**DETECTING PHISHING WEBSITES USING**

**CLASSIFICATION IN DATA MINING TECHNIQUES**

**A PROJECT REPORT**

**Submitted by**

**M.PRIYA (810015104068)**

**S.VARSHA(810015104100)**

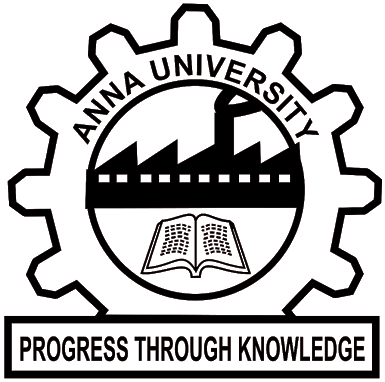
**in partial fulfilment for the award of the degree**

**of**

**BACHELOR OF ENGINEERING**

**IN**

**COMPUTER SCIENCE AND ENGINEERING**



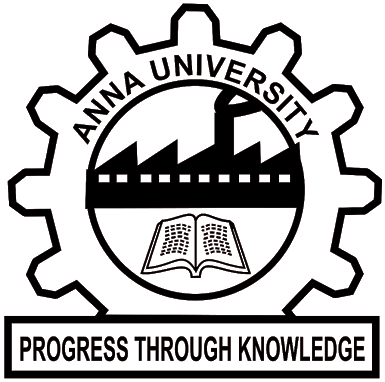
**UNIVERSITY COLLEGE OF ENGINEERING-BIT CAMPUS**

**ANNA UNIVERSITY, TIRUCHIRAPALLI-620 024**

**ANNA UNIVERSITY: CHENNAI-600 025**

APRIL 2019

**ANNA UNIVERSITY: CHENNAI 600 025**



**BONAFIDE CERTIFICATE**

Certified that this project report **“ DETECTING PHISHING WEBSITES USING CLASSIFICATION IN DATA MINING TECHNIQUES ”** is the bonafide work of **“M.PRIYA, S.VARSHA”** who carried out project work under my supervision.

**SIGNATURE SIGNATURE**

Dr.D.VENKATESAN Dr.M.PADMA

**HEAD OF THE DEPARTMENT TEACHING FELLOW**

Department of CSE Dept of CSE

UCE, BIT Campus UCE, BIT Campus

Anna University, Anna University,

Tiruchirappalli -24 Tiruchirappalli -24.

Submitted to University Viva-Voce Examination held on \_\_\_\_\_\_\_\_\_\_

**INTERNAL EXAMINER EXTERNAL EXAMINER**

**DECLARATION**

We hereby declare that the work “**DETECTING PHISHING WEBSITES USING CLASSIFICATION IN DATA MINING TECHNIQUES”** is submitted in partial fulfilment of the requirement for the award of the degree in B.E., UNIVERSITY COLLEGE OF ENGINEERING, BIT CAMPUS, TIRUCHIRAPPALLI. Is a record of our own work carried out by us during the academic year 2018-2019 under the supervision and guidance of **Dr.M.PADMA,** Teaching Fellow, Department of Computer Science and Engineering, UNIVERSITY COLLEGE OF ENGINEERING, BIT CAMPUS, TIRUCHIRAPPALLI. The extent and the source of information are derived from the existing literature and have been indicated through the dissertation at the appropriate places.The matter embodied in this work is original and has not been submitted for the award of any other degree or diploma, either in this or any other University.

**M.PRIYA – 810015104068**

**S.VARSHA – 810015104100**

I certify that the declaration made above by the candidate is true

**Signature of the Supervisor**

Dr.M.PADMA**,**

TeachingFellow, Department of CSE,

Anna University (BIT Campus),

Tiruchirappalli-620024.

**ACKNOWLEDGEMENT**

We wish to express our profound thanks to **Dr. T.SENTHILKUMAR, M.E., Ph.D.,** the Dean, University College of Engineering, BIT Campus, Anna University Tiruchirappalli, for granting us permission for doing this project work.

It is our privilege and honour to extend our sincere gratitude to **Dr.D.VENKATESAN,** Head of the Department of Computer Science and Engineering, University College of Engineering, BIT Campus, Anna University, Tiruchirappalli, for not only giving an opportunity to do the project work, but also to provide us the facilities and support throughout our study to improve the outcome of the project.

We take it as a privilege with great pleasure, deep satisfaction, immense respect and deep gratitude that we express our sincere thanks to our esteemed guide **Dr.M.PADMA,** Teaching Fellow, Department of CSE, UCE, BIT Campus, Anna University, Tiruchirappalli, for her continuous encouragement, invaluable and incredible guidance and ever-readiness to solve our problems throughout the period of this work.

We express our heartfelt thanks to our project coordinators, **Assistant professors,** Department of Computer Science and Engineering for their valuable advice and guidance throughout this work. We must thank them all from our depth of our heart.

**ABSTRACT**

Classification Data Mining (DM) Techniques can be a very useful tool for detecting phishing websites. In this work, presented a novel approach to overcome the difficulty and complexity in phishing website detection. Here proposed an intelligent resilient and effective model that is based on using classification Data Mining algorithms. These algorithms were used to characterize and identify all the factors and rules in order to classify the phishing website and the relationship that correlate them with each other. Here implemented five different classification algorithm and techniques to extract the phishing training data sets criteria to classify its legitimacy, and also compared performances relative such as, accuracy, number of rules generated and speed. A Phishing case study was applied to illustrate the website phishing process. The rules generated from the Classification model showed the relationship between some important characteristics like URL, Domain Identity, Security and Encryption criteria in the final phishing detection rate. The experimental results demonstrated the feasibility of using Classification techniques in real applications and its better performance as compared to other traditional classification algorithms.

