Detection of Phishing Websites Using Ensemble Machine Learning Approach

Dharani M.[1], *Soumya* Badkul[1], *Kimaya* Gharat[1], *Amarsinh* Vidhate[1], and *Dhanashri* Bhosale[1]

[1]Computer Engineering Dept., Ramrao Adik Institute of Technology, Navi Mumbai ,India

Abstract. In this paper, we propose the use of Ensemble Machine Learning Methods such as Random Forest Algorithm and Extreme Gradient Boosting (XGBOOST) Algorithm for efficient and accurate phishing website detection based on its Uniform Resource Locator. Phishing is one of the most widely executed cybercrimes in the modern digital sphere where an attacker imitates an existing - and often trusted - person or entity in an attempt to capture a victim's login credentials, account information, and other sensitive data. Phishing websites are visually and semantically similar to real ones. The rise in online trading activities has resulted in a rise in the number of phishing scams. Cybersecurity jobs are the most difficult to fill, and the development of an automated system for phishing website detection is the need of the hour. Machine Learning is one of the most feasible methods to approach this situation, as it is capable of handling the dynamic nature of phishing techniques, in addition to providing an accurate method of classification.

1 Introduction

1.1 Overview

The internet has seen rapid growth in the last decade. With the internet connecting billions of people globally, it is critically important to acknowledge the fact that the safety and privacy of internet users are not optimal. The rate of cyber-crime is on the increase and leads to great financial losses each year.

Phishing attacks account for more than 80% of reported security incidents. According to Verizon's 2021 Data Breach Investigations Report (DBIR) [1], 3,841 phishing incidents were reported till May 2021, wherein data disclosure was confirmed for at least 50% of the cases. Number of breaches involving phishing shows an 11% increase in 2021 as compared to the previous year. 95% of these attacks were financially motivated causing a loss of thousands of dollars per minute.

Many programs and seminars are conducted across the world to educate users and spread awareness about phishing scams. A Uniform Resource Locator (URL) is a unique identifier that locates a resource on the internet. It carries various parts such as protocol, domain name, port, path, query, etc. There are certain aspects in the URL of a phishing website that may be used to distinguish it from legitimate ones. However, it is not always possible to identify a website's legitimacy just by looking at the URL from a human perspective. Fortunately, Machine Learning algorithms prove to be an accurate and efficient method in recognizing these features, and in predicting whether a given website is a phishing website or a safe one.

1.2 Background

1.2.1 Lexical Features of URL

Lexical Features are the textual properties of a word. The most significant lexical features of a URL are length, hostname, and path. These features can be used to judge the legitimacy of a website without actually visiting it.

1.2.2 Ensemble Methods

Ensemble machine learning methods train multiple classification models using the same learning algorithm which generally consists of a pool of Decision Trees. Since each constituent learning algorithm will have its individual output, the final result is obtained with the help of a combining mechanism such as:

1. Voting (majority wins)
2. Weighted voting (some classifier has more authority than the others)
3. Averaging the results

2 Related Work

A very extensive range of studies and research has been carried out on the subject of phishing detection. General classification approaches include page-based, content-based, domain-based, and URL-based methodologies. In addition to user education, email regulation and classification could aid in the reduction of phishing incidents. "Anti Phishing Simulator" software was developed at Firat University, Turkey to facilitate the detection of phishing and spam emails by examining the email content [2].

The use of Convolutional Neural Networks for classification has also yielded positive outcomes, according to this study [3].

Most of the work done on the subject made use of traditional machine learning algorithms like Naïve Bayes, Support Vector Machine and Decision Trees [4].

The Remove Replace Feature Selection Technique (RRFST) aims at reducing the number of features by excluding insignificant features. It randomly selects a feature to test its effect and replaces it with another if it has a negative effect [5].

This study [6] made use of the XGBOOST algorithm for phishing detection which evidently outperformed other popular machine learning methods.

3 Limitations of Existing System

In most cases, a large number of features have been used in training the classification model, leading to an increase in the search space dimension. According to Hughe's effect, also known as the Curse of Dimensionality, the efficiency of a classifier demonstrates a steady increase only up to a certain threshold dimensionality, after which a decline is observed. To overcome this problem, a feature selection technique must be used.

The use of traditional machine learning algorithms may produce fair results but are highly susceptible to underfitting and overfitting, and may not always produce optimum results. Ensemble machine learning algorithms may be used to address this problem.

