CLASSIFICATION OF WEBSITES FOR PHISHING DETECTION

A PROJECT REPORT
Submitted
by

SMITA SINDHU (1BM16CS107)
SUNIL PARAMESHWAR PATIL (1BM16CS112)
ARYA SREEVALSAN (1BM16CS133)
FAIZ RAHMAN (1BM16CS135)

in partial fulfillment for the award of the degree of
BACHELOR OF ENGINEERING
in
COMPUTER SCIENCE AND ENGINEERING

Under the Guidance
of
Ms. SARITHA A. N
Assistant Professor, CSE, BMSCE

B. M. S. COLLEGE OF ENGINEERING
(Autonomous Institution under VTU)
BENGALURU-560019
2019-2020

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

Certified that the project entitled "**CLASSIFICATION OF WEBSITES FOR PHISHING DETECTION**" is a bonafide work carried out by SMITA SINDHU (1BM16CS107), SUNIL PARAMESHWAR PATIL(1BM16CS112), ARYA SREEVALSAN (1BM16CS133), FAIZ RAHMAN (1BM16CS135) in partial fulfillment for the award of Bachelor of Engineering in Computer Science and Engineering of the Visvesvaraya Technological University, Belagavi during the academic year 2019-2020. The project report has been approved as it satisfies the academic requirements in respect of project work prescribed for the said degree.

Guide	Head of Department
Ms. Saritha A. N Assistant Professor, Dept. of Computer Science and Engineering B.M.S. College of Engineering	**Dr. Umadevi V** Associate Professor and HOD, Dept. of Computer Science and Engineering B.M.S. College of Engineering

Principal

Dr. B. V. Ravishankar

B.M.S. College of Engineering

External Viva

Name of the Examiners Signature with Date

1.

2.

3.

TABLE OF CONTENTS

ABSTRACT

Phishing is a major threat in today's world. Online fraudulent acts are on the rise with the fast-growing internet technology. Our primary motivation behind taking up this project was our common interest in the fields of Machine Learning and Security.

The report gives a proper introduction of the topic along with a brief literature survey. It gives specific details about our proposed approach to solve the defined problem including different types of requirements for completing the project, high level design and system architecture for the proposed system. In this project, we have improved Random Forest classification method, SVM classification method and Neural Network with backpropagation classification algorithm to detect phishing websites with better accuracies. The best classifier is incorporated in google chrome extension to classify websites as phishing or legitimate.

The report ends by enlisting the possible enhancements that can be worked upon by others in the future to add more functionality to our system and the mapping of program outcomes and how our project has helped us gain a better insight from these outcomes.

DECLARATION

We, hereby declare that the dissertation work entitled **"Classification of Websites for Phishing Detection"** is a bonafide work and has been carried out by us under the guidance of Ms. Saritha A.N., Assistant Professor, Department of Computer Science and Engineering, B.M.S. College of Engineering, Bengaluru, in partial fulfillment of the requirements of the degree of Bachelor of Engineering in Computer Science and Engineering of Visvesvaraya Technological University, Belagavi.

I further declare that, to the best of my knowledge and belief, this project has not been submitted either in part or in full to any other university for the award of any degree.

Candidate details:

SL. NO.	Student Name	USN	Student's Signature
1	SMITA SINDHU	1BM16CS107	
2	SUNIL PARAMESHWAR PATIL	1BM16CS112	
3	ARYA SREEVALSAN	1BM16CS133	
4	FAIZ RAHMAN	1BM16CS135	

Place: Bengaluru

Date:

Certified that these candidates are students of Computer Science and Engineering Department of B.M.S. College of Engineering. They have carried out the project work of titled **"Classification of Websites for Phishing Detection"** as final year (7^{th}& 8^{th}Semester) dissertation project. It is in partial fulfillment for completing the requirement for the award of B.E. by VTU. The works is original and duly certify the same.

Guide Name: Signature
Ms. Saritha A. N
Assistant Professor,
Department of Computer Science and Engineering
B.M.S. College of Engineering
Date:

ACKNOWLEDGEMENT

We have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals and organizations. We would like to extend our sincere thanks to all of them.

We are highly indebted to Ms. Saritha A. N., Assistant Professor, Department of Computer Science and Engineering, B.M.S. College of Engineering, Bengaluru for her guidance and constant supervision as well as for providing necessary information regarding the project & also for her support in completing the project.

We would like to express our gratitude towards our parents for their kind cooperation and encouragement which helped us in completion of this project.

We would like to express our special gratitude and thanks to industry persons for giving us such attention and time.

Our thanks and appreciations also go to our colleagues in developing the project and people who have willingly helped us out with their abilities.

LIST OF TABLES

LIST OF FIGURES

Chapter 1

INTRODUCTION

1.1 Overview

Phishing is a fraudulent practice in which an attacker's motive is to learn sensitive information like login credentials or account information by tricking people into believing that they are on a legitimate website and sell this information for financial purpose. One can even land on a phishing website by typing a wrong URL. We can detect phishing websites using several machine learning algorithms and deep learning techniques and hence prevent our sensitive information from being stolen.

1.2 Motivation

Due to extensive growth of Internet Technology, security threats to systems are on rise. One such serious threat is "phishing", in which an attacker attempts to steal user's sensitive information through fake websites. Hence it is important to make sure that victim is warned or stopped before the attack occurs, which is the reason phishing detection is important, as it helps to stop the attack from attack. It can help in protecting the victim from losing important and sensitive information.

1.3 Objective

The objective of this project is to create a Machine Learning model that takes URL in the form of user input. Lexical analysis is performed to extract features of URL and then Random Forest, SVM classification method and Neural Network with backpropagation classification algorithms are run to detect whether the URL is a legitimate website or a phishing website.

1.4 Scope

Build a system to automatically identify whether a URL corresponds to a phishing website or a legitimate one. The algorithm adopted in the system has to be extremely accurate otherwise people's sensitive information is likely to be stolen and misused.

1.5 Existing Systems

The following are the existing systems:

1. **DOM Based Phishing Detection**

   In DOM Tree Structure Matching algorithm, DOM tree structure of original and phishing websites are made and comparison is made between the two.When a user

provides same confidential information which has been used previously on some other site, a warning is generated. When information is used on a site that is completely different, it assumes legitimate data reuse.

2. **Anti-Phishing Database**

   This database has the list of phishing URLs and domains. If your mail contains any of the URLs or domains, it is automatically blocked. This method fails for newly created URLs.

3. **Genetic Algorithm Based Phishing Detection**

   If phishing is suspected (ex: -Same password used for different IP address site form), then using DOM API tree is generated for both pages. Then, first similarity of webpages is computed using selection, crossover and mutation with some pre-defined constants. If the computed value exceeds a pre-defined threshold(delta) then phishing is detected. Disadvantage of this method is that it requires a lot of trial and error to fix delta constant. Incorrect setting of constant can lead to false positives.

4. **Blacklist Based Phishing Detection**

   Blacklists hold URLs (or parts thereof) that refer to sites that are considered malicious. Whenever a browser loads page, it queries blacklist to determine whether currently visited URL is on this list. If so, appropriate countermeasures can be taken. Otherwise, the page is considered legitimate.

   Effectiveness of blacklist-based solution is the time it takes until a phishing site is included. Blacklists do not provide protection against zero-hour phishing attacks as a site needs to be previously detected first in order to be blacklisted.

1.6 Proposed System

Our proposed system will include the following features:

1. The proposed solution is used to identify whether the URL correspond to a phishing website or a legitimate website.
2. The dataset is loaded and its features are extracted.
3. Neural Network Classifier with Backpropagation model is used.
4. Random Forest Classification method is used.
5. SVM classification method is used.
6. The best classifier is selected and is used to classify URL as phishing or legitimate.

Chapter 2

LITERATURE SURVEY

1. Random Forest Classification

Amani Alswailem, Bashayr Alabdullah, Norah Alrumayh and Aram Alsedrani[1] experimented with the use of Random Forest Classification Model using a combination of features from 1 to 36. The dataset included 12000 URLs from PhishTank dataset and 4000 URLs from a survey of 10 users.

The features used for the model were based on URL, page content and rank-based. They established best maximum and minimum accuracy when combination of 29 features were used which were 98.8% and 91.33% respectively.

2. Support Vector Machines

Che-Yu Wu, Cheng-Chung Kuo, Chu-Sing Yang [2] proposed a model based on Support Vector Machine (SVM) with Fuzzy Logic replacing the Boolean Logic System. The model uses editing distance to get difference between domain names of main URL and contained URLs and similarly derived href and src based features. The experimental set used in this experiment was 5000 phishing pages taken by Phishtank with 10000 normal web pages collected by DMOZ a multilingual open content directory of World Wide Web links. The model TP rate is 92.6% and TN rate is 93.8%.

3. SQL Injection method

In the paper "Detection and Prevention Approach to SQLi and Phishing Attack using Machine Learning" [3], the method used is SQLi. In SQL injection method, a website is exploited using a definite pattern through SQL queries. Data can be retrieved from database where any one where clause in select query is true. To prevent this attack, the firewall is trained using Machine Learning algorithm. If the firewall detects the website to be malicious, it denies the permission to open the URL. If the website is safe, user is granted the permission to open it.

4. URL Phishing Data Analysis using NLP

In the paper "URL Phishing Data Analysis and Detecting Phishing Attacks using Machine Learning in NLP" [4], the method used is support vector machine. In this method, the model is trained using several features to distinguish phishing websites. The features considered are length of URL, IP address, sub-domain, HTTPs, symbols and website

traffic. The length of URL should not be more than 56 characters. Fake websites use symbols like @.If a website has more traffic, then it is considered to be a legitimate one. SVM uses a dataset and divides it into two classes. The hyper-plane is also divided into two. The aim is to find the hyper-plane with maximum margin between plane and a point in training set. The advantage of this method is that time taken to detect a phishing website is very less.