**TABLE OF CONTENTS**

**CHAPTER NO. TITLE. PAGE NO.**

**ABSTRACT**

**LIST OF TABLE**

**LIST OF FIGURES**

**LIST OF SYMBOLS**

1. **INTRODUCTION** 1
   1. What is Phishing 1

1.2 Anatomy of a Real Phishing Attack 2

1.3 Phishing Motives 2

* 1. Mitigation of Phishing Attacks 3

1. **LITERATURE REVIEW** 4

2.1Governmental Policies Against Online Frauds 4

2.2 Strategies to Educated and Train Users

against Phishing 5

* 1. Technology based countermeasures to

automatically detect Phishing attack 6

2.3.1 Blacklist Based Approach 6

2.3.2 White-List Based Approach 7

2.3.3 Heuristics Based Phishing Detection 7

2.4 Existing System 8

**3.** **PROPOSED SYSTEM** 11

3.1 Introduction 11

3.2 Objective of the Study 12

3.3 Proposed system 12

3.4 Why use Data Mining? 12

3.5 Data Mining Algorithm 12

3.5.1 Naïve Bayes 13

3.5.2 j48 13

3.5.3 Logistic Regression 13

3.5.4 Random Forest 13

3.6 Feature Extraction 13

3.7 URL based Features 14

3.7.1 Using the IP Address 14

3.7.2 Long URL to hide the Suspicious part 14

3.7.3 Adding Prefix or Suffix Separated by ( - )

to the Domain 14

3.8 System Specifications 15

3.9 System Architecture 16

3.10 Advantages 16

**4.**  **Implementation** 17

4.1 Data Sets 17

4.2 Training data sets 17

4.3 Test data sets 17

4.4 Tools 17

4.5 Evaluation Metric 17

4.6 Source Code 18

**5. Output** 31

**6. Accuracy** 32

**7. Results** 36

**8. Conclusion and future work** 38

**9. References** 39

**LIST OF FIGURES**

**NO TITLE PAGE NO**

1.3 Phishing Targets 3

2.1 Phishing Webpage attacks 6

3.1 System Architecture 16

5.1 Output 31

6.1 Loading the dataset into WEKA tool 32

6.2 Naïve Bayes accuracy 33

6.3 Random Forest accuracy 34

6.4 Logistic accuracy 35

6.5 j48 accuracy 36

7.2 Accuracy of Classification Algorithm 37

**LIST OF ABBREVATIONS**

|  |  |  |
| --- | --- | --- |
| DM |  | Data Mining |
| URL |  | Uniform Resource Locator |
| SMS |  | Short Messaging Service |
| VoIP |  | Voice over Internet Protocol |
| APWG |  | Anti-Phishing Working Group |
| AIWL |  | Automated Individual White-List |
| DOM |  | Document Object Model |
| HTML |  | Hyper Text Markup Language |
| TP |  | True Positive |
| TN |  | True Negative |
| FP |  | False Positive |
| FN |  | False Negative |

**LIST OF TABLES**

**S.NO TITLE PAGE NO**

1 Confusion Matrix 18

2 Table accuracy 36

**CHAPTER 1**

**INTRODUCTION**

**1.1 What is Phishing?**

In today’s world, phishing is seen as a challenging threat of growing rapidly every year. It is considered as a criminal act that integrates social-engineering and technical methods to steal confidential data of consumers such as usernames and passwords.

Phishing websites is a semantic attack which targets the user rather than the computer. It is a relatively new Internet crime in comparison with other forms, e.g., virus and hacking. The attacker create an exact replica of the original website, which looks very convincing to users. This fake web page is used to performed phishing attack using social engineering techniques. The motive behind such attacks may be financial gain or notoriety (i.e., to get recognition).

Detection and prevention from phishing attacks is a big challenge to scientist and researcher because attacker performs these attacks in such a way that they bypass the existing anti-phishing techniques and even an educated and experience user may fall under these attacks. Usually, phishing attack is performed by sending a fake websites which appears to come from popular and trusted brand or organization, asking to input credential like bank login, password, etc.

Phishing messages are spread over using emails, SMSs, instant messages, social networking sites, VoIP, etc, however, email is the popular way to perform this attack. 65% of the total phishing attacks are performed by sending the malicious hyperlinks within the e-mail.

**1.2 Anatomy of a real phishing attack**

The Anti-Phishing Working Group (APWG) collects and archives examples of phishing attacks. It is a valuable service because phishing websites exist only for a short time before they get shut down or blocked. One example on APWG is an attack against eBay customers, ﬁrst reported on March 9, 2004 . The attack begins when a potential victim receives an email, purporting to be from eBay, that claims that the user’s account information is invalid and must be corrected. The email contains an embedded hyperlink that appears to point to a page on eBay’s website. This web page asks for the user’s credit card number, contact information, Social Security Number, and eBay username and password.

**1.3 Phishing Motives**

The Primary motive begin phishing attacks, from an attacker’s perspective, are:

**Financial Gain:** Phishers can use stolen banking credentials to their financial benefits.

**Identity Hiding:** Instead of using stolen identities directly, phishers might sell the identities to others whom might criminals seeking ways to hide their identities and a activities.

**Fame and Notoriety:** Phishers might attack victims for the sake of peer recognition.



Fig 1.3 – Phishing Targets

**1.4 Mitigation of Phishing Attacks**

Once the phishing attack is detected, a number of actions could be applied against the compaign. The following categories approaches exist are,

* + - * Detection Approach
      * Offensive Defence Approach
      * Correction Approach
      * Prevention Approach

**CHAPTER 2**

**LITERATURE REVIEW**

Several different solutions for phishing have been developed during the past few years. These solutions include government policies against online frauds, creating awareness to users and technology countermeasures. Review of these researches improved our basic understanding towards this problemand helped us to build a more compressive research model for the current study.

**2.1 Governmental Policies Against Online Frauds**

The progressive increasing phishing attack every year causes financial loss and reputational lose the individuals and companies. To mitigate this growing number of attacks, the number of laws and regulations has been introduced by governing bodies around the globe. In the following section, some of the laws designed against phishing attack are discussed.

The Fair and Accurate Credit Transactions Act of 2003, In section 113 states that the account number printed on the sale receipt of credit cards or debit cards transaction should be shortened. This legislation protects consumer’s private details from others who may have access to the sale receipt. Section 114 of FATCA, recommends the regulations to financial institutions to establish procedures to identify and safeguard the customers and institutions from identity theft. FATCA also contains several other deliberate provisions to reduce the identity theft. This law is commonly referred to as ”Red-flag Guidelines”.

Indian Government’s Information Technology Act of 2000, provides legal recognition of fund transactions carried out through electronic communications. Section 66 A of 2008 amendment, can be interpreted against phishing attacks.

Through this section sending of threatening, distressing messages and also false information about the source of the message has become punishable with imprisonment up to three years and /or fine (Shobhalata et al 2012).

**2.2 Strategies to Educated And Train Users Against Phishing**

Phishing is a criminal activity and largely exploits the human behaviour, so educating the users to understand about phishing prevents them from future attacks. Researches on this area focus on developing user skills to avoid misclassification of phishing websites as legitimate one manually.

Robila et al (2006) proposed a strategy for phishing education which is combination of phishing IQ test and discussion on context information this discussion includes nature of phishing emails, various methods of creating targeted messages and vulnerabilities of web browsers as well as web. Finally, the quality of the discussion was assessed by class assessments survey. The assessment results indicate that the participants were able to identify phishing attacks better.

Sheng et al (2007) developed a game “**Anti-phishing Phil”** using iterative design process, it teaches users not to fall victim to Phishing attacks. This game teaches users to differentiate phishing URLs from legitimate one, to interpret visual cues displayed in web browsers, and also teaches to use search engines to identify legitimate page when currently visiting page is suspicious. Authors had evaluated the game using the pilot test and observed the improvements among users in identifying phishing websites those who had played the game.

Kumaraguru et al (2007) developed an embedded training email system that regularly sends training emails to users during their normal use of email. These emails carry a message that tempts users to visit phished websites and log-in. If users click the link in the raised regarding their fall in phishing attack with some security information.

Odaro et al (2011) conducted a survey which revealed a lack of awareness among users regarding certificate of the website and also through this survey they educated the users on the same. Also, the research concluded that the Extended Validation Secure Sockets Layer as a security indicator is quiet effective against phishing attacks.