Even though email is the most common method of approaching potential phishing victims, it is not the only one. There are other forms of communication, such as social media, advertisements, text messages, telephonic conversations, etc., which can also lead an unsuspecting internet user to a phishing website. Thus, the scope of detection must not be limited to email-based phishing attacks.

Phishing is not only financially motivated but could also have identity theft or cloning as its incentive. Hence, in addition to securing banking and e-commerce websites, it is equally important to focus on generic sites as well, to ensure the overall safety and security of an internet user.

4 Proposed Methodology

4.1 System Architecture

The input is collected from the user in the form of a Uniform Resource Locator string which is treated as raw data. It is then processed to extract the lexical features and other characteristics. The processed input data is further passed to the trained model which predicts whether the URL entered is legitimate or fraudulent.

The training dataset for the classifier is collected from the standard machine learning repository made available for use by the University of California, USA, which contains over 2000 records with 30 attributes. The information present in this dataset is collected from a global community called 'Phishtank' which collectively reports and verifies the presence of suspicious activities on a potential phishing website. After data pre-processing and feature selection, the dataset is used to train the classification models which are based on bagging and boosting ensemble machine learning algorithms.

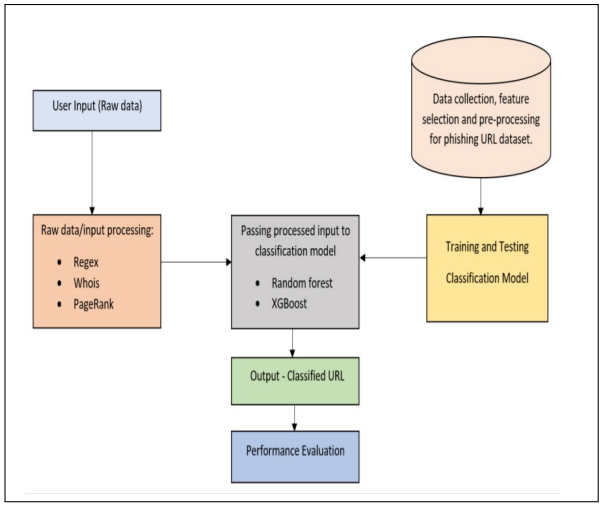

Figure 1. System Design

4.2 Raw input processing

The following pre-processing techniques and tools were used to transform the raw URL input text into predictable and analyzable features:

- Regex: A regular expression (Regex) is a sequence of characters that define a search pattern. It is used to determine the following features of the input URL: '@' symbol, '//' redirect symbol, Prefix, Suffix, Protocol, Subdomain etc.
- whois : whois is a protocol which can be used to calculate the domain age by retrieving the creation date.
- PageRank: The PageRank algorithm indicates the importance of a website on the internet based on its popularity determined by the number of visitors.
- Prefix or Suffix : Prefixes or suffixes are generally added to domain names, usually separated by a hyphen, giving the illusion of legitimacy.
- Long URL : Phishers might use long URL to hide malicious file locations within multiple sub-directories.
- Domain length : Maximum length of domain can be 63 characters. Phishers might use long domain names to confuse the users.

4.3 Feature selection

Feature selection is a pre-processing step in which the insignificant features present in the data are eliminated.

Chi-Square test, a filter method, is used to evaluate the relationship of a feature with the target variable. Chi-Square value is given by (eqn. 1):

$$\chi_c^2 = \sum_{i=1}^{n} \frac{(O_i - E_i)^2}{E_i} \quad (1)$$

where c is the degree of freedom, O is the observed value, and E is the expected value.

Clearly, Chi-square value is small if the observed value is close to the expected value. Therefore, features are more independent if Chi-Square value is small.

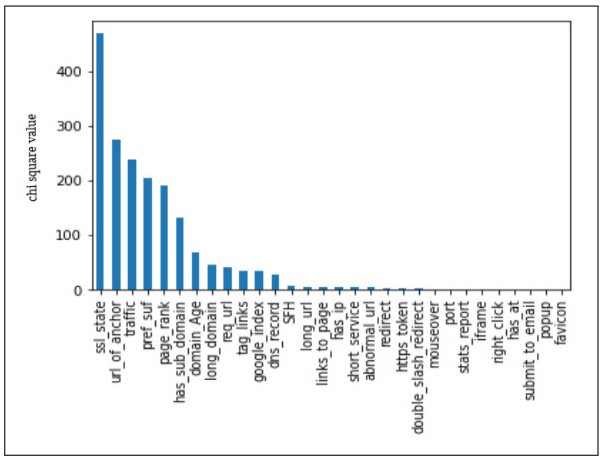

Figure 2. Chi-Square value against feature set

Fig.2. shows the plot of features against the chi-square value. The top 15 features having high Chi-square value were selected for training the classification model.