5.Bayesian Network Classifiers

In a paper uploaded by the faculty of the Firat University [5], the algorithm used for detection of phishing website is done by using Bayesian network classifiers. The model then calculates the conditional probability of these words. The BayesNet then calculates the common distribution probability of the whole set of words given in the dataset. The network does not require any prior of the given problem (which is phishing detection here). Using these results (that is collected in a database) one can classify whether a word belong to malicious category or not. Here the method used is BayesNet, that brings out the scores of words and finds out words that are exciting in nature. The scores are added back to the database. BayesNet hence classifies whether a site is malicious or not. The database that is maintained has two features, namely, “add spam” and “URL control”. The former is category that holds all the unwanted sites or URLs, hence making the detection readily available. While the latter is used by experts who wants an in depth understanding of malicious content and further fragmentation of the links of a website.

6. Random Forest and Generalized Linear Model

Ishant Tyagi, Shubham Sharma, Jatin Shad, Sidharth Gaur and Gagandeep Kaur [6] conducted an experiment using a combination of Random Forest (RF) and Generalized Linear Model (GLM) as a Generalized Additive Model (GAM). Gradient Boosting (GBM) was applied to the models to improve the model fitting. The primary dataset used was the PhishTank dataset. Around 30 URL features were extracted for the models. The accuracy without applying Principal Component Analysis (PCA) was 96.71% and with it was 98.40%.

7. Semantic Analyis and NLP

Tianrui Peng, Ian G. Harris, Yuki Sawa[7] proposed a model on NLP, the first semantic analysis of the text was performed to verify the appropriateness of each sentence. Then Natural Language Processing techniques are applied to parse each sentence. In this

method, each sentence is evaluated using NLP and determined if it exhibits characteristics like malicious question/command, urgent tone generic greeting and malicious URL link. Naive Bayes classifier is used to generate topic block list, which is a list of pairs whose presence in sentence suggests the malicious intent. This method has a precision of 95%.

8. Web Mining Technique

In the paper "Extraction of Features and Classification on Phishing Websites using Web Mining Techniques" [8], the method used is web mining technique where BOG (Bag of Words) representation model is used which is used to extract information from documents. BOG is used to classify documents. The document is classified and put into topic hierarchy where it best fits in. In this method, web phishing dataset is taken, pre-processed and features are selected. Then, various classification algorithms like Naive Bayes, Random Forest, KNN, SVM are used and their performances are assessed. Using Naive Bayes algorithm, 92.9806% of phishing data instances were classified correctly, using KNN,97.1777% of phishing data instances were classified correctly, using Random Forest,97.2592% of phishing data instances were classified correctly and using SVM,93.8037% of phishing data instances are classified correctly. Hence, Random Forest method achieves better performance than remaining algorithms.

9. Wrappers Features Selection method

In the paper "Phishing Website Detection based on Supervised Machine Learning with Wrappers Features Selection" [9], the method used is Wrappers Feature Selection that uses a classifier to predict significant features in predicting phishing websites. It is practically not possible to include all the features to train classifier. In this method, inductive classifier is used. The basic idea is to remove redundant features by training the classifier. For each features subset, a score is assigned depending on classification error rate of model. It provides most distinguished features set and improves the performance of Machine Learning classifier. This method uses N-fold cross validation technique to predict phishing websites. The small dataset is partitioned into n equal datasets and the model is trained using remaining datasets. This process is repeated n times. The final accuracy achieved is the average of n-accuracies obtained after running the classifier model n times. The TPR obtained using this method is 0.971 and FPR is 0.969. The advantages of this method is that it provides most important features used for classifier and also improves the performance of phishing website detection. The disadvantages of this method is that it is more time-consuming and involves extra computational overhead.

10. Filtering Methods

In another paper [10], the basic idea is to reduce the set of data or features taken into consideration to detect phishing. Hence filtering is done. Three filtering methods were chosen after conducting test on a sample of 47 features. The test was conducted to find out which filtering method gives the minimalistic set of features as output. The first method, Correlation-based feature's subset (CFS), filters out dataset by selecting the subset that contains useful attributes (high correlation to a class and less correlated to each other), calculated by the Person's correlation equation. The second method, Information gain (IG), calculates the how informative is an attribute and hence selects subsets with attributes having high IG. The third method, Chi-Square, calculates the relative frequencies between a class label and attribute value in an interval. This method is used for attributes with continuous values. These methods generate a small dataset typically consisting of 30 features which are used in rule-based algorithms that detect phishing.

11. Fuzzy Logic

K.N. Manoj Kumar, K. Alekhya[11] proposed a method using Fuzzy logic. In this method, the Fuzzy logic approach is used to determine the URL (website) legitimacy. This method uses It classifies URLs based on the set of rules and the degree if phishing is determined. It is done in 4 phases. 1.Fuzzification: In this step, crisp input is converted into fuzzy inputs. Membership Function is used to convert the crisp input into fuzzy input. 2. Evaluating Rules: if…then statements are used to evaluate phishing. if the input satisfies with given rule then it is treated as a legitimate website else fraudulent website. 3. Aggregating the rule outputs: In this step, the outputs of all rules are combined to form a single output or single fuzzy set. 4.Defuzzification: This is the final step, Defuzzification is a process where the fuzzy set is converted into crisp values. Website is categorized as highly legitimate, legitimate, suspicious, phished, highly phished based on the crisp value The results obtained from this model are not 100% accurate and designing this model is a little complex.

12. SVM with Adaline Network and SVM with Backpropagation

Priyanka Singh, Yogendra Maravi, Sanjeev Sharma [12] proposed a supervised learning approach to the problem, comparing two models-Support Vector Machine(SVM) with Adaline Network and SVM with Backpropagation. The dataset contained 358 URLs taken from Alexa and PhishTank, both legitimate and phishing. The experiment was conducted

on a system with CPU Intel Core i3 processor 2.30 GHz, RAM - 4 GB on a Windows 7 64-bit OS. Learning rate constant was 0.2 and number of iterations were 1000.The training time was 0.0729s and 0.0005s for Backpropagation and Adaline network respectively while Prediction Accuracy was 96.99% and 99.14% respectively.

13. Heuristic Image Based method

In "Web Phishing Detection in Machine Learning Using Heuristic Image Based Method" [13] paper, heuristic image-based method was adopted to combat the drawbacks found in blacklisting URLs. A phishing dictionary is maintained to store the images of websites for comparison. A website is labelled as phishing website if its visual similarity is higher than the threshold value. True Positive Rate of 90.06% and False Positive Rate of 1.95% were obtained using this technique. The advantages of this method is that it is more efficient than blacklisting method and easy to perform.

14. Whitelist and SVM

A. Belabed, E. Aïmeur, A. Chikh[14] proposed a method that uses a combination of whitelist with support vector machine (SVM). Phishing websites that are not blocked whitelist are passed to the SVM classifier. This method is designed implemented as an extension in a web browser. Degree of similarity is calculated using similarity matrix. If there is high similarity, then it considered legitimate if the domain name for visited page and pages in the whitelist are same. Otherwise, it is considered as a phishing site. If there is low similarity then the page is processed by the SVM classifier, which decides whether a page is legitimate or not. SVM classifier uses the features such as URL with IP address, special characters in the URL, presence of SSL certificate, frequency of links, Nil anchors to classify the website. This method detects 98 % of the phishing pages.

15. Phishing Characteristics in Webpage Source Code

In the paper "Phishing Websites Detection based on Phishing Characteristics in the Webpage Source Code" [15], the various features of the webpage are extracted such as image, http, source code, etc. each of these is given a small initial weight and given as an input to the model. The model calculates the final weights and accordingly gives a percentage to the input. High percentage indicates that the webpage is secure, medium is doubtful and low is risky and hence is a phished website.

16. Lexical Analysis

In a paper published by IEEE [16], the model uses the method of lexical analysis of URL to extract features that helps in detecting phished webpage. The training data is obtained from the lexical feature or the surface level features of a URL. This is then fed to a confidence-weighted learning algorithm. This algorithm then classifies or matches each binary vector from the URL to the binary vectors that it has been trained to detect a malicious website. A URL is split into three units: the protocol (e.g. http), the domain (the parent site or the one that follows the protocol) and path of the object being accessed. These are then converted to tokens. Any confidence-weighted level algorithm can be used here, only difference is that instead host based features, the algorithm uses lexical features of a URL. This method has produced a result with error rates lower than 3%.

17. Logistic Regression method

A paper published in WORM'07, November 2, 2007, Alexandria, Virginia, USA [17], identifies the different ways a site can be phished and the algorithms to identify these. The methods for obfuscating (making it a phishing website) a URL are: replacing a hostname with IP address, replacing a domain name with a fake but valid looking name, appending extra letters and numbers after the domain name and misspelling host and domain names. The model was trained with features that are categorized into four types Page Based, Domain Based, Type Based and Word Based features. Then a logistic regression is used to classify the input into phishing or benign URL. Using this technique around 777 unique webpages per day were found as phished website using with this model.

Chapter 3

SOFTWARE AND HARDWARE REQUIREMENT SPECIFICATION

3.1 Functional Requirements

1. The list of phishing websites (a large dataset).
2. The system should be able to produce required output based on the input provided.
3. There should be interaction between user and the system.
4. Complete analysis and prediction of the output based on the model.
5. Ease of usability and maintainability to user.

3.2 Non-Functional Requirements

We aim at attaining the following attributes of quality in our project:

1. Ease of use

   As our target population are the citizens of our society, the project developed must be easy to use requiring minimum assistance in interacting with the environment keeping in mind the physical limitations.

2. Data Integrity

   We need to ensure that data is accurate, complete and repeatable which is an utmost requirement for running the suggestive Deep learning algorithm to give accurate results.

3. Maintainability

   The error debugging process is also an essential factor as the rendering speed of the environment must be fast so as to produce the required output.

4. Reliability

   The system should propose a reliable outcome as it is concern with the life of people.

3.3 Hardware Requirements

CPU	Intel i5 +
Memory	8 GB+ RAM
OS	Windows, Linux

Table 3.1 Hardware Requirements

3.4 Software Requirements

Libraries Required:

1. scikit-learn(Version: 0.21.0): used for implementation of algorithms (random forest, SVM and neural networks)
2. Utils
3. Sys
4. Pandas (Version: 0.24.2)
5. Numpy (Version: 1.18.1)
6. ipaddress — IPv4/IPv6 manipulation library
7. Requests (Version: 2.23.0)
8. Re - Regular expression operations
9. Programming Language:Python

3.5 Cost Estimation

Our project does not require us to buy any hardware because all the hardware needs can be satisfied with our laptop computers. All the software required for our project are open-source, hence they are all free. So apart from the cost for publishing research paper and printing reports, it is costless.