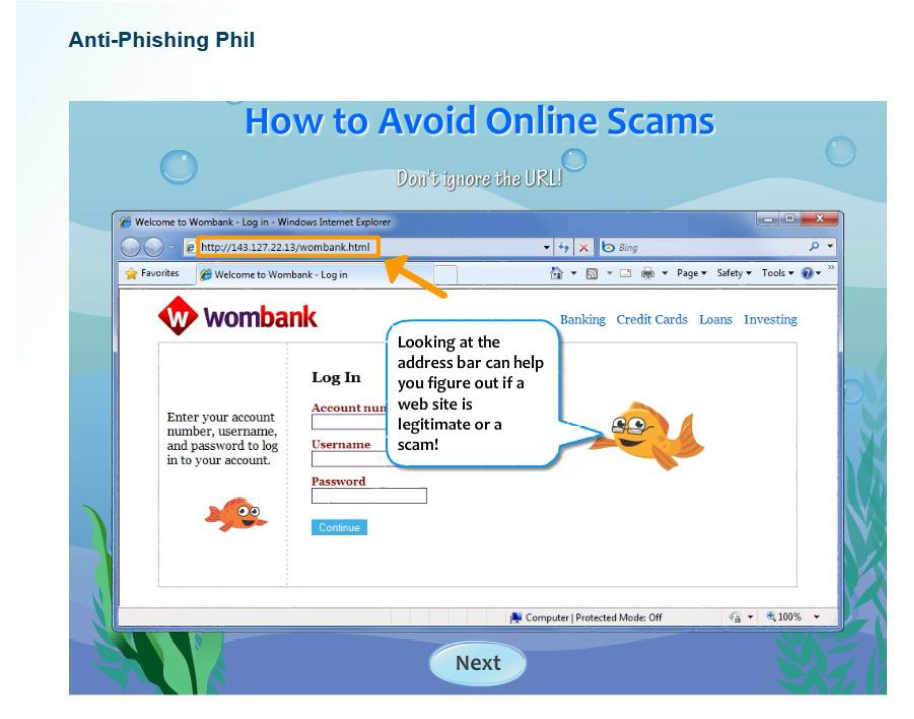


Fig 2.1 – Phishing Webpage Attacks

**2.3 Technology-based countermeasures to automatically detect Phishing attack**

This section, reviews some of the researches, offering technological solutions against a phishing attack. These researches are broadly grouped into three categories based on the techniques used to assess the genuineness of the webpage,

**2.3.1 Blacklist Based Approaches**

Blacklist approach maintain a list of known phishing sites to check the currently visiting website against the list. This blacklist is usually gathered from multiple data sources likes spam traps or spam filters, user post, and verified phish compiled by third parties such as takedown vendors or financial institutions.

Prakash et al (2010) proposed a predictive blacklist based approach to detect phishing websites. These method identifies new phishing URLs using five heuristics and an approximate matching algorithm. Heuristics used in this approach creates new URLs by combining pieces of known phishing URL from existing blacklist entries, and these new urls are tested for its presence in the web using a verification process. In the evaluation process, this method had discovered around 18,000 new phishing URLs from a set of 6,000 new blacklist entries.

**2.3.2 White-List Based Approach**

In contrast to the blacklist approach, white-list approach maintain list of all safe websites and their associated information. Any website that does not appear in the list is treated as a suspicious website.

Han et al (2012) developed an Automated Individual White-List (AIWL) approach to protect users online credentials in this methods users maintain a individual white-list that record the well known legitimate website the user visits, rather than maintaining a list of all legitimate website on the internet. In the white-list, along with URL, AIWL also maintains a list of features of a webpage where the user inputs his or her details.

**2.3.3 Heuristic based Approach**

Heuristic-based method extract features of a webpage to decide the legitimacy of the website instead of depending on any precompiled list. Most of the features are extracted from the URL and HTML Document Object Model (DOM) of the given webpage. The extracted features are compared with known features collected from phishing and legitimate pages to decide its legitimacy.

CANTINA+ is an upgraded version of CANTINA proposed by Xiang et al (2011) , where new components are included to achieve better results. Particularly, we have included ten other features along with four of the CANTINA feature and one extended feature. Both of these are falling in detecting phishing webpage made up of images since they have no text in the page to analysis.

**2.4 Existing System**

Many researchers have been studied and implemented the detecting Phishing websites through various Classification Algorithm. Some of techniques were used to detect the system. Weka tool has played a major role in calculating the accuracy of the system. These system have various drawbacks some of them are,

**A.CANTINA**

Zhang et al proposed page content based anti-phishing technique called CANTINA, which takes a rich set of feature set from various field of a webpage. Proposed technique calculates term frequency-inverse document frequency (TF-IDF) of the content of a website and creates a lexical signature. Top 5 terms with highest TF-IDF values submit into the search engine. The top ‘n’ result is used to check the legitimacy of a website. Though, the performance of CANTINA is affected by the TF-IDF algorithm and language used on the website. To reduce the false positive rate author also take some heuristic like the age of domain, URL contains “-” or “@” symbol, dots(.) count in URL, etc.

**Drawbacks**

Depend on the accuracy of the TF–IDF algorithm, fail to detect phishing page is embedded.

**B.Detection of Phishing attacks in Iranian e-banking**

Montaze et al [5] proposed phishing Detection system to be used in

e-banking system in Iran. Authors identified the 28 features used by the phisher to deceive the Irani banking sites. The accuracy to filter the phishing sites is 88% on Iranian banking system. The system uses the fuzzy expert system to train the system.

**Drawbacks**

False positive rate is high.

**C.The Comparison of Machine Learning Techniques**

Abu-Nimeh et al. [6] compared six machine learning algorithms for phishing detection. Machine learning algorithms are Bayesian Additive Regression Trees, Logical Regression, SVM, RF, Neural Network, and Regression Trees. The result shows that there are no standard machine learning algorithms for phishing detection. Authors trained dataset using 43 features. There is the trade-off between false positive rate and false negative rate, if some algorithm has low false Positive rate then it has a high false negative rate. Logical Regression whose false positive rate is 4.9%, obtained a large number of false negative rate of 17%.

**D.Associative Classification Data Mining**

Neda Abdelhamid et al [7] proposed Multi-label Classifier based Associative Classification method for website phishing. Associate classification is effectively detecting phishing websites with high accuracy and MCAC are used to generate new rules and it also enhances its classifiers predictive performance.

**Drawbacks**

Accuracy of the system is not explained.

**E.Classification mining techniques**

Aburrous M present a technique based on Classification Data Mining (DM) for detection on e-banking phishing websites. Author implement the six different classification algorithm to measure the performance and accuracy of each of algorithm, the drawback of this algorithm is that the false positive rate is very high (13%).

**Drawbacks**

False positive rate is very high.

**F.An SVM-based technique to detect phishing URLs**

H Huang et present an approach based on the URL based features. They have taken 23 features from URL and train the system using SVM. The system takes Decision based on lexical and brand name features of URL.

**Drawbacks**

Based on URL feature so accuracy is low.