4.4 Classification model

The next step is to train a machine learning model for classifying a website as a legitimate or a phishing website. It is crucial to choose the right classification method which adheres to the application's needs.

There are 3 main types of ensemble methods: Bagging, Boosting, and Stacking. Bagging stands for Bootstrapping and Aggregating. It is designed to avoid overfitting by the random selection of data. Boosting is an ensemble approach that aims to convert weak learners into strong learners by reducing bias. Stacking uses a meta-classifier to combine the predictions of heterogeneous weak learners. This project proposes the use of two classification algorithms, Random Forest Classifier and XGBoost.

4.4.1 Random Forest Algorithm

Random forest is a popular bagging ensemble machine learning algorithm, which aims to reduce the variance by training the model on different parts of the same training set using multiple deep decision trees, whose individual results are then averaged to obtain the final classification. To decide branching of nodes on a decision tree during classification, the Gini Index is used, which is calculated as given below (eqn. 2):

$$Gini = 1 - \sum_{i=1}^{c} (P_i)^2 \quad (2)$$

where, Pi is the relative frequency of the observed class, and c is the number of classes.

Gini Index is calculated by subtracting the sum of the squared probabilities of each class from 1. Thus, the class with the least value of Gini Index is given the highest preference, as its probability of being wrong is less. The use of Gini Index is advantageous as it favours large partitions and has a simple implementation.

Random Forest Algorithm (Algo. 1):

Step 1 - Random samples are selected from the dataset.

Step 2 - Decision trees for individual samples are constructed. The algorithm will then obtain the prediction result from the formed decision trees.

Step 3 - Every predicted result will go through voting.

Step 4 - Result with the majority of votes will become the final prediction result.

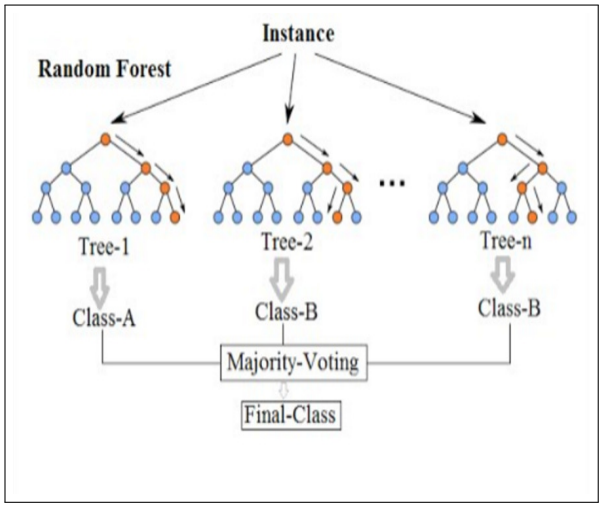

Figure 3. Random Forest [7]

4.4.2 XGBoost Algorithm:

XGBoost (Extreme Gradient Boosting) is an optimized distributed gradient boosting algorithm designed to be highly efficient, flexible, and portable. It uses parallelization, tree pruning, and weighted classifiers to produce superior results.

XGBoost Algorithm (Algo. 2):

Step 1 - Regression trees are the weak learners and each regression tree maps an input point to one of its leaf nodes.

Step 2 - An initial model F1, that will be associated with the residual (F1-y), is defined to predict the target variable y.

Step 3 - A new model T2 is trained with independent variables and residual errors from the previous step as the data to get the predictions.

Step 4 - F2, a combination of F1 and T2, is the boosted version of initial model F1. It is calculated with the additive predictions and residual errors along with some learning rate from output predictions obtained previously from the model.

Step 5 - Steps 3 and 4 are iterated for M number of times

until the required number of models are built.
Step 6 - The final prediction from boosting is the additive sum of all the previous predictions made by the models.