Chapter 4

DESIGN

4.1 High level design

The proposed solution is to train the Random Forest, SVM and Neural Network with backpropagation models with the extracted features from dataset and after testing select optimum model for the system.

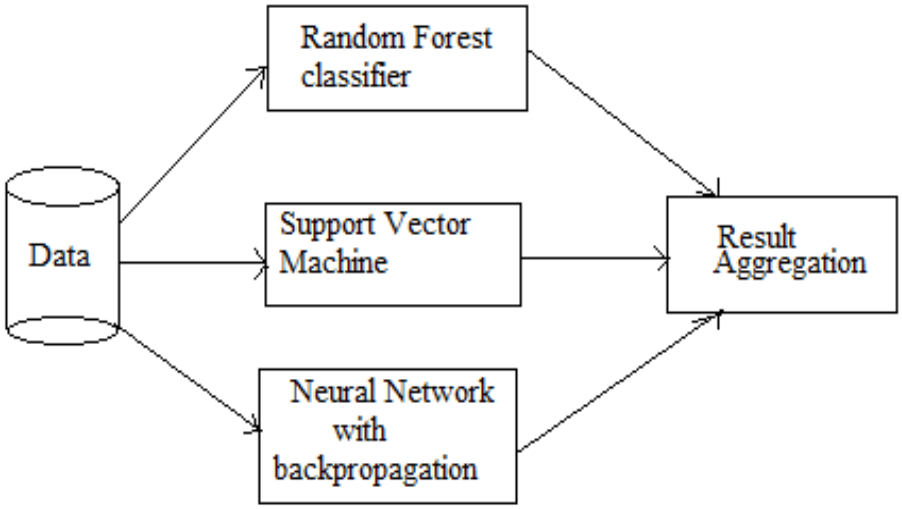

Figure 4.1: High Level design

The above figure depicts the high level approach adopted to the problem statement through the training of Random Forest, Support Vector Machine and Neural Network Models with our training dataset and aggregating the results to compare and select best model.

4.1.1 System Architecture

In the Figure 4.2, high level graphical representation of architecture of system is portrayed. Dataset was optimized through extraction of lexical features. After training and testing the proposed models with optimized dataset, selection of best classifier was done for deployment.

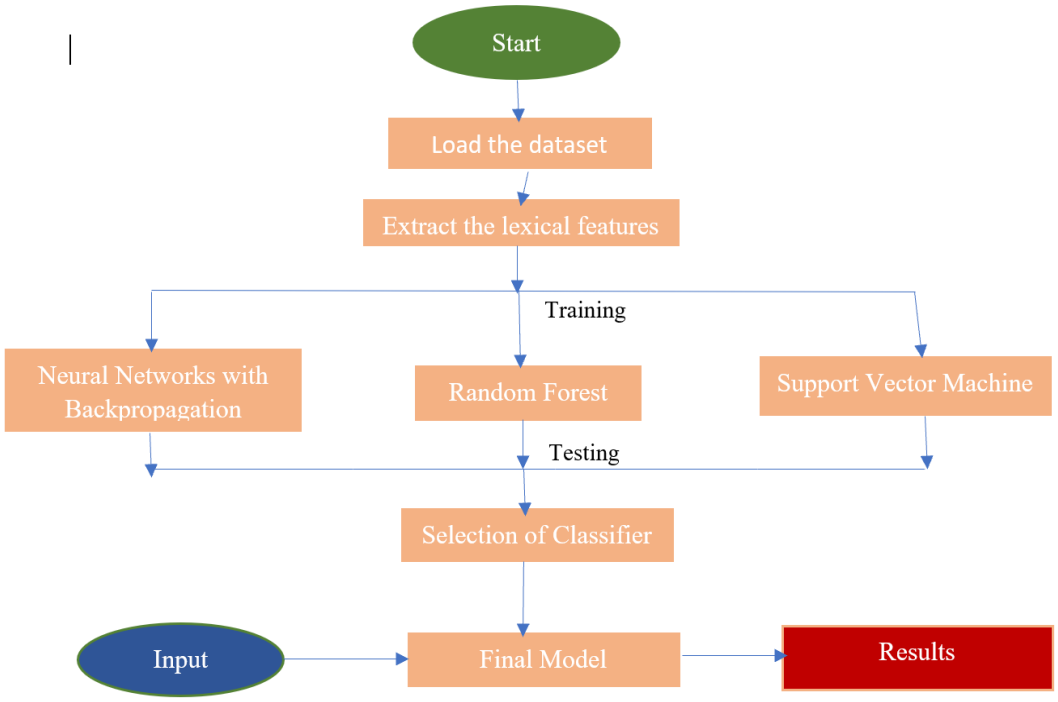

Figure 4.2: System Architecture

4.1.2 Use-case Diagram

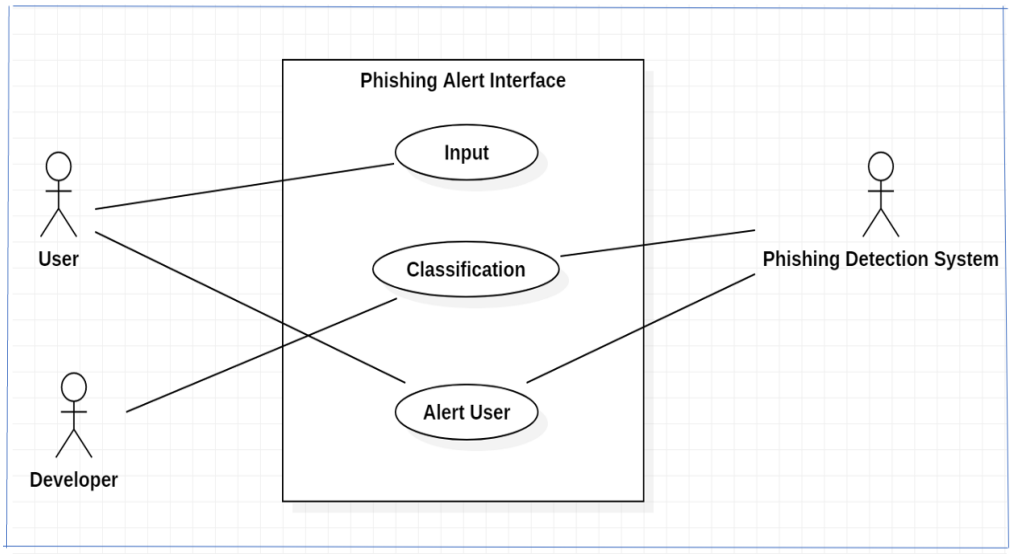

Figure 4.3: Use-case Diagram

The use case diagram for a normal execution is represented in the above figure. A user (who has loaded the chrome extension) opens a webpage, which is an input to the project. The developer handles the accuracy rates to classify whether a webpage is phished or legitimate. The phishing detection system classifies the webpage (phished or legitimate) and sends the result back to the interface. The interface then alerts the user that webpage open is phished or non-phished website through an alert pop-up.

4.2 Detailed Design

The detailed design of our solution involved construction of a classifier with our custom-made dataset. To select an optimum algorithm, we trained classifier of Random Forest, SVM and Neural Network with Backpropagation and got best results for SVM. We developed a chrome extension with JavaScript and integrated it with the trained classifier.

4.2.1 System Architecture

In the Figure 4.4, a detailed depiction of system architecture has been portrayed. For the classifier construction, lexically optimized dataset from UCI Machine Learning Repository was used to train classifiers-RF,SVM and NN with BP.

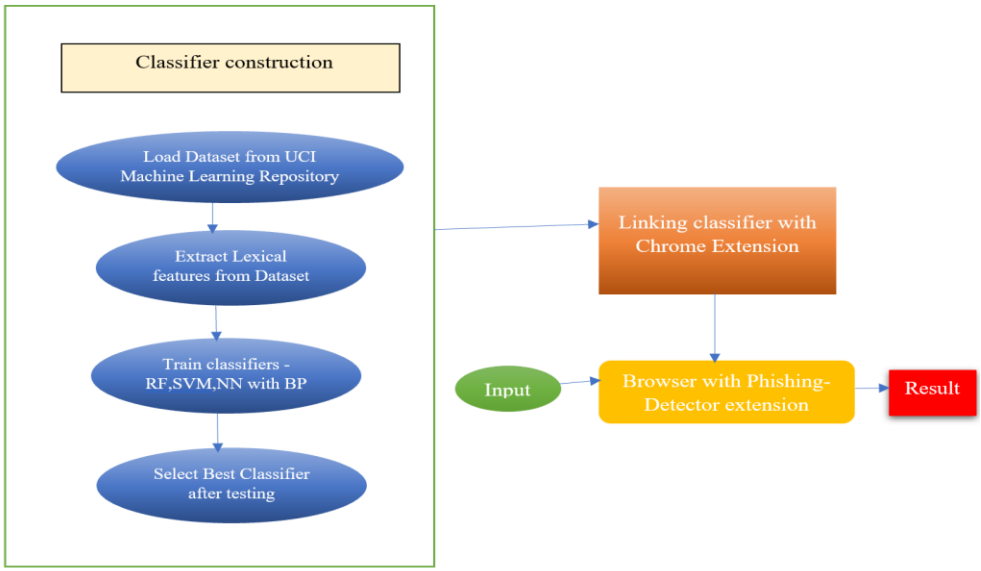

Figure 4.4: Detailed Architecture

The best classifier was linked with Chrome Extension and integrated with browser for classification of websites.

