IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Phishing Website Detection Using Machine Learning

Jiya Sharma
Graphics & Multimedia
Moradabad Institute of Technology
Moradabad, India

Richa Saxena
Deptt. Of CS&E
Moradabad Institute of Technology
Moradabad, India

Abstract *— Phishing attacks send bogus messages that are likely to come from legal sources. This is usually done by email. The main aim of phishing is to sneak sensitive data such as credit card numbers, and login credentials and install malware on the victim's computer and bank account details. This paper describes a machine-learning technique to detect phishing URLs by unsheathing & investigating profuse characteristics of legitimate and phishing URLs. Machine learning techniques used for phishing websites are decision trees, random forests, support vector machines, and XGBoost.*

Keywords—Phishing, Machine Learning

I. INTRODUCTION

Phishing simply means stealing someone's confidential information from an organization, individual, or group of people, this is mostly banking information. This can be done through impersonation, malicious activity, etc [1]. These attacks are mostly known as cyberattacks, hence small to large-scale crimes are perpetrated by thieves. Phishing is the most common scam that attempts to trick you into giving away usernames, passwords, or other sensitive information by impersonating someone you know and trust.

II. LITERATURE SURVEY

In this work, the authors Rishikesh Mahajan and Irfan Siddavatam [2] choose 3 algorithms: the decision tree algorithm, the random forest algorithm, and the support vector machine algorithm. 17,058 benign and 19,653 phishing contained by their dataset URLs gathered from the online site named Alexa and Phish-tank respectively, each containing sixteen functions. The dataset changed into split into education and testing sets with ratios of 50:50, 70:30, and 90:10, respectively. Accuracy rating, false-poor charges, and fake-tremendous fees have been taken into consideration as performance metrics. With the random forest algorithm, they achieved 97.14% accuracy and the lowest false negative rate. According to the conclusion of this paper to get better precision, use more data as possible for training.

Take a look at via Jitendra Kumar et al. [3] With a selection of pre-educated classifiers which include logistic regression, naive Bayes classifiers, random forests, and selection trees, based totally on features extracted from the lexical structure of the URL. They created the URL dataset in a way that solves the problems of data imbalance, biased education, distribution, and overfitting. The dataset contained an identical number of flagged phishing and legitimate URLs and become further cut up 7:3 for training and trying-out purposes. The AUC (region below the ROC curve) of all classifiers turned nearly identical, however, the Naive Bayes classifier became located to be advanced. Naive Bayes carried out the best accuracy of ninety-eight% with an accuracy of one, a don't forget of 0.95, and an F1 rating of 0. Ninety-seven,

Mehmet Korkmaz et al [4] gadget getting to know-primarily based phishing detection machine the use of eight specific algorithms on three one-of-a-kind datasets. The algorithms used had been Logistic Regression (LR), k Nearest acquaintances (KNN), guide Vector Machines (SVM), selection timber (DT), Naive Bayes (NB), XG-improve, Random-forest (RF), and synthetic Neural Networks. (Ann). it has been determined that fashions using LR, SVM, and NB are much less accurate. In terms of training time, NB, DT, LR, and ANN algorithms gave better consequences. They concluded that using RF or ANN algorithms reduces education time and accordingly can be used with better accuracy. To detect the phishing attack Mohammad Nazmul Alam et al. [5] create a system by using random forests and decision trees.

III. METHODOLOGY

The module description of the analysis presents below in figure 1.

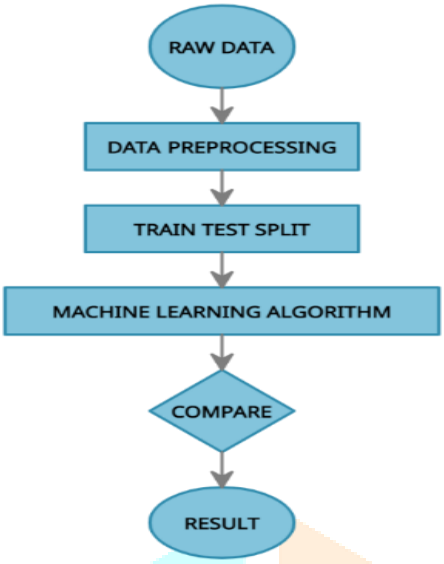

Figure 1: Methodology

A. Dataset

The URLs of phishing websites were obtained from the link:(https://www.phishtank.com/developer_info.php). The URLs of appropriate websites were earned from the link:(https://www.unb.ca/cic/datasets/url-2016.html) website which has a total of one lakh dataset, 50,000 is phishing websites and 50,000 is legitimate websites.