**G.Learning to detect phishing emails**

Fette et al. proposed a technique to classify phishing and legitimate emails. They used a large publicly available corpus of genuine and fake emails. Their learning classifiers examine 10 different features such as the presence of hyperlink an email, IP Address, age of link, etc

**Drawbacks**

Phishing and legitimate email not well classified.

**CHAPTER 3**

**PROPOSED SYSTEM**

**3.1 Introduction**

This section describes the proposed model of phishing attack detection. The proposed model focuses on identifying the phishing attack based on checking phishing websites features, Blacklist and WHOIS database. According to a few selected features can be used to differentiate between legitimate and spoofed web pages. These selected features are many such as URLs, domain identity, security & encryption, source code, page style and contents, web address bar and social human factor. This study focuses only on URLs and domain name features. Features of URLs and domain names are checked using several criteria such as IP Address, long URL address, adding a prefix or suffix, redirecting using the symbol “//”, and URLs having the symbol “@”. These features are inspected using a set of rules in order to distinguish URLs of phishing webpages from the URLs of legitimate.

**3.2 Objective of the Study**

* + - * To detect whether the website is phishing or not.
      * To calculate the accuracy of classification algorithm.
      * To find false positive rate.

**3.3 Proposed system**

The Existing System deals mainly with detecting Phishing emails using Associative Classification Algorithm and higher false positive rate. But in this work, proposed a method to detect to Phishing Websites that we used in day-to-day activities. Some of them not aware of the websites that they are legitimate or phishing. So, the proposed framework was detecting the websites whether it is legitimate or not using classification algorithm in Data mining.

**3.4 Why use Data Mining?**

Data Mining has been described as "the nontrivial extraction of implicit, unknown, and potentially useful information from large data sets. It is a powerful new technology to help researchers focus on the important information in their data archive. Data mining tools predict future trends and behaviours, allowing businesses to make proactive, knowledge-driven decisions.

**3.5 Data Mining Algorithm**

Here, implement Five different classification algorithms, to train and test the accuracy of phishing email detection with the grouped features. The reason behind selecting these algorithms is the different training strategy they use in discovering the rules and the mechanism of learning and testing, the below listed selected algorithms are considered as well-known algorithms.

* Naive Bayes
* Random Forest
* J48
* Logistic Regression

**3.5.1 Naive Bayes**

The Naive Bayes algorithm is an intuitive method that uses the probabilities of each attributes belonging to each class to make a prediction.

**P(H|E) = P(E|H) \* P(H) / P(E)**

**3.5.2 J48**

Decision Tree Algorithm is to find out the way attributes-vector behaves for a number of instances. In the WEKA data mining tool, J48 is an open source java implementation of the C4.5 algorithm.

**3.5.3 Logistic Regression**

Logistic Regression is one of the basic and popular algorithm to solve the classification problem. It is used to predict binary outcomes for a given set of independent variables. The dependent variables outcome is discrete.

**3.5.4 Random Forest**

Random Forest is an ensemble learning method for classification, regression and other task that operates by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes or mean prediction of the individual trees.

**3.6 Feature Extraction**

Preprocessing of Phishing set

Web phishing data set

Feature Selection

Applying Classification Algorithm

Performance Measures

* 1. **URL Based Features**

**3.7.1 Using the IP address**

If an IP address is used as an alternative of the domain name in the URL, such as “http://125.98.3.123/fake.html”, users can be sure that someone is trying to steal their personal sensitive information. Sometimes, the IP address is even transformed into hexadecimal code as shown in the following link

**“http://0x58.0xCC.0xCA.0x62/2/paypal.ca/index.html”.**

Rule: IF The Domain Part has an IP

Address → Phishing

Otherwise→ Legitimate

**3.7.2 Long URL to Hide the Suspicious Part**

Phishers can use long URL to hide the doubtful part in the address bar.

For example:

**“http://federmacedoadv.com.br/3f/aze/ab51e2e319e51502f416dbe46b773a5e/?cmd=\_home&amp;dispatch=11004d58f5b74f8dc1e7c2e8dd4105e811004d58f5b74f8dc1e7c2e8dd 4105e8@phishing.website.html “**

To ensure the accuracy, calculated the length of URLs in the dataset and produced an average URL length. The results showed that if the length of the URL is greater than or equal 54 characters then the URL classified as phishing. By reviewing our dataset we were able to find 1220 URLs lengths equals to 54 or more which constitute 48.8% of the total dataset size.

**Rule:**

IF URL length is ≤ 75→legitimate

otherwise→ Phishing

We have been able to update this feature rule by using a method based on frequency and thus improving upon its accuracy.

* + 1. **Adding Prefix or Suffix Separated by (-) to the Domain**

The dash symbol is rarely used in legitimate URLs. Phishers tend to add prefixes or suffixes separated by (-) to the domain name so that users feel that they are dealing with a legitimate webpage.

For example :http://www.confirmepaypal.com/.

**Rule:**

IF Domain Name Part Includes (-)Symbol → Phishing

Otherwise → Legitimate

**3.8 System Specifications**

**Software Requirements**

* + - * Windows 7 or higher
      * Weka tool (Machine Learning Algorithm)
      * Idle Python 3.7