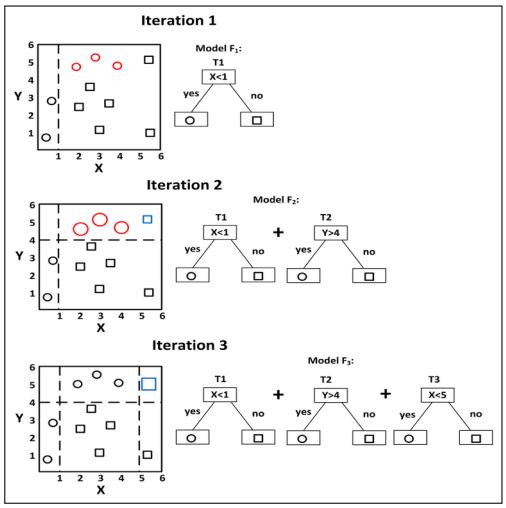

Figure 4. Gradient Boosting Visualization [8]

Bagging methodologies like Random Forest are known to reduce the variance and thus solve the overfitting problem, but may lead to high bias or underfit. Boosting algorithms rectify each learner's errors, thereby solving the problem of underfitting, but may lead to overfitting. Bagging and boosting when used in conjunction, will lead to a satisfactory bias-variance trade-off. XGBoost algorithm may be used to improve the performance of the Random Forest model, when applied for Phishing Detection as in this case, the predictive accuracy is far more important than model interpretability.

4.4.3 Performance measures

The predictive efficiency is indicated by evaluation metrics as discussed below. Accuracy is defined as the number of correct predictions over the total number of predictions made by the model (eqn. 3). Precision is the number of true predictions made by the model (eqn. 4). Recall is the number of positives returned by the model (eqn. 5). F-score is the mean of precision and recall (eqn. 6).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{3}$$

$$Precision = \frac{TP}{TP + FP} \tag{4}$$

$$Recall = \frac{TP}{TP + FN} \tag{5}$$

$$F - score = \frac{2 * Recall * Precision}{Recall + Precision} \tag{6}$$

Table 1. Confusion Matrix

	Positive	Negative
Positive	TP	FP
Negative	FN	TN

These metrics are used to create the Confusion Matrix illustrated in Table 1, which describes the model's performance on test data for which the true values are known.

5 Result and Discussion

Using the Random Forest Classifier, a prediction accuracy of 91% was obtained. Using XGBoost, a prediction accuracy of 93% was obtained.

The values of performance measures for both the algorithms are shown in Table 2.

Table 2. Summary of test results

Algorithm	Accuracy	Precision	Recall	F-score
Random Forest	0.911	0.883	0.914	0.898
XGBoost	0.937	0.938	0.949	0.928

The Confusion matrix values of Random Forest and XGBoost are shown in Table 3.

Table 3. Confusion Matrices values of Random Forest and XGBoost

n=461	Random Forest		XGBoost	
	Positive	Negative	Positive	Negative
Positive	239	24	224	19
Negative	17	181	10	188

For obtaining the final result, the website's legitimacy is predicted individually by both the models and then an average is drawn. This method produces three possible values - Zero (0) indicating that the website is safe, Half (0.5) indicating that the website has a 50% risk of being malicious, One(1) indicating that the website is phishy, and must be avoided.

6 Conclusion

As the number of internet users is on a steady rise, the number of unregulated websites today is more than ever. Phishing evolves with time, as illegitimate websites do not exist permanently, and undergo changes frequently.

Using the URL of a website for phishing detection with the aid of Ensemble Machine Learning Algorithms like Random Forest and XGBoost with a curated feature set is expected to produce highly accurate predictions with a satisfactory bias-variance trade-off in a timely and secure manner.

References

[1] Verizon, Data Breach Investigations Report (2021)

[2] M. Baykara, Z. Z. Gürelr, 6th International Symposium on Digital Forensic and Security, 1 (2018)

[3] A. Vazhayil, R. Vinayakumar, K. P. Soman, 9th International Conference on Computing, Communication and Networking Technologies, 1 (2018)

[4] R. Kiruthiga and D. Akila, Int J Recent Technol Eng, **8** , 111 (2019)

[5] H.S. Hota, A.K. Shrivas, Rahul Hota, Procedia Comput. Sci., **132**, 900 (2018)

[6] H. Musa, A.Y Gital, F.U Zambuk, A. Umar, A.Y. Umar, J.U Waziri, J Theor Appl Inf Technol , **97**, 1434 (2019)

[7] V. Jagannath, Random Forest Template for Spotfire, TIBCO (2020)

[8] Z. Zhang, G. Mayer, Y. Dauvilliers, G. Plazzi, F. Pizza, R. Fronczek, J. Santamaria, M. Partinen, S. Overeem, R. Peraita-Adrados, et al., Sci. Rep, **8**, 1 (2018)