B. Data Preprocessing

Data preprocessing includes cleansing, example selection, function extraction, normalization, transformation, and many others. Absolute training data set is the outcome of Data processing. Preprocessing of data can affect how the results of final processing are interpreted. Data cleaning can be the steps to fill in missing data, remove noise and detect or remove outliers. [6] Data transformation involves performing collection and normalization to measure specific data. Data reduction allows you to get an overview of your dataset.

C. Feature Extraction

We have implemented many features in the python language. Features extracted for phishing URL detection are listed below: -

1. Using the IP address in the URL
2. Extend URL to Hide the dubious Part
3. Use of "Small URL" URL shortening services
4. URL's having "@" Symbol
5. Redirecting using "//"
6. Adding Prefixes or Suffixes
7. Parted by (-) to the Domain
8. Hostname length
9. Sensitive terms present in the URL.
10. Website Traffic
11. I-Frame Redirection
12. Page Rank
13. Website Rank

D. Train-Test-Split

The dataset is divided into two subsets, the test dataset, and the training dataset so that we can arm the algorithm with the training dataset and use it to detect phishing websites on the test dataset. 30% of the data is checked for the test set so that the training model can effectively train and learn from the data.

IV. ALGORITHM USED

A. Decision Tree Classifier

A decision tree algorithm is trouble-free and can easily be used. The decision tree goes ahead by picking up the finest sliver from the attribute available for classification and is termed the root of the tree. Decision trees are similar to flowcharts, where each inner node represents a "test" for an attribute, each node reflects the conclusion of the test, and each leaf node represents the result of the test. Structure. class label (determined by the sum of all attributes).[7] Each inner node of the tree corresponds to an aspect, and each leaf node of the tree corresponds to a class label used to predict a target value or class. The Gini index and an information gain approach are used to determine these nodes.

The diagram in fig.2 represents a decision tree classifier.

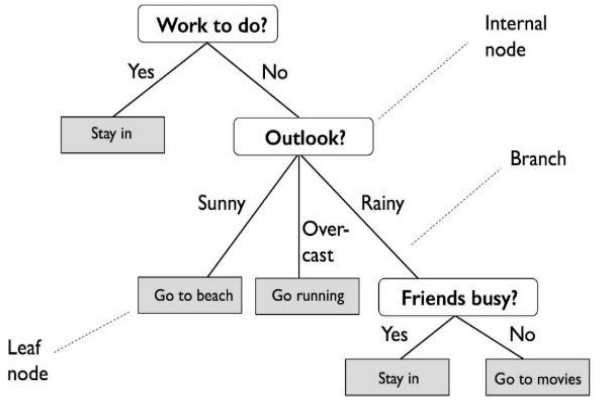

Figure 2: Decision Tree Classifier

B. Random Forest

Random Forest Algorithm is based on the concept of a D-tree algorithm and is known as the strengthiest classification algorithm.

A bootstrapping method is used to create the tree. To generate a single tree, the bootstrap approach randomly selects and replaces features and samples from the dataset with randomly selected ones.[8] Random forest algorithms, like decision tree techniques, select the best based on the feature splitters chosen for classification. The Gini index and information are also used in the random forest algorithm. Acquire access to ways for locating the greatest splitter. This procedure should go ahead until the random forest produces n trees. The random forest algorithm is capable to create a forest with n number of decision trees. Many trees deal with the issue of detection accuracy on a high level.

The diagram in fig.3 represents a Random-forest.

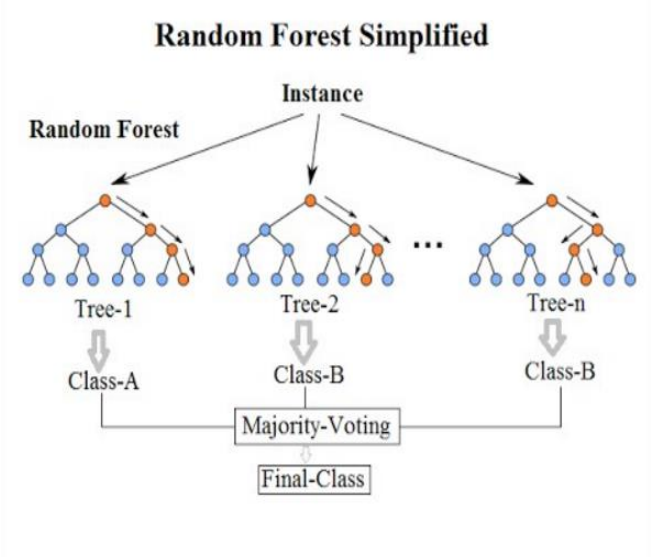

Figure 3: Random Forest

C. Support Vector Machine

Support Vector Machine is one of the highly effective classification algorithms which divides the input elements by a straight line, called Maximum Margin Hyperplane. Hyperplane divides the elements into two classes. The nearest points to the hyperplane, called support vectors, are found by the support vector engine, and lines connecting them are created. The support vector engine then creates a dividing line perpendicular to the connecting line that bisects. The margins should widely possible to classify the data accurately. In the real world, it is nearly impossible to distinguish between complex and nonlinear data. To counter this problem, support vector machines use a kernel trick that transforms a low-dimensional space into a high-dimensional space.