**Hardware Requirements**

* Internet Connection

**3.9 System Architecture**

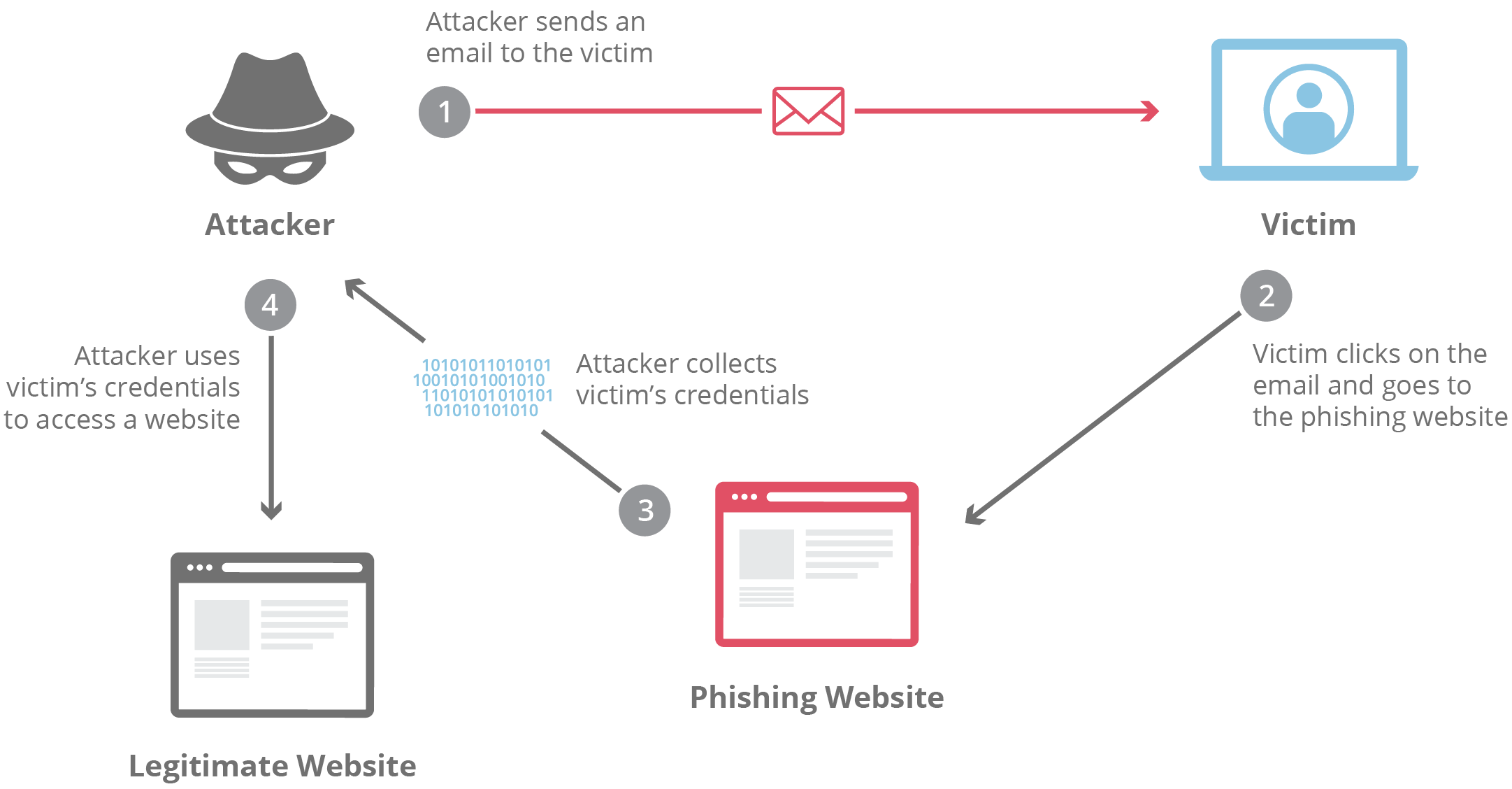


Fig 3.1 – System Architecture

**3.10 Advantages**

Data mining algorithm used in this system provides better performance as compared to another traditional classification algorithm.

**CHAPTER 4**

**IMPLEMENTATION**

**4.1 Data Sets**

The utilized data set contains 11,0000 phishing and legitimate websites. The websites are obtained from www.phishtank.com

**4.2 Training data sets**

In Data Mining, a training set is a dataset used to train a model. In training

model, specific features are picked out from the training set. The features are then incorporated into the model.

**4.3 Test data sets**

Typically, separate a datasets into test sets and training sets, most of the data is used for testing, and a smaller portion of the data is used for testing. Among 11,000 datasets, here implemented using 100 datasets with 32 attributes.

**4.4 Tools**

The file was converted to CSV format in order to be compatible to be tested with the selected five algorithms through WEKA tool. Weka is a group of machine learning algorithms used for data mining tasks, and it’s contain several tools for data pre-processing, regression, classification, association rules, visualization and clustering. The algorithms can either be implemented on the dataset directly or called from a Java code. It is also quite suitable for developing new machine learning schemes.

**4.5 Evaluation Metric**

Accuracy is the rate of correct predictions that the model achieving when

compared with the actual classifications in the dataset. On the other hand, Precision and recall are two evaluation techniques, which calculated based on confusion matrix as

**Precision** = TP / TP + FP

**Recall =** TP /TP + FN

**Accuracy =** ( TP+TN ) / ( TP + FP + TN + FN )

Where,

True Positive (TP): The number of correct detected phishing websites.

False Negative (FN): The number of phishing websites was detected as legitimate websites.

False Positive (FP): The number of legitimate websites was detected as phishing website,

True Negative (TN): The number of legitimate websites was detected as legitimate websites.

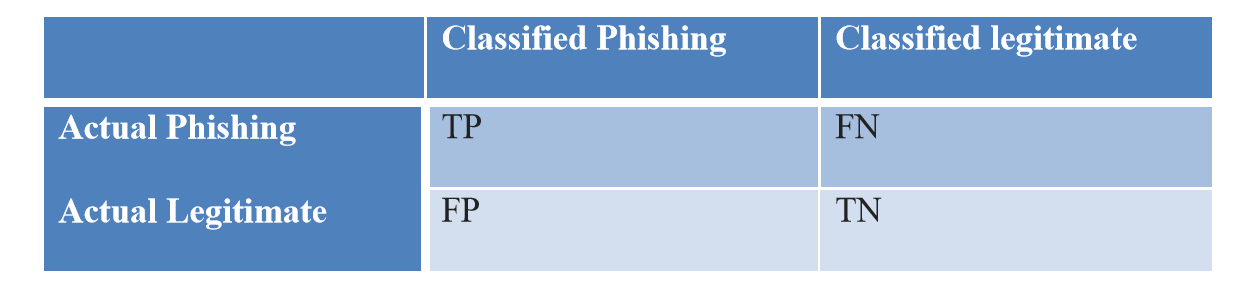


Table 4.1 Confusion Matrix

**4.6 Source Code**

**Index.py**

# -\*- coding: utf-8 -\*-

**#importing libraries**

from sklearn.externals import joblib

import warnings

import inputScript

warnings.filterwarnings('ignore')

**#load the pickle file**

classifier = joblib.load('final\_models/svm\_final.pkl')

**#input url**

print("enter url")

url = input()

**#checking and predicting**

checkprediction = inputScript.main(url)

prediction = classifier.predict(checkprediction)

print(prediction)

if prediction==-1:

print("Not a phishing website")

else:

print("Phishing website")

**inputScript.py**

**# -\*- coding: utf-8 -\*-**

import regex

from tldextract import extract

import ssl

import socket

from bs4 import BeautifulSoup

import urllib.request

import whois

import datetime

def url\_having\_ip(url):

**#using regular function**

# symbol = regex.findall(r'(http((s)?)://)((((\d)+).)\*)((\w)+)(/((\w)+))?',url)

# if(len(symbol)!=0):

# having\_ip = 1 #phishing

# else:

# having\_ip = -1 #legitimate

#return(having\_ip)

return 0

sss

def url\_length(url):

length=len(url)

if(length<54):

return -1

elif(54<=length<=75):

return 0

else:

return 1

def url\_short(url):

**#ongoing**

return 0

def having\_at\_symbol(url):

symbol=regex.findall(r'@',url)

if(len(symbol)==0):

return -1

else:

return 1

def doubleSlash(url):

**#ongoing**

return 0

def prefix\_suffix(url):

subDomain, domain, suffix = extract(url)

if(domain.count('-')):

return 1

else:

return -1

def sub\_domain(url):

subDomain, domain, suffix = extract(url)

if(subDomain.count('.')==0):

return -1

elif(subDomain.count('.')==1):

return 0

else:

return 1

def SSLfinal\_State(url):

try:

**#check wheather contains https**

if(regex.search('^https',url)):

usehttps = 1

else:

usehttps = 0

**#getting the certificate issuer to later compare with trusted issuer**

**#getting host name**

subDomain, domain, suffix = extract(url)

host\_name = domain + "." + suffix

context = ssl.create\_default\_context()

sct = context.wrap\_socket(socket.socket(), server\_hostname = host\_name)

sct.connect((host\_name, 443))

certificate = sct.getpeercert()

issuer = dict(x[0] for x in certificate['issuer'])

certificate\_Auth = str(issuer['commonName'])

certificate\_Auth = certificate\_Auth.split()

if(certificate\_Auth[0] == "Network" or certificate\_Auth == "Deutsche"):

certificate\_Auth = certificate\_Auth[0] + " " + certificate\_Auth[1]

else:

certificate\_Auth = certificate\_Auth[0]

trusted\_Auth = ['Comodo','Symantec','GoDaddy','GlobalSign','DigiCert','StartCom','Entrust','Verizon','Trustwave','Unizeto','Buypass','QuoVadis','Deutsche Telekom','Network Solutions','SwissSign','IdenTrust','Secom','TWCA','GeoTrust','Thawte','Doster','VeriSign']

**#getting age of certificate**

startingDate = str(certificate['notBefore'])

endingDate = str(certificate['notAfter'])

startingYear = int(startingDate.split()[3])

endingYear = int(endingDate.split()[3])

Age\_of\_certificate = endingYear-startingYear

**#checking final conditions**

if((usehttps==1) and (certificate\_Auth in trusted\_Auth) and (Age\_of\_certificate>=1) ):

return -1 **#legitimate**

elif((usehttps==1) and (certificate\_Auth not in trusted\_Auth)):

return 0 #suspicious

else:

return 1 **#phishing**

except Exception as e:

return 1

def domain\_registration(url):

try:

w = whois.whois(url)

updated = w.updated\_date

exp = w.expiration\_date

length = (exp[0]-updated[0]).days

if(length<=365):

return 1

else:

return -1

except:

return 0

def favicon(url):

#ongoing

return 0

def port(url):

#ongoing

return 0

def https\_token(url):

subDomain, domain, suffix = extract(url)

host =subDomain +'.' + domain + '.' + suffix

if(host.count('https')): **#attacker can trick by putting https in domain part**

return 1

else:

return -1

def request\_url(url):

try:

subDomain, domain, suffix = extract(url)

websiteDomain = domain

opener = urllib.request.urlopen(url).read()

soup = BeautifulSoup(opener, 'lxml')

imgs = soup.findAll('img', src=True)

total = len(imgs)

linked\_to\_same = 0

avg =0

for image in imgs:

subDomain, domain, suffix = extract(image['src'])

imageDomain = domain

if(websiteDomain==imageDomain or imageDomain==''):

linked\_to\_same = linked\_to\_same + 1

vids = soup.findAll('video', src=True)

total = total + len(vids)

for video in vids:

subDomain, domain, suffix = extract(video['src'])

vidDomain = domain

if(websiteDomain==vidDomain or vidDomain==''):

linked\_to\_same = linked\_to\_same + 1

linked\_outside = total-linked\_to\_same

if(total!=0):

avg = linked\_outside/total

if(avg<0.22):

return -1

elif(0.22<=avg<=0.61):

return 0

else:

return 1

except:

return 0

def url\_of\_anchor(url):

try:

subDomain, domain, suffix = extract(url)

websiteDomain = domain

opener = urllib.request.urlopen(url).read()

soup = BeautifulSoup(opener, 'lxml')

anchors = soup.findAll('a', href=True)

total = len(anchors)

linked\_to\_same = 0

avg = 0

for anchor in anchors:

subDomain, domain, suffix = extract(anchor['href'])

anchorDomain = domain

if(websiteDomain==anchorDomain or anchorDomain==''):

linked\_to\_same = linked\_to\_same + 1

linked\_outside = total-linked\_to\_same

if(total!=0):

avg = linked\_outside/total

if(avg<0.31):

return -1

elif(0.31<=avg<=0.67):

return 0

else:

return 1

except:

return 0

def Links\_in\_tags(url):

try:

opener = urllib.request.urlopen(url).read()

soup = BeautifulSoup(opener, 'lxml')

no\_of\_meta =0

no\_of\_link =0

no\_of\_script =0

anchors=0

avg =0

for meta in soup.find\_all('meta'):

no\_of\_meta = no\_of\_meta+1

for link in soup.find\_all('link'):

no\_of\_link = no\_of\_link +1

for script in soup.find\_all('script'):

no\_of\_script = no\_of\_script+1

for anchor in soup.find\_all('a'):

anchors = anchors+1

total = no\_of\_meta + no\_of\_link + no\_of\_script+anchors

tags = no\_of\_meta + no\_of\_link + no\_of\_script

if(total!=0):

avg = tags/total

if(avg<0.25):

return -1

elif(0.25<=avg<=0.81):

return 0

else:

return 1

except:

return 0

def sfh(url):

#ongoing

return 0

def email\_submit(url):

try:

opener = urllib.request.urlopen(url).read()

soup = BeautifulSoup(opener, 'lxml')

if(soup.find('mailto:')):

return 1

else:

return -1

except:

return 0

def abnormal\_url(url):

#ongoing

return 0

def redirect(url):

#ongoing

return 0

def on\_mouseover(url):

#ongoing

return 0

def rightClick(url):

#ongoing

return 0

def popup(url):

#ongoing

return 0

def iframe(url):

#ongoing

return 0

def age\_of\_domain(url):

try:

w = whois.whois(url)

start\_date = w.creation\_date

current\_date = datetime.datetime.now()

age =(current\_date-start\_date[0]).days

if(age>=180):

return -1

else:

return 1

except Exception as e:

print(e)

return 0

def dns(url):

#ongoing

return 0

def web\_traffic(url):

#ongoing

return 0

def page\_rank(url):

#ongoing

return 0

def google\_index(url):

#ongoing

return 0

def links\_pointing(url):

#ongoing

return 0

def statistical (url) :

#ongoing

return 0

def main (url) :

check = [[url\_having\_ip(url),url\_length(url),url\_short(url),having\_at\_symbol(url),

doubleSlash(url),prefix\_suffix(url),sub\_domain(url),SSLfinal\_State(url),

domain\_registration(url),favicon(url),port(url),https\_token(url),request\_url(url),

url\_of\_anchor(url),Links\_in\_tags(url),sfh(url),email\_submit(url),abnormal\_url(url),

redirect(url),on\_mouseover(url),rightClick(url),popup(url),iframe(url),

age\_of\_domain(url),dns(url),web\_traffic(url),page\_rank(url),google\_index(url),

links\_pointing(url),statistical(url)]]

print(check)