4.2.2 Use-case Diagram

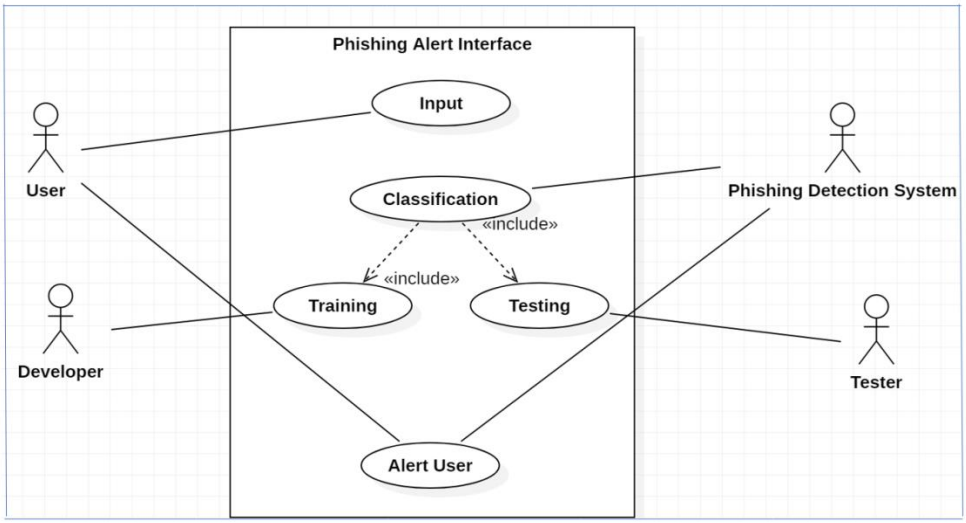

Figure 4.5: Detailed Use-case Diagram

In the above figure, a detailed depiction of the system's normal execution use case diagram is portrayed. A user (who has loaded the chrome extension) opens a webpage, which is an input to the project. The developer handles the accuracy rates to classify whether a webpage is phished or legitimate and hence is responsible for the selecting parameters for training and testing the project. The tester tests the result from the trained model with the test data. The phishing detection system takes the input, applies the input to the final trained model and gives the result back to interface.

4.2.3 Class Diagram

In our project, we have implemented the classifier and GUI for the system with Python and JavaScript. We didn't use any classes and simply connected the files through function calls. Hence, we have not included class diagram in the report to avoid misinterpretation.

4.2.4 Sequence Diagram

In the Figure 4.6, the sequence diagram for the use case diagram (fig 4.5) has been depicted. A user (who has loaded the chrome extension) opens a webpage, which is an input to the project.

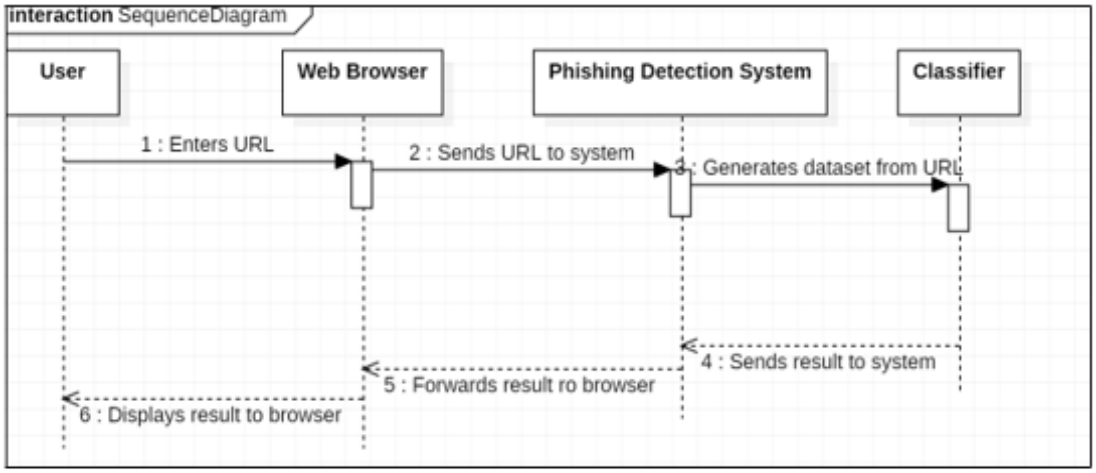

Figure 4.6: Sequence Diagram

The phishing detection system classifies the webpage (phished or legitimate) and sends the result back to the interface. The interface then alerts the user that webpage open is phished or non-phished website through an alert pop-up. The developer handles the accuracy rates to classify whether a webpage is phished or legitimate.

Chapter 5

IMPLEMENTATION

5.1 Overview of Technologies Used

Neural Network and Backpropagation:

Neural Network with backpropagation is used to detect the phishing websites.

Activation Function: Sigmoid Function

The Sigmoid Function curve looks like a S-shape.The main reason why we use sigmoid function is because it exists between 0 and 1. Therefore, it is especially used for models where we have to predict the probability as an output. Since probability of anything exists only between the range of 0 and 1**.**

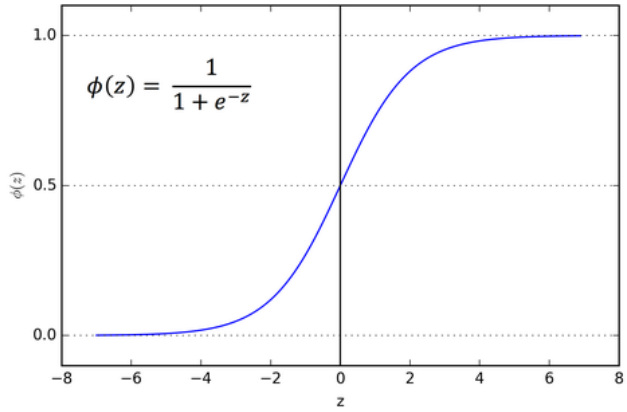

Figure 5.1: Sigmoid Function

Loss function:

Loss functionis basically a performance metric on how well the Neural Network manages to reach its goal of generating outputs as close as possible to the desired values.Loss function is calculated by the following method:

$$L = -y * \log(p) - (1-y) * \log(1-p) = \begin{cases} -\log(1-p), & if\ y = 0 \\ -\log(p), & if\ y = 1 \end{cases}$$

Figure 5.2: Loss Function

The **cross-entropy loss** for output label ‘y’ (can take values 0 and 1) and predicted probability ‘p’.

Back-propagation (BP): Back-propagation algorithms work by determining the loss (or error) at the output and then propagating it back into the network. The weights are updated to minimize the error resulting from each neuron.

Random Forest:

Random forest or random decision forest is used to classify the websites as phishing or legitimate. Random forest operates by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees.

The idea is that by training each tree on different samples, although each tree might have high variance with respect to a particular set of the training data, overall, the entire forest will have lower variance but not at the cost of increasing the bias. This procedure of training each individual learner on different bootstrapped subsets of the data and then averaging the predictions is known as bagging, short for bootstrap aggregating.

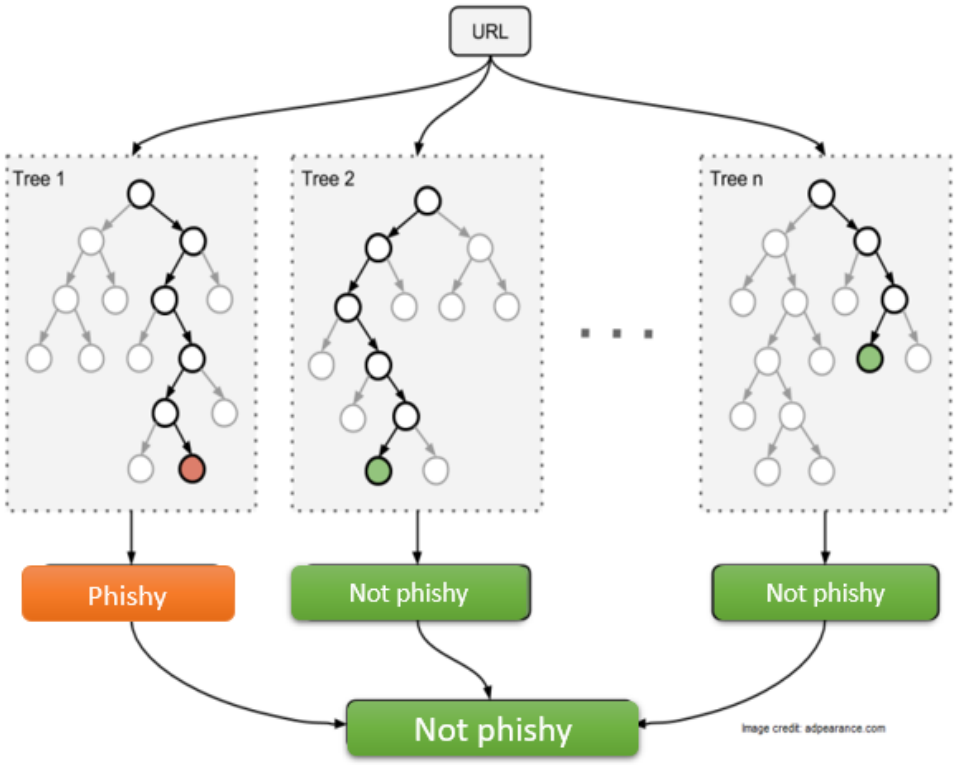

Figure 5.3: Random Forest Classification

We are using C4.5 decision tree algorithm to train the model. The new features of C4.5 (versus ID3) are: (i) accepts both continuous and discrete features; (ii) handles incomplete data points; (iii) solves over-fitting problem by (very clever) bottom-up technique usually known as "pruning"; and (iv) different weights can be applied the features that comprise the training data.

The information gain ratio is just the ratio between the information gain and the intrinsic value:

$$IG(T,a) = \mathrm{H}(T) - \mathrm{H}(T|a),$$

where $\mathrm{H}(T|a)$ is the conditional entropy of T given the value of attribute a.

$$IGR(Ex,a) = IG/IV$$

Figure 5.4: Information Gain

SVM:

The objective of the support vector machine algorithm is to find a hyperplane in an N-dimensional space that distinctly classifies the data points. To separate the two classes of data points, there are many hyperplanes that could be chosen. Our objective is to find a plane that has the maximum margin, i.e. the maximum distance between data points of both classes. Maximizing the margin distance provides some reinforcement so that future data points can be classified with more confidence.

Lexical Features Extraction:

The lexical feature extraction on URLs is performed to extract the most important features. Some of the URL features taken into consideration are IP Address, URL Length, sub-domain, domain registration length, @ symbol and request URL.