Figure 5 represents the Support Vector Machine (SVM)

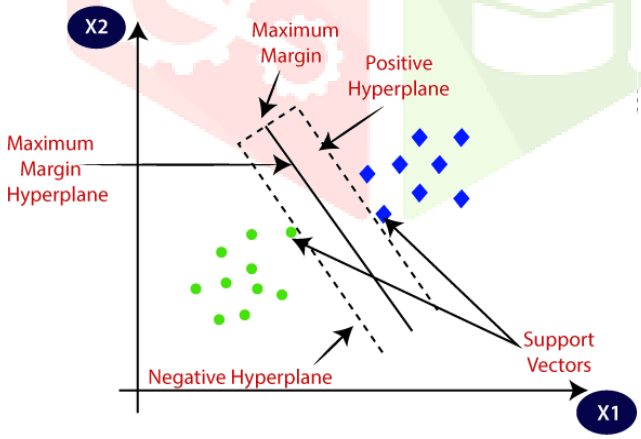

Figure 5: Support Vector Machine (SVM)

D. XGBoost

XGBoost is a gradient-boosting decision tree implementation. In XGBoost, weights play a vital role. Every independent variable is assigned some weight and entered into a decision tree that predicts outcomes [10]. The variables which are predicted incorrectly by the tree are weighted up and are passed to his 2nd decision tree. These individual classifiers/predictors are then combined to produce authoritative & greater accurate models. XGBoost can solve complications like regression, classification, ranking, and user-defined prediction complications.

V. RESULTS

The division of the dataset into training and testing has been done in a manner that the accuracy of the models becomes efficient. The ratio between them is 80:20 respectively. The performance of the classifiers can be evaluated by training each classifier using a training set and testing by using a testing set.

The following detection accuracies have been found using distinct algorithms and also the algorithms perform better on high training sets:

a) XGBoost algorithm gives a detection accuracy of 85.91% which is higher than other algorithms.
b) Decision tree algorithm gives a detection accuracy of 81.7%.
c) Support Vector Machine algorithm gives a detection accuracy of 81.53%.
d) Random Forest algorithm gives a detection accuracy of 83.2%.

Fig.4 represents the Model Comparison.

	ML Model	Train Accuracy	Test Accuracy
0	Decision Tree	0.810	0.817
1	Random Forest	0.819	0.832
2	SVM	0.798	0.815
3	XGBoost	0.868	0.859

Figure 4: Model Comparison

VI. CONCLUSION

This paper attempts to utilize machine learning techniques to upgrade the observation mechanisms for phishing websites. With the help of a random forest, we were able to attain 97.14% detection perfection with a low false positive rate. Furthermore, the consequences show that the classifier performs better while using more data than the training data. In the future, to detect phishing sites, we plan to adopt hybrid technology by combining the blacklist method and the random forest algorithm.

REFERENCES

[1] Jansson, K.; von Solms, R. (2011-11-09). "Phishing for phishing awareness". Behaviour & Information Technology. 32 (6): 584–593. doi:10.1080/0144929X.2011.632650. ISSN 0144-929XS2CID 5472217

[2] https://www.researchgate.net/publication/328541785_Phishing_Website_Detection_using_Machine_Learning_Algorithms

[3] Jitendra Kumar and A. Santhanavijayan, "Phishing Website Classification and Detection Using Machine Learning ", International Conference on Computer Communication and Informatics, 2020.

[4] Mehmet Korkmaz and Ozgur Koray Sahingoz, "Detection of Phishing Websites by Using Machine LearningBased URL Analysis", IEEE 2020.

[5] Mahmoud Khonji, Youssef Iraqi, "Phishing Detection: A Literature Survey," IEEE, and Andrew Jones, 2013.

[6] https://www.researchgate.net/publication/351929313_Phishing_website_detection_using_machine_learning_and_deep_learning_techniques

[7] Chen J, Wu C. Improvement and application of decision tree C4.5 algorithm [J]. Software Guide. 2018(10)

[8] Shan L. Fault analysis of distribution transformer and realization of random forest diagnosis[J]. Technology Wind,2016(24):120.

[9] Robert Johnson analysis of distribution Support_vector_machine 2007

[10] R. Bekkerman, M. Bilenko, and J. Langford. Scaling Up Machine Learning: Parallel and Distributed Approaches. Cambridge University Press, New York, NY, USA, 2011.