return check

**5.Output**

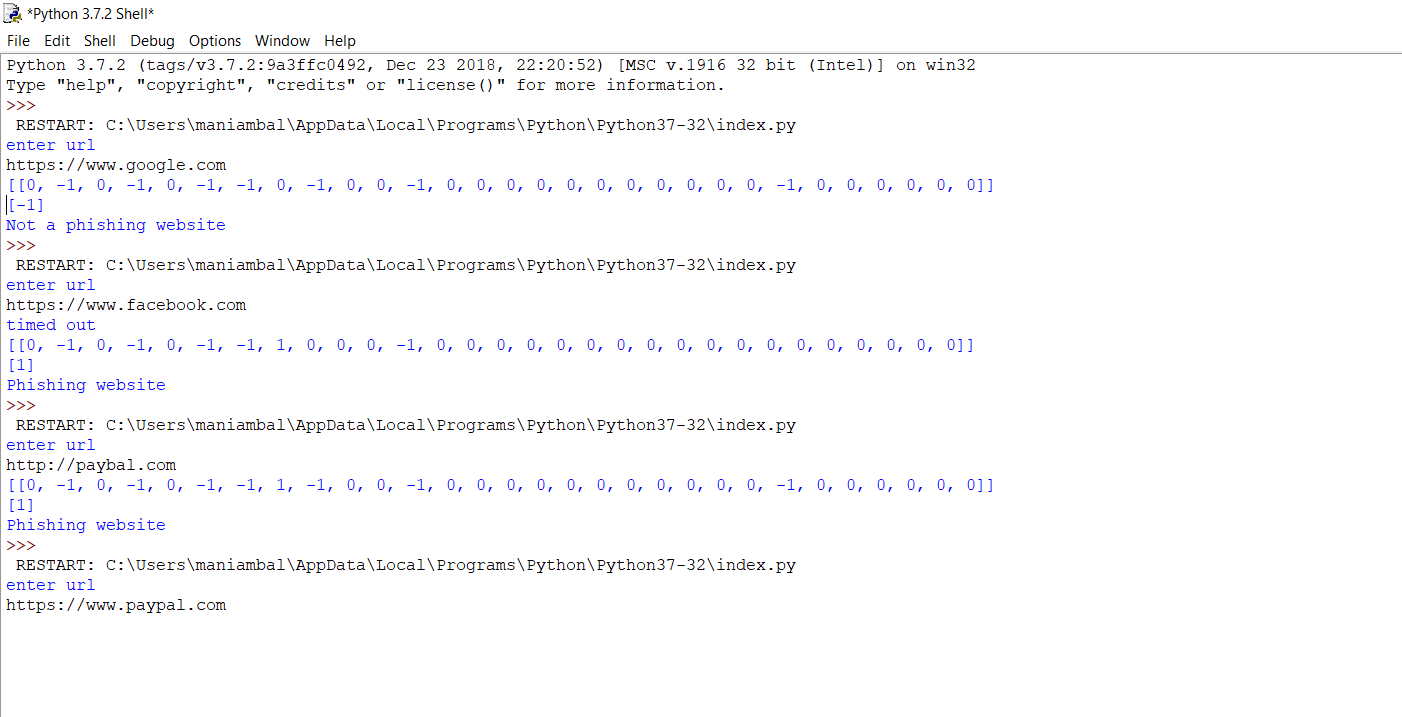


Fig 5.1 Output

**6.Accuracy**

**( i ) Load the Data set**

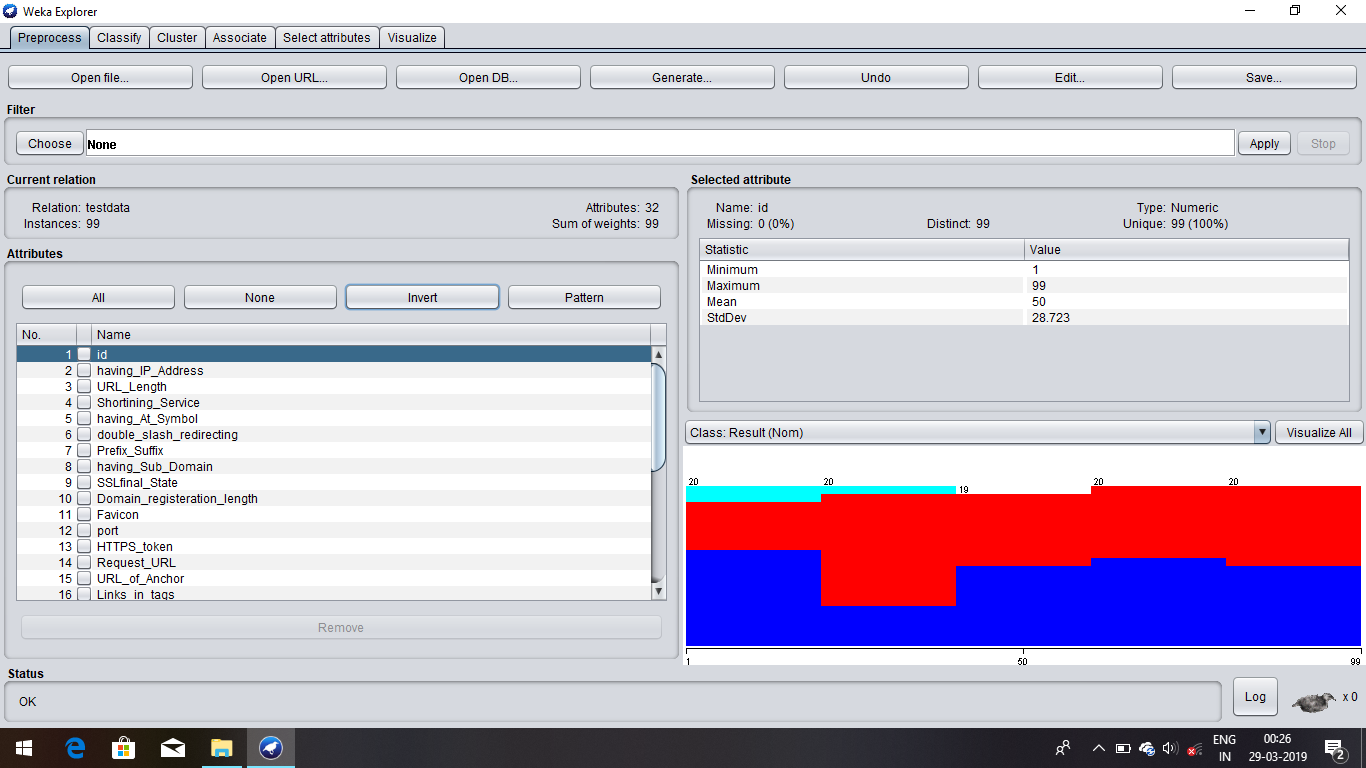


Fig 6.1 Loading the data set into WEKA tool

**( ii ) Naïve Bayes**

87.8788% of web page phishing data instances are correctly classified and remaining 12.1212% of instances are incorrectly classified. The percentage of correctly classified instances is often called accuracy or sample accuracy.

