taking votes from the different decision trees of the model. The diverse subsets of the dataset are created by a technique known as bagging or bootstrap aggregating. This helps the random forest algorithm to train and predict the nature of URL accurately.

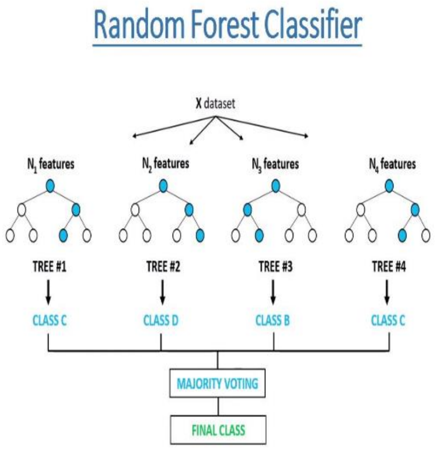


Figure 6. Random Forest classifier

3.5.1 Training and testing Random Forest

The dataset will get split in the ratio of 80:20 Using train_test_split from scikit-learn. This data then will be passed to Fit it into the model for the training and testing of the Random Forest model.

The split ratio of dataset indicates that the Random Forest model will be trained with 80% of the dataset and the 20% of the dataset will be used for the testing of the model accuracy.

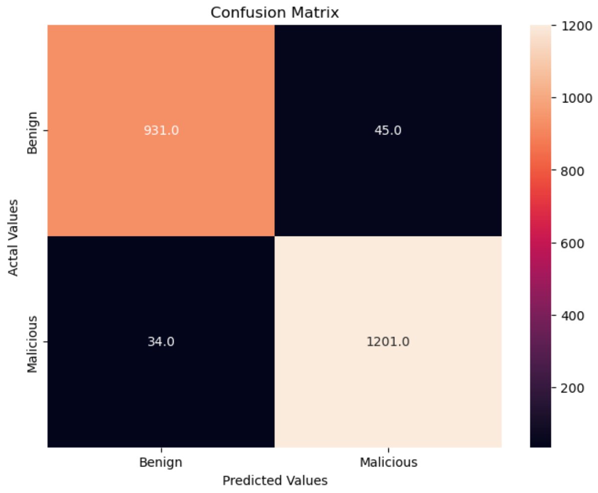
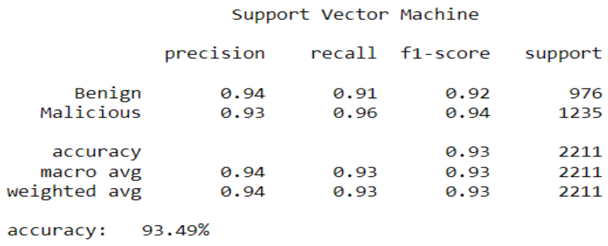


Figure 7. Confusion Matrix Random Forest

In figure 7, we can observe the result of the testing of model presented by the confusion matrix has passed 2211 URLs in total to the model for the prediction of malicious URL. In which there were 976 safe URLs among which 931 was predicted correctly as safe and 45 was predicted wrongly as malicious and among 1235 malicious URLs, 1201 were predicted correctly as malicious and 34 were predicted wrongly as safe URLs.

4. RESULTS AND DISCUSSION

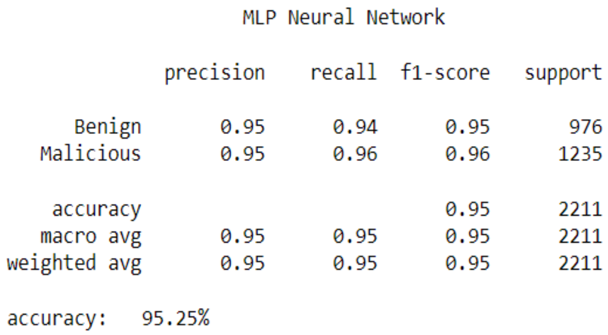
Now we will observe the classification report of each machine learning model and compare the accuracy of the models. In the classification report we can see the f1-score, precision and recall for benign and malicious URL prediction. The accuracy mentioned in the classification report is percentage of correct predictions.



```
                    Support Vector Machine

              precision    recall  f1-score   support

      Benign       0.94      0.91      0.92       976
   Malicious       0.93      0.96      0.94      1235

    accuracy                           0.93      2211
   macro avg       0.94      0.93      0.93      2211
weighted avg       0.94      0.93      0.93      2211

accuracy:   93.49%
```

Figure 8. SVM model classification report

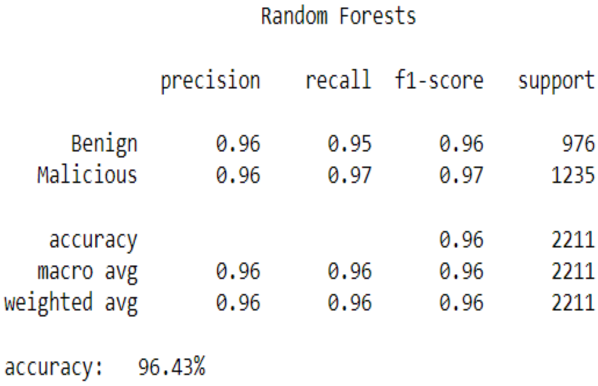
The Support Vector Machine model is giving 93.49% accuracy in the prediction of malicious URLs.



```
                     MLP Neural Network

              precision    recall  f1-score   support

      Benign       0.95      0.94      0.95       976
   Malicious       0.95      0.96      0.96      1235

    accuracy                           0.95      2211
   macro avg       0.95      0.95      0.95      2211
weighted avg       0.95      0.95      0.95      2211

accuracy:   95.25%
```

Figure 8. MLP model classification report

The MLP neural network model is giving 95.25% accuracy in the prediction of malicious URLs.



```
                         Random Forests

               precision    recall  f1-score   support

      Benign       0.96      0.95      0.96       976
   Malicious       0.96      0.97      0.97      1235

    accuracy                           0.96      2211
   macro avg       0.96      0.96      0.96      2211
weighted avg       0.96      0.96      0.96      2211

accuracy:   96.43%
```

Figure 8. Random Forest model classification report

The Random Forest model is giving 96.43% accuracy in the prediction of malicious URLs.

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 predictions that was positive and is actually positive.

TN (True Negative): - number of predictions that was negative and is actually negative.

FP (False Positive): - number of predictions that was positive and is actually negative.

FN (False Negative): - number of predictions that was negative and is actually positive.

The formula of f1-score is:

f1-score = (2*(precision*recall))/(precision + recall)

After the careful observation of all the classification reports obtained from the machine learning models. The SVM model has the least accuracy of 93.49% and MLP model sits in the middle with 95.25% accuracy and the Random Forest model has the highest accuracy of 96.43% in the prediction of malicious URLs.

5. CONCLUSION

we have discussed about the Support Vector Machine algorithm, MLP Neural Network, and Random Forest algorithm in this paper. All these machine learning models were trained using a dataset of 11055 unique URL instances with 80:20 data split. We have observed the testing result with the help of confusion matrix and compared the accuracy of the models by analyzing the classification report of each machine learning models discussed in this paper. With the careful analysis and observation, we can say that the Random Forest is giving the highest accuracy of prediction among the three machine learning models discussed in this paper. Thus the Random Forest model is more accurate than SVM model and MLP model in prediction of malicious URLs.

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.