5.2 Implementation details of modules

- **URL classification using random forest**

The dataset is taken from UCI Machine Learning repository. Lexical analysis is performed on dataset and random forest classification method is run which is an ensemble learning method for classification. It can handle missing values. In this method, a number of classification trees are used which are created randomly by making use of different subsets of dataset to ensure that overfitting does not happen. The accuracy obtained using Random Forest method is 97.369%.

- **URL classification using SVM classification method**

The dataset used here is from UCI Machine Learning repository. After performing lexical features extraction to extract most important features of URL, SVM classification algorithm is run to classify the websites as phishing or legitimate. The accuracy achieved using this model is 97.451%.

- **URL classification using Neural Network with backpropagation**

The dataset used here is from UCI Machine Learning repository. After performing lexical analysis on the dataset, neural network with backpropagation classification algorithm is run to classify the websites as phishing or legitimate. The accuracy achieved using this model is 97.259%.

- **Lexical Features Extraction of dataset**

Lexical features extraction is performed on dataset to reduce the number of features in a dataset by creating new features from the existing ones and then discarding the original features. The new reduced set of features are able to summarize most of the information contained in the original set of features.

- **Chrome Extension to detect phishing websites**

The google chrome extension developed using JavaScript is used to warn user while actively preventing phishing attacks. The extension incorporates SVM classifier algorithm to detect the phishing websites.

5.3 Difficulties encountered and Strategies used to tackle

1. Chrome extension for jQuery was difficult to set up. Online tutorials help was taken for this.
2. The dataset had many features. The most important features were extracted using Lexical Features extraction.

Chapter 6

TESTING AND RESULTS

6.1 Unit Testing

Here we test separate modules of the project which is the feature extraction of a given input URL.

Unit	Expected Outcome	Actual Outcome
Extract domain	This should extract the domain name of URL.	Extracted DNS of URL.
URL_Length	This should return false if length of URL is greater than 75 characters.	Returned false for URLs with length greater than 75 characters, and true for URLs with length less than 75.
Get_port	This should return true if port number is present in URL. Otherwise it should return false.	Returned true if port number is present in URL.
Google_Index	If website is google indexed then return true, else false.	Returned true if is google indexed then return true.
Page_Rank	It should return true of page rank is below 100000, otherwise false.	Retuned true if of page rank is below 100000, otherwise false.

Table 6.1 Unit Testing

6.2 Integration Testing

On integrating the modules, we test whether the project as a whole works as expected (interactions between the modules work in right manner).

Module	Expected Outcome	Actual Outcome
Feature Extraction	Each of the feature set must show correct value for recorded URL.	URL features are evaluated after clicking on Evaluation.
Classifier	Should Classify Phishing website as Phishing and Legitimate Websites as Legitimate.	Classified Phishing website as Phishing and Legitimate Websites as Legitimate.

Table 6.2 Integration Testing

6.3 System Testing

This is the final testing of the project, wherein the observed output is compared to the excepted output (whether they match).

System	Expected Outcome	Actual Outcome
Send Phishy website to chrome extension	Chrome extension should block the website.	Chrome extension blocked the website
Send Legitimate website to chrome extension	Chrome extension should allow the website.	Chrome extension allowed the website.

Table 6.3 System Testing

6.4 Evaluation Metric

Random Forest classification method gave an accuracy of 97.835%, with an error of 2.165%. SVM classification method gave an accuracy of 97.89% with an error of 2.11 %. Neural Network with backpropagation classification algorithm gave an accuracy of 95.944% with an error of 4.056 %.

6.5 Experimental Dataset

The system is tested with the dataset that was obtained from the UCI - Machine Learning Repository which contains the Phishing Web Site Dataset. This dataset is composed of 11055 entries of websites which are classified as phishing and benign.

6.6 Performance Analysis

Random Forest classification method gave an accuracy of 97.369%. SVM classification method gave an accuracy of 97.451%. and Neural Network with backpropagation classification algorithm gave an accuracy of 97.259%.

The improved Random Forest classification method, SVM classification method and Neural Network with backpropagation classification algorithm gave better accuracies than existing methods.

ML Algorithm	Old Result Accuracy	New Result Accuracy(improved using lexical feature analysis on each algorithm)
Random Forest	87.34%	97.369%
Support Vector Machine	89.63%	97.451%
Neural Network with Backpropagation	89.84%	97.259%

Table 6.4 Comparison of accuracies of Machine Learning algorithms

Chapter 7

CONCLUSION AND FUTURE ENHANCEMENTS

7.1 Conclusion

While this phishing website isn't a fully fleshed-out unicorn product that will change the world, you can see just how easy it is to get started with it. It is remarkable to see the success of machine learning in such varied real-world problems.Detecting Phishing attacks will be a remarkable challenge in the future because malicious features evolve continually and unknown features are introduced daily. We have demonstrated how to classify positive and negative URL data from a basic URL input. The improved Random Forest classification method, SVM classification method and Neural Network with backpropagation classification algorithm gave better accuracies than existing methods. Random Forest classification method was implemented with an accuracy of 97.369%, SVM classification method with an accuracy of 97.451% and Neural Network with backpropagation classification algorithm was implemented with an accuracy of 97.259%.Since SVM gave better accuracy as compared to that of Random Forest classifier and Neural Network with backpropagation classifier, SVM is chosen as final classifier algorithm and was implemented in chrome extension for classification of websites as phishing or legitimate.

7.2 Future Enhancements

In the future, this work could be extended to detect and classify URL links in the content of the browser view and pass it to the system to classify. So unsuspecting users while browsing their emails are alerted about the URL being phishing before clicking the URL saving their time. This could be done by using a Deep Learning Model to detect text in the browsing content which is a URL.

REFERENCES

[1] Che-Yu Wu, Cheng-Chung Kuo , Chu-Sing Yang,"A Phishing Detection System Based on Machine Learning", International conference on Intelligent Computing and its Emerging Applications (ICEA) 2019.

[2] J. Jagadessan, Akshat Shrivastava, Arman Ansari, Laxmi Kanta, Mukul Kumar," Detection and Prevention Approach to SQLi and Phishing Attack using Machine Learning", International Journal of Engineering and Advanced Technology(IJEAT) ISSN:2249-8958, Issue-A:2019.

[3] Amani Alswailem, BashayrAlabdullah, Norah Alrumayh,AramAlsedrani ,"Detecting Phishing Websites Using Machine Learning" ,2nd International Conference on Computer Applications & Information Security 2019.

[4] Dr. G. Ravi Kumar, Dr. S. Gunasekaran, Nivetha R., Sangeetha Prabha K, Shanthini G., Vignesh A. S., "URL Phishing Data Analysis and Detecting Phishing Attacks using Machine Learning in NLP", International Journal of Engineering Applied Sciences and Technology(IJEAST) Vol. 3, Issue 8, ISSN No.2455-2143, Pages 70-75, Published: December 2018.

[5] MuhammetBaykara, ZahitZiyaGürel "Detection of Phishing Attacks", 6th International Symposium on Digital Forensic and Security (ISDFS), 22-25 March, 2018.

[6] Ishant Tyagi, Jatin Shad, Shubham Sharma, Sidharth Gaur & Gagandeep Kaur," A Novel Machine Learning Approach to Detect Phishing Websites".5th International Conference on Signal Processing and Integrated Networks (SPIN) Feb 2018.

[7] Tianrui Peng, Ian G. Harris, Yuki Sawa , "Detecting Phishing Attacks Using Natural Language Processing and Machine Learning" 12th IEEE International Conference on Semantic Computing 2018.

[8] S. Jagadeesan, Anchit Chaturvedi, Shashank Kumar, "URL Phishing Analysis using Random Forest", International Journal of Pure and Applied Mathematics, 2018.

[9] OzgurKoraySahingoz, SaideIşilay Baykal and Deniz Bulut, " Phishing Detection from URLs by using Neural Networks", International Conference on Computer Science engineering and Applications, 2018.

[10] Nandhini S., Dr. V. Vasanthi," Extraction of Features and Classification on Phishing Websites using Web Mining Techniques", IJEDR Volume 5, Issue 4, ISSN:2321-9939, Published:2017.

[11] Waleed Ali" Phishing Website Detection based on Supervised Machine Learning with Wrappers Features Selection", IJACSA (International Journal of Advanced Computer Science and Applications, Vol. 8 No. 9, Issue:2017.

[12] FadiThabtah, Neda Abdelhamid "Deriving Correlated Sets of Website Features for Phishing Detection: A Computational Intelligence Approach", Journal of Information & Knowledge Management Vol. 15, No. 4 (2016) 1650042 (17 pages) World Scientific Publishing Co., 25 November 2016.

[13] K.N. Manoj Kumar,K.Alekhya-"Detecting Phishing Websites using Fuzzy Logic", International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume 5, Issue 10, October 2016.

[14] Priyanka Singh, Yogendra P.S. Maravi, Sanjeev Sharma,"Phishing Websites Detection through Supervised Learning Networks", International Conference on Computing and Communications Technologies (ICCCT) 2015.

[15] VinnarasiTharania. I, R. Sangareswari, M. Saleembabu," Web Phishing Detection in Machine Learning Using Heuristic Image Based Method", International journal of

Engineering Research and Applications(IJERA) ISSN:2248-9622 Vol. 2, Issue: October 2012.

[16] A. Belabed, E. Aïmeur, A. Chikh "A personalized whitelist approach for phishing webpage detection" Seventh International Conference on Availability, Reliability and Security 2012.

[17] Mona GhotaishAlkhozae,Omar Abdullah Batarfi "Phishing Websites Detection based on Phishing Characteristics in the Webpage Source Code", International Journal of Information and Communication Technology Research, Volume 1 No. 6, October 2011

[18] Aaron Blum, Brad Wardman, Thamar Solorio, Gary Warner "Lexical Feature Based Phishing URL Detection Using Online Learning", Proceedings of the 3rd ACM Workshop on Security and Artificial Intelligence, AISec 2010, Chicago, Illinois, USA, October 8, 2010.

[19] Sujata Garera, Niels Provos, Monica Chew, Aviel D. Rubin "A framework for detection and measurement of phishing attacks", WORM '07 Proceedings of the 2007 ACM workshop on Recurring malcode Pages 1-8, Alexandria, Virginia, USA, November 02 - 02, 2007.

APPENDIX A: Results/Snapshots

1. Accuracy scores of each algorithm were improved from 87.34%, 89.63% and 89.84% to 97.259%, 97.369% and 97.451% for neural networks, random forest and SVM classifiers, respectively. The accuracy improvement was achieved by using lexical feature analysis.

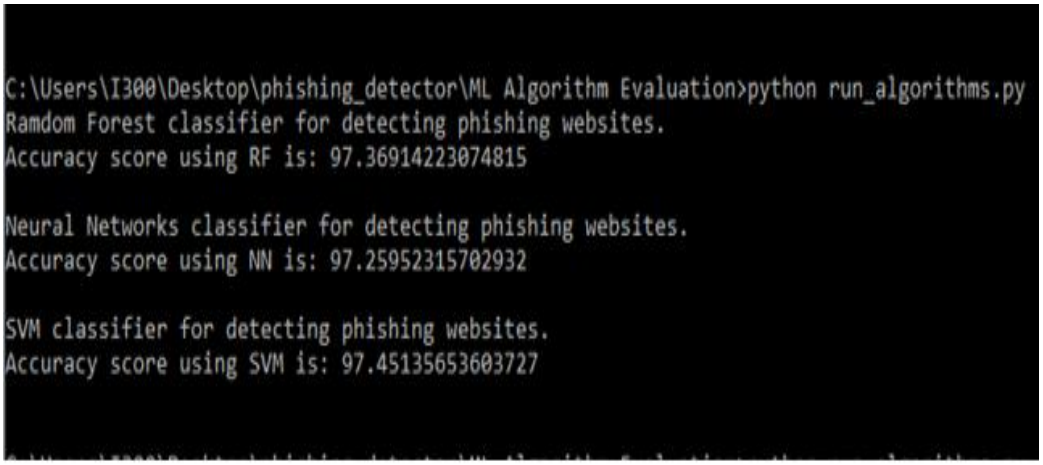

Figure 11: Accuracy score using Random Forest, Neural Network with backpropagation and SVM Classifier

2. Using Chrome Extension: the project on being deployed as a chrome extension can identify phished websites (Figure 12) and non-phished (Figure 13) websites and shows a pop-up alert, respectively.

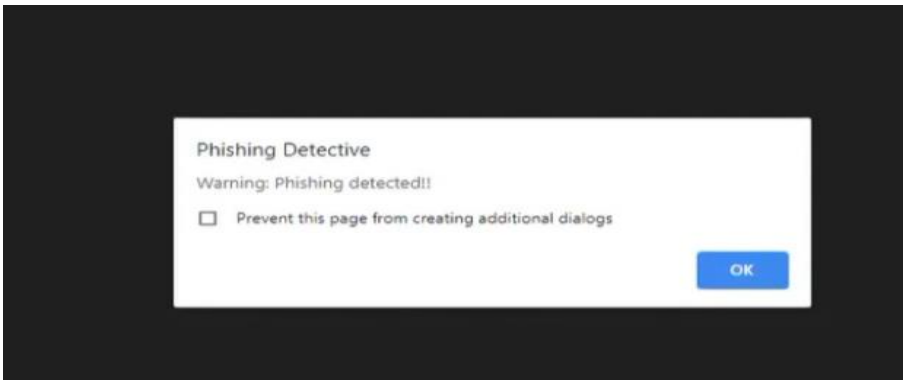

Figure 12: Chrome Extension giving warning “Phishing detected” when phishing URL is clicked

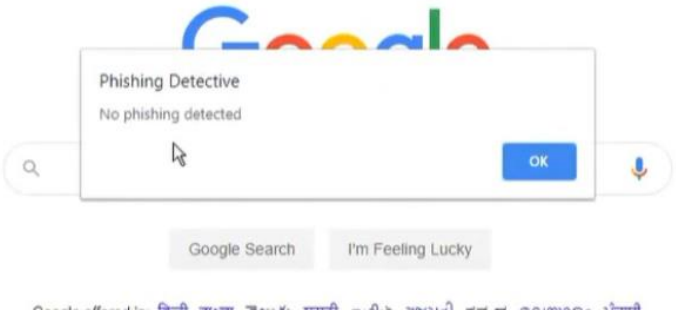

Figure 13: Chrome Extension showing "No phishing detected" when legitimate URL is clicked

APPENDIX B: Plagiarism report

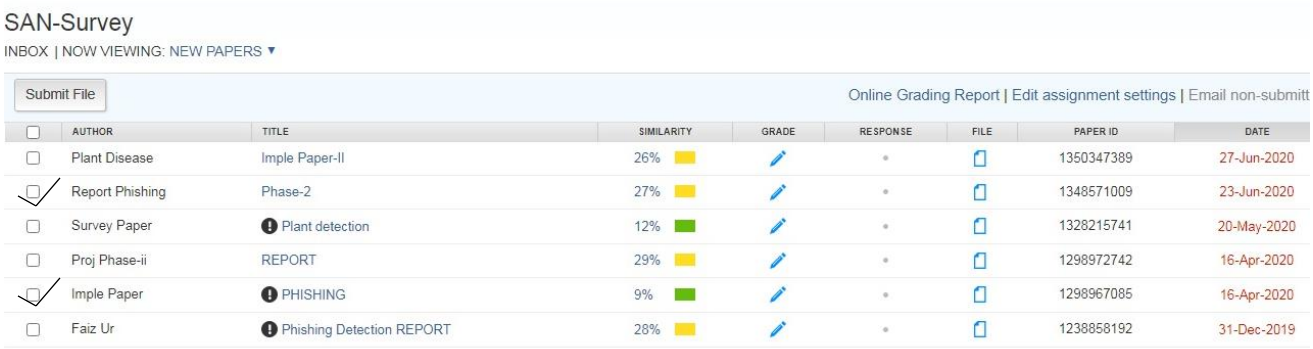

Figure 14: Plagiarism report – screenshot (1)

The plagiarism check using Turnitin software yielded the following results:

1. Report: 27%
2. Implementation Paper: 9%

Phase-2

ORIGINALITY REPORT

27%	16%	15%	21%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Visvesvaraya Technological University, Belagavi Student Paper	2%
2	Submitted to Higher Education Commission Pakistan Student Paper	1%
3	"Extreme Learning Model Based Phishing Classifier", International Journal of Recent Technology and Engineering, 2019 Publication	1%
4	Submitted to National University Of Science and Technology Student Paper	1%
5	Submitted to BMS College of Engineering Student Paper	1%
6	towardsdatascience.com Internet Source	1%
7	www.slideshare.net Internet Source	1%

Figure 15: Plagiarism report – screenshot (2)

APPENDIX C: Details of publications

Survey Paper:

Author Names: Smita Sindhu, Sunil P. Patil, Arya Sreevalsan, Faiz Rahman, Saritha A. N

Paper Title: Phishing Detection using Random Forest, SVM and Neural Network

Name of the Conference or Journal: International Conference on Recent Trends in Electrical, Electronics and Computer Science Engineering (ICEECS)

Place of the conference or Vol No., Issue No., Page No. of Journal: Vol-68-Issue-30-February-2020

Date of Conference or Date of Publication: 20/02/2020

Link of the paper: https://www.ourheritagejournal.com/index.php/oh/article/view/5836

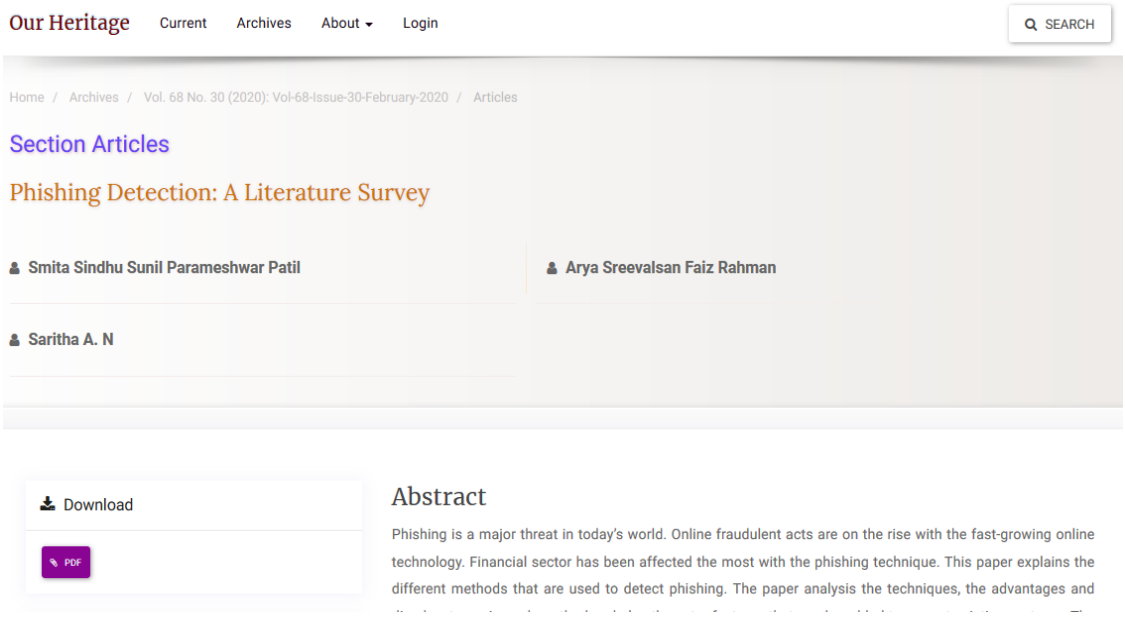

Figure 16: Published Survey Paper

Published survey paper:

Phishing Detection: A Literature Survey

Smita Sindhu
USN:1BM16CS107

Sunil Parameshwar Patil
USN:1BM16CS112

Arya Sreevalsan
USN:1BM16CS133

Faiz Rahman
USN:1BM16CS135

Department: CSE
B.M.S. College of Engineering, Bengaluru, Karnataka

Under the guidance of
Saritha A. N
Assistant Professor, CSE
B.M.S. College of Engineering, Bengaluru, Karnataka

***Abstract*— Phishing is a major threat in today's world. Online fraudulent acts are on the rise with the fast-growing online technology. Financial sector has been affected the most with the phishing technique. This paper explains the different methods that are used to detect phishing. The paper analysis the techniques, the advantages and disadvantages in each method and also the extra features that can be added to current existing systems. The paper is concluded with the proposed model for phishing detection.**

Keywords: *Phishing, Machine Learning, Deep Learning, NLP*

I. Introduction

Most of the phishing attacks are recreated from the previous attacks. The phisher tries to get confidential information from the users in order to benefit himself/herself in a malicious way. Phishing is a fraudulent practice that prompts people to reveal confidential information like username, password and credit card details by impersonating someone else. Common phishing attack is to get authentication details from the user and reuse it in another site because some people tend to reuse their passwords. The sensitive information is directed from phishing server to the phisher. It is harmful for both individuals and community. Phishers send emails which look similar to the one from government agencies to obtain confidential information. The URLs of phishing websites look similar to the original websites but differ in IP address. Most of the phishers use images rather than text which are difficult to detect. Various tools and mechanisms have been developed to detect phishing websites and to prevent attacker from obtaining sensitive information. Blacklisting is one the easy way to detect phishing websites but can't be used to find new phishing websites. It is also a time consuming process.

II. LITERATURE SURVEY

Amani Alswailem, Bashayr Alabdullah, Norah Alrumayh and Aram Alsedrani [1] experimented with the use of Random Forest Classification Model using a combination of features from 1 to 36. The dataset included 12000 URLs from PhishTank dataset and 4000 URLs from a survey of 10 users. The features used for the model were based on URL, page content and rank-based. This establishes best maximum and minimum accuracy when combination of 29 features were used which were 98.8% and 91.33% respectively.

Che-Yu Wu, Cheng-Chung Kuo, Chu-Sing Yang [2] proposed a model based on Support Vector Machine (SVM) with Fuzzy Logic replacing the Boolean Logic System. The model uses editing distance to get difference between domain names of main URL and contained URLs and similarly derived *href* and *src* based features. The experimental set used in this experiment was 5000 phishing pages taken by Phishtank with 10000 normal web pages collected by DMOZ a multilingual open content directory of World Wide Web links. The model TP rate is 92.6% and TN rate is 93.8%.

In the paper" Detection and Prevention Approach to SQLi and Phishing Attack using Machine Learning" [3], the method used is SQLi. In SQL injection method, a website is exploited using a definite pattern through SQL queries. Data can be retrieved from database where any one where clause in select query is true. To prevent this attack, the firewall is trained using Machine Learning algorithm. If the firewall detects the website to be malicious, it denies the permission to open the URL. If the website is safe, user is granted the permission to open it.

In the paper "URL Phishing Data Analysis and Detecting Phishing Attacks using Machine Learning in NLP" [4], the method used is support vector machine. In this method, the model is trained using several features to distinguish phishing websites. The features considered are length of URL, IP address, sub-domain, HTTPs, symbols and website traffic. The length of URL should not be more than 56 characters. Fake

Figure 17: Survey paper – Screenshot (1)

websites use symbols like @. If a website has more traffic, then it is considered to be a legitimate one. SVM uses a dataset and divides it into two classes. The hyper-plane is also divided into two. The aim is to find the hyper-plane with maximum margin between plane and a point in training set. The advantage of this method is that time taken to detect a phishing website is very less.

In a paper uploaded by the faculty of the Firat University [5], the algorithm used for detection of phishing website uses Bayesian network classifiers. This network classifier uses probabilistic approach. Each word is assigned a weight. The model then calculates the conditional probability of these words. The BayesNet then calculates the common distribution probability of the whole set of words given in the dataset. The network does not require any prior of the given problem (which is phishing detection here). Using this result (that is collected in a database) one can classify whether a word belong to malicious category or not. The model waits in loop for an input. Once input arrives, the content is matched with the database. Here BayesNetis used that brings out the scores of words and finds out words that are exciting in nature. The scores are added back to the database. BayesNet hence classifies whether a site is malicious or not. The database that is maintained has two features, namely, "add spam" and "URL control". The former is category that holds all the unwanted sites or URLs, hence making the detection readily available. While the latter is used by experts who wants an in depth understanding of malicious content and further fragmentation of the links of a website.

Ishant Tyagi, Shubham Sharma, Jatin Shad, Sidharth Gaur and Gagandeep Kaur [6] conducted an experiment using a combination of Random Forest (RF) and Generalized Linear Model (GLM) as a Generalized Additive Model (GAM). Gradient Boosting (GBM) was applied to the models to improve the model fitting. The primary dataset used was the PhishTank dataset. Around 30 URL features were extracted for the models. The accuracy without applying Principal Component Analysis (PCA) was 96.71% and with it was 98.40%.

Tianrui Peng, Ian G. Harris, Yuki Sawa [7] proposed a model on NLP, the first semantic analysis of the text was performed to verify the appropriateness of each sentence. Then Natural Language Processing techniques are applied to parse each sentence. In this method, each sentence is evaluated using NLP and determined if it exhibits characteristics like malicious question/command, urgent tone generic greeting and malicious URL link. Naive Bayes classifier is used to generate topic block list, which is a list of pairs whose presence in sentence suggests the malicious intent. This method has a precision of 95%.

In the paper "Extraction of Features and Classification on Phishing Websites using Web Mining Techniques" [8], the method used is web mining technique. This method, BOG (Bag of Words) representation model is used which is used to extract information from documents. Application of data mining techniques on text present in documents to extract useful information is called web content mining. BOG is used to classify documents. The document is classified and put into topic hierarchy where it best fits in. In this method, web phishing dataset is taken, pre-processed and features are selected. Then, various classification algorithms like Naive Bayes, Random Forest, KNN, SVM are used and their performances are assessed. Using Naive Bayes algorithm, 92.9806% of phishing data instances were classified correctly, using KNN,97.1777% of phishing data instances were classified correctly, using Random Forest,97.2592% of phishing data instances were classified correctly and using SVM,93.8037% of phishing data instances are classified correctly. Hence, Random Forest method achieves better performance than remaining algorithms.

In the paper "Phishing Website Detection based on Supervised Machine Learning with Wrappers Features Selection" [9], the method used is Wrappers Feature Selection that uses a classifier to predict significant features in predicting phishing websites. It is practically not possible to include all the features to train classifier. So, only the most distinguished features are included to train the classifier to detect phishing websites. In this method, inductive classifier is used. The basic idea is to remove redundant features by training the classifier. For each features subset, a score is assigned depending on classification error rate of model. It provides most distinguished features set and improves the performance of Machine Learning classifier. This method uses N-fold cross validation technique to predict phishing websites. The small dataset is partitioned into 'n' equal datasets and the model is trained using remaining datasets. This process is repeated n times. The final accuracy achieved is the average of n-accuracies obtained after running the classifier model n times. The TPR obtained using this method is 0.971 and FPR is 0.969. The advantages of this method is that it provides most important features used for classifier and also improves the performance of phishing website detection. The disadvantages of this method is that it is more time-consuming and involves extra computational overhead.

In another paper [10], the basic idea is to reduce the set of data or features taken into consideration to detect phishing. Hence filtering is done. Three filtering methods were chosen after conducting test on a sample of 47 features. The test was conducted to find out which filtering method gives the minimalistic set of features as output. The first method, Correlation-based feature's subset (CFS), filters out dataset by selecting the subset that contains useful attributes (high correlation to a class and less correlated to each other), calculated by the Person's correlation equation. The second method, Information gain (IG), calculates the how informative is an attribute and hence selects subsets with attributes having high IG. The third method, Chi-Square, calculates the relative frequencies between a class label and attribute value in an interval. This method is used for attributes with continuous values. These methods generate a small dataset typically consisting of 30 features which are used in rule-based algorithms that detect phishing.

Figure 18: Survey paper – Screenshot (2)

K.N. Manoj Kumar,K. Alekhya [11], proposed a method using Fuzzy logic , In this method, the Fuzzy logic approach is used to determine the URL(website) legitimacy. This method uses It classifies URLs based on the set of rules and the degree if phishing is determined. It is done in 4 phases:

1. Fuzzification: In this step, crisp input is converted into fuzzy inputs. Membership Function is used to convert the crisp input into fuzzy input.

2. Evaluating Rules: In this step, if…then statements are used to evaluate phishing. if the input satisfies with given rule then it is treated as a legitimate website else fraudulent website.

3. Aggregating the rule outputs: In this step, the outputs of all rules are combined to form a single output or single fuzzy set.

4. Defuzzification: This is the final step, Defuzzification is a process where the fuzzy set is converted into crisp values. The website is classified based on the final crisp value. Website is categorized as highly legitimate, legitimate, suspicious, phishy, highly phishy. This method uses very less memory when compared with other techniques and its inference speed is high. But the results obtained from this model are not 100% accurate and designing this model is a little complex.

Priyanka Singh,Yogendra Maravi,Sanjeev Sharma [12] proposed a supervised learning approach to the problem, comparing two models-Support Vector Machine(SVM) with Adaline Network and SVM with Backpropagation. The dataset contained 358 URLs taken from Alexa and PhishTank, both legitimate and phishing. The experiment was conducted on a system with CPU Intel Core i3 processor 2.30 GHz, RAM - 4 GB on a Windows 7 64 bit OS. Learning rate constant was 0.2 and number of iterations were 1000.The training time was 0.0729s and 0.0005s for Backpropagation and Adaline network respectively while Prediction Accuracy was 96.99% and 99.14% respectively.

In "Web Phishing Detection in Machine Learning Using Heuristic Image Based Method" [13] paper, heuristic image-based method was adopted to combat the drawbacks found in blacklisting URLs. A phishing dictionary is maintained to store the images of websites for comparison. A website is labelled as phishing website if its visual similarity is higher than the threshold value. True Positive Rate of 90.06% and False Positive Rate of 1.95% were obtained using this technique. The advantages of this method is that it is more efficient than blacklisting method and easy to perform.

A. Belabed, E. Aïmeur, A. Chikh [14] proposed a method that uses a combination of whitelist with support vector machine (SVM). Phishing websites that are not blocked whitelist are passed to the SVM classifier. This method is designed implemented as an extension in a web browser. Degree of similarity is calculated using similarity matrix. If there is high similarity, then it considered legitimate if the domain name for visited page and pages in the whitelist are same. Otherwise, it is considered as a phishing site. If there is low similarity then the page is processed by the SVM classifier, which decides whether a page is legitimate or not. SVM classifier uses the features such as URL with IP address, special characters in the URL, presence of SSL certificate, frequency of links, Nil anchors to classify the website. This method detects 98 % of the phishing pages.

In the paper "Phishing Websites Detection based on Phishing Characteristics in the Webpage Source Code" [15], the various features of the webpage are extracted such as image, http, source code, etc. each of these is given a small initial weight and given as an input to the model. The model calculates the final weights and accordingly gives a percentage to the input. High percentage indicates that the webpage is secure, medium is doubtful and low is risky and hence is a phished website.

In a paper published by IEEE [16], the model uses the method of lexical analysis of URL to extract features that helps in detecting phished webpage. The training data is obtained from the lexical feature or the surface level features of a URL. This is then fed to a confidence-weighted learning algorithm. This algorithm then classifies or matches each binary vector from the URL to the binary vectors that it has been trained to detect a malicious website. A URL is split into three units: the protocol (e.g. http), the domain (the parent site or the one that follows the protocol) and path of the object being accessed. These are then converted to tokens. A domain token usually helps in classifying an input URL as malicious or not and hence is considered as- fuzzy blacklist. Also there are certain rules maintained that can be used for classification based on the surface level features. Any confidence-weighted level algorithm can be used here, only difference is that instead host based features, the algorithm uses lexical features of a URL. This method has produced a result with error rates lower than 3%.

A paper published in WORM'07, November 2, 2007, Alexandria, Virginia, USA[17], identifies the different ways a site can be phished and the algorithms to identify these. The methods for obfuscating (making it a phishing website) a URL are: replacing a hostname with IP address, replacing a domain name with a fake but valid looking name, appending extra letters and numbers after the domain name and misspelling host and domain names. The model was trained with the dataset containing whitelist and blacklist. The model is trained with features that are categorized into four types Page Based, Domain Based, Type Based and Word Based features. Then a logistic regression is used to classify the input into phishing or benign URL. Using this technique around 777 unique webpages per day were found as phished website using with this model.

III. Conclusion

Different methods to detect phishing websites have been provided in this survey paper. Phishing attack is on the rise in today's world. Various techniques have been discussed to detect phishing websites. On the study and analysis of the different techniques, accuracy rates have proved to be higher in case of random forest model and neural networks. The error rates can be lowered by using lexical feature extraction to less than 3%. Hence the proposed model for phishing detection can be a combination of neural networks and random forest

Figure 19: Survey paper – Screenshot (3)

techniques, which can be trained after conducting a lexical feature extraction of the given URL.

REFERENCES

[1] Amani Alswailem, Bashayr Alabdullah, Norah Alrumayh,Aram Alsedrani, "Detecting Phishing Websites Using Machine Learning" ,2nd International Conference on Computer Applications & Information Security 2019.

[2] Che-Yu Wu, Cheng-Chung Kuo, Chu-Sing Yang, "A Phishing Detection System Based on Machine Learning", International conference on Intelligent Computing and its Emerging Applications (ICEA) 2019.

[3] J. Jagadessan, Akshat Shrivastava, Arman Ansari, Laxmi Kanta, Mukul Kumar, "Detection and Prevention Approach to SQLi and Phishing Attack using Machine Learning", International Journal of Engineering and Advanced Technology (IJEAT) ISSN:2249-8958, Issue-A:2019.

[4] Dr. G. Ravi Kumar, Dr. S. Gunasekaran, Nivetha R., Sangeetha Prabha K, Shanthini G., Vignesh A. S., "URL Phishing Data Analysis and Detecting Phishing Attacks using Machine Learning in NLP", International Journal of Engineering Applied Sciences and Technology(IJEAST) Vol. 3, Issue 8, ISSN No.2455-2143, Pages 70-75, Published: December 2018.

[5] Muhammet Baykara, Zahit Ziya Gürel, "Detection of Phishing Attacks", 6th International Symposium on Digital Forensic and Security (ISDFS), 22-25 March, 2018.

[6] Ishant Tyagi, Jatin Shad, Shubham Sharma, Sidharth Gaur & Gagandeep Kaur, "A Novel Machine Learning Approach to Detect Phishing Websites", 5th International Conference on Signal Processing and Integrated Networks (SPIN) Feb 2018.

[7] Tianrui Peng, Ian G. Harris, Yuki Sawa, "Detecting Phishing Attacks Using Natural Language Processing and Machine Learning", 12th IEEE International Conference on Semantic Computing 2018.

[8] Nandhini S., Dr. V. Vasanthi, "Extraction of Features and Classification on Phishing Websites using Web Mining Techniques", IJEDR Volume 5, Issue 4, ISSN:2321-9939, Published:2017.

[9] Waleed Ali, "Phishing Website Detection based on Supervised Machine Learning with Wrappers Features Selection", IJACSA (International Journal of Advanced Computer Science and Applications, Vol. 8 No. 9, Issue:2017.

[10] Fadi Thabtah, Neda Abdelhamid, "Deriving Correlated Sets of Website Features for Phishing Detection: A Computational Intelligence Approach", Journal of Information & Knowledge Management Vol. 15, No. 4 (2016) 1650042 (17 pages) World Scientific Publishing Co., 25 November 2016.

[11] K.N. Manoj Kumar,K. Alekhya- "Detecting Phishing Websites using Fuzzy Logic", International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume 5, Issue 10, October 2016.

[12] Priyanka Singh, Yogendra P.S. Maravi, Sanjeev Sharma, "Phishing Websites Detection through Supervised Learning Networks", International Conference on Computing and Communications Technologies (ICCCT) 2015.

[13] Vinnarasi Tharania. I, R. Sangareswari, M. Saleembabu, "Web Phishing Detection in Machine Learning Using Heuristic Image Based Method", International journal of Engineering Research and Applications (IJERA) ISSN:2248-9622 Vol. 2, Issue: October 2012.

[14] A. Belabed, E. Aïmeur, A. Chikh, "A personalized whitelist approach for phishing webpage detection", Seventh International Conference on Availability, Reliability and Security 2012.

[15] Mona Ghotaish Alkhozae, Omar Abdullah Batarfi, "Phishing Websites Detection based on Phishing Characteristics in the Webpage Source Code", International Journal of Information and Communication Technology Research, Volume 1 No. 6, October 2011.

[16] Aaron Blum, Brad Wardman, Thamar Solorio, Gary Warner, "Lexical Feature Based Phishing URL Detection Using Online Learning", Proceedings of the 3rd ACM Workshop on Security and Artificial Intelligence, AISec 2010, Chicago, Illinois, USA, October 8, 2010.

[17] Sujata Garera, Niels Provos, Monica Chew, Aviel D. Rubin, "A framework for detection and measurement of phishing attacks", WORM '07 Proceedings of the 2007 ACM workshop on Recurring malcode Pages 1-8, Alexandria, Virginia, USA, November 02 - 02, 2007.

Figure 20: Survey paper – Screenshot (4)

Implementation Paper

Author Names: Smita Sindhu, Sunil P. Patil, Arya Sreevalsan, Faiz Rahman,Saritha A. N

Paper Title: Phishing Detection using Random Forest, SVM and Neural Network with backpropagation

Name of the Conference or Journal: International Conference on Smart Technologies in Computing, Electrical and Electronics (ICSTCEE)

Place of the conference or Vol No., Issue No., Page No. of Journal: Reva University, Bengaluru, India

Date of Conference: 10-11 July 2020

[ICSTCEE 2020] Paper Acceptance Notification for Paper ID: 337

General Chair <chair@icstcee-conference.org>
to Arya, me, Sunil, Faiz, Saritha

Dear Arya Sreevalsan, Smita Sindhu, Sunil Patil, Faiz Rahman, Saritha A.N.,

Congratulations!

On behalf of the ICSTCEE 2020 Technical Program Committee, we are pleased to inform you that your paper:

Paper ID : 337
Title : Phishing Detection using Random Forest, SVM and Neural Network with Backpropagation

Author(s) : Arya Sreevalsan, Smita Sindhu, Sunil Patil, Faiz Rahman, Saritha A.N.,

has been approved for presentation at The International Conference on Smart Technologies in Computing, Electrical and Electronics (ICSTCEE 2020) will be held at REVA University, Bengaluru, India from 10 to 11th July 2020.

CoViD-19 Update : Please know that our thoughts are with those affected by

Figure 21: Acceptance of Implementation paper

APPENDIX D: Details of patents (if Any): -Nil-

APPENDIX E: Details of funding (if Any): -Nil-

APPENDIX F: Programme Outcomes and Programme Specific Outcomes Mapped

Batch no.: B7 Date: 28th June, 2020

Project Title: Classification of Websites Using Phishing Detection

PROGRAM OUTCOMES

PO	Level (3/2/1) 3-High 2-Medium 1-Low	Justification if addressed
PO1	High	Mathematical concepts and algorithms were applied for phishing detection and lexical analysis.
PO2	High	Various algorithms for classification from computer science were analyzed and suitable ones were incorporated.
PO3	High	Different methods were experimented and analyzed to draw conclusions.
PO4	High	Problems of classification of phishing websites for security purposes were addressed.
PO5	High	Python is used for the project as it performs well for the project and has support for lots of computing libraries.
PO6	High	The needs of our institution were addressed, mainly by proposing an automated system.
PO7	High	Unavailability of single solution to address problems related to classification of phishing websites.
PO8	High	Security of the system is maintained and unauthorized people cannot access the same.
PO9	High	Team completed tasks on time with proper work distribution.
PO10	High	Timely documentation in the form of Synopsis, Literature Survey and presentations have been completed.
PO11	High	Cost effective and highly efficient system has been the main goal and motivation of our project.
PO12	High	Various issues in educational institutions like parking admission and theft identification have been addressed which uses concepts from beyond syllabus have been effectively employed.

PROGRAM SPECIFIC OUTCOMES

PSO	Level (3/2/1) 3-High 2-Medium 1-Low	Justification if addressed
PSO1	3	Software Engineering Principles and Practices were applied in the project requirement collection and design phases.
PSO2	3	Phishing System design developed with realistic constraints.
PSO3	3	Analysis of different algorithms done and selection of best one done.