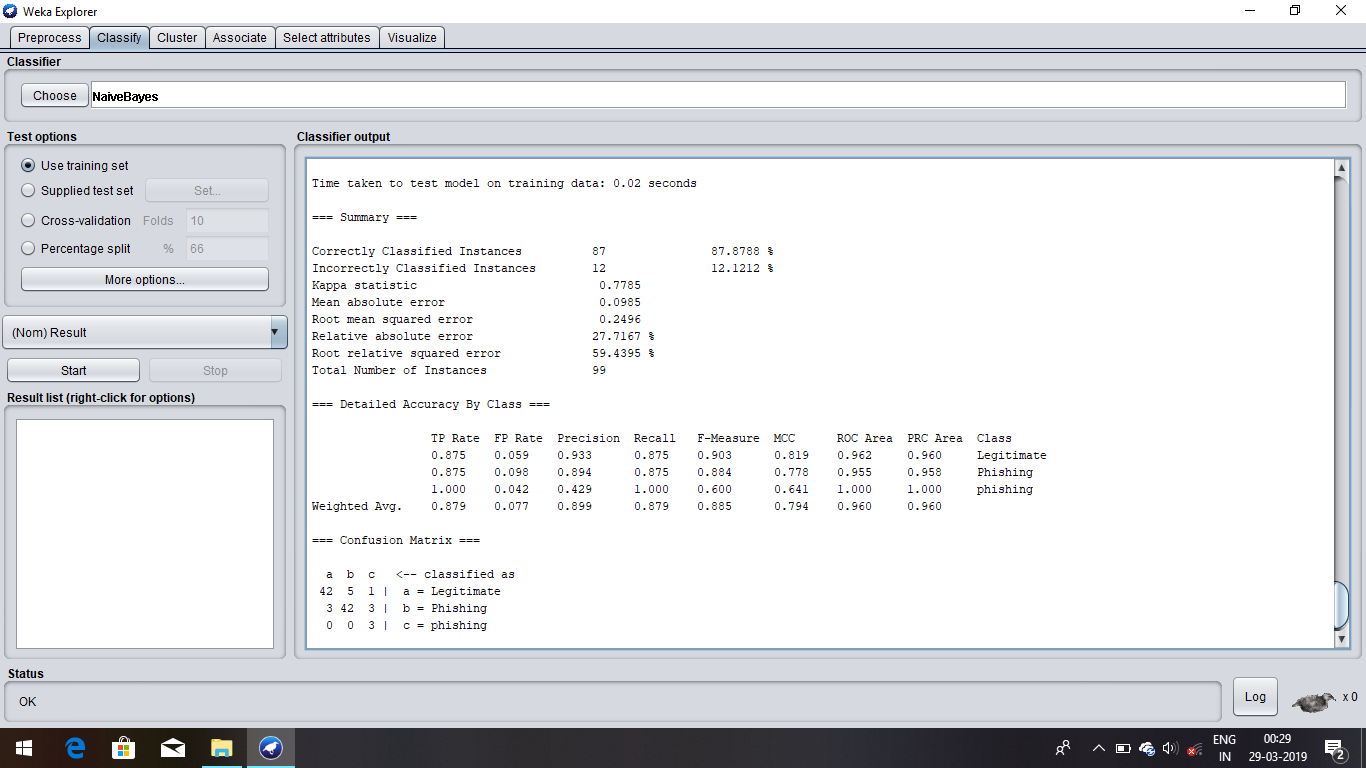


Fig 6.2 Naïve Bayes accuracy

**( iii ) Random forest**

99.89% of web page phishing data instances are correctly classified and remaining is 11% of instances are incorrectly classified. The percentage of correctly classified instances is often called accuracy or sample accuracy. So this data set consists of 99.89% accurate instances.

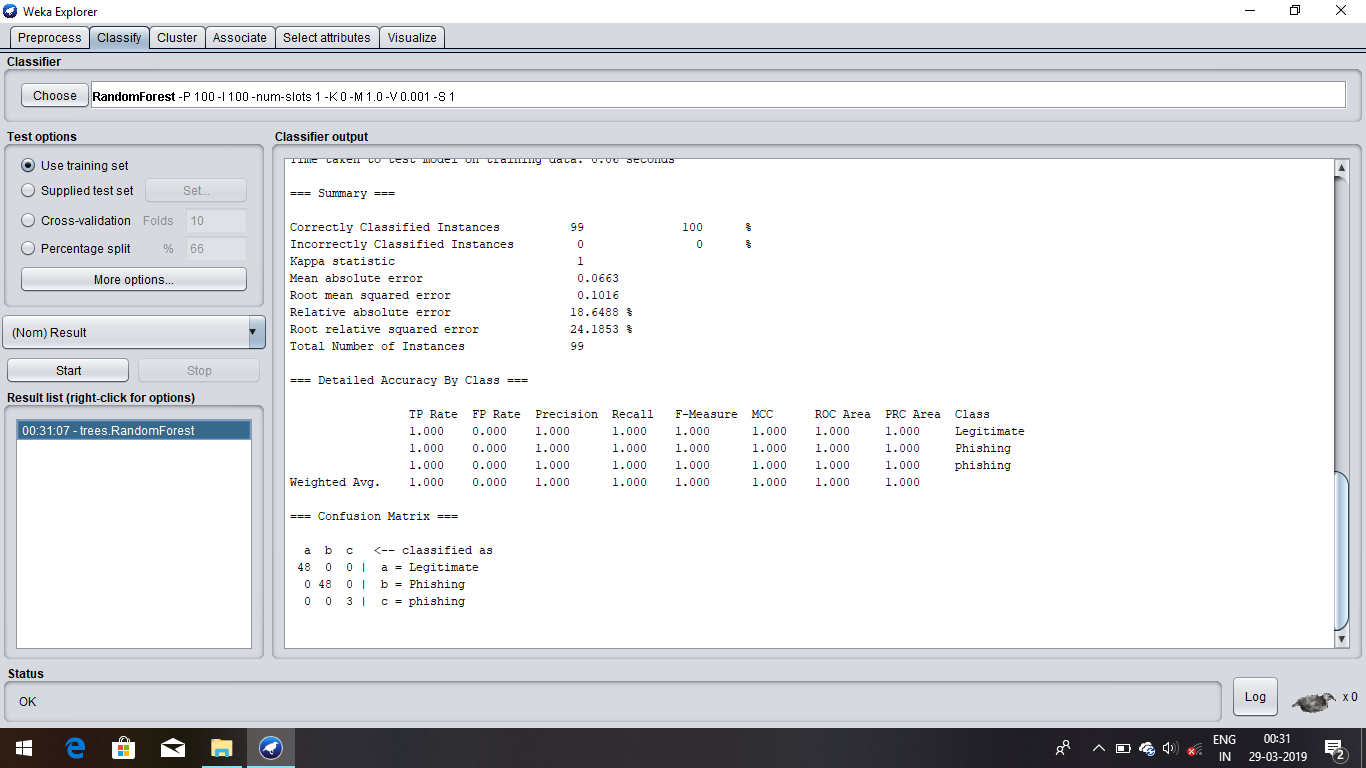


Fig 6.3 Random Forest accuracy

**( iv ) Logistic Regression**

97.2592% of web page phishing data instances are correctly classified and remaining 2.7408% of instances are incorrectly classified. The percentage of correctly classified instances is often called accuracy or sample accuracy. So this data set consists of 97.18% accurate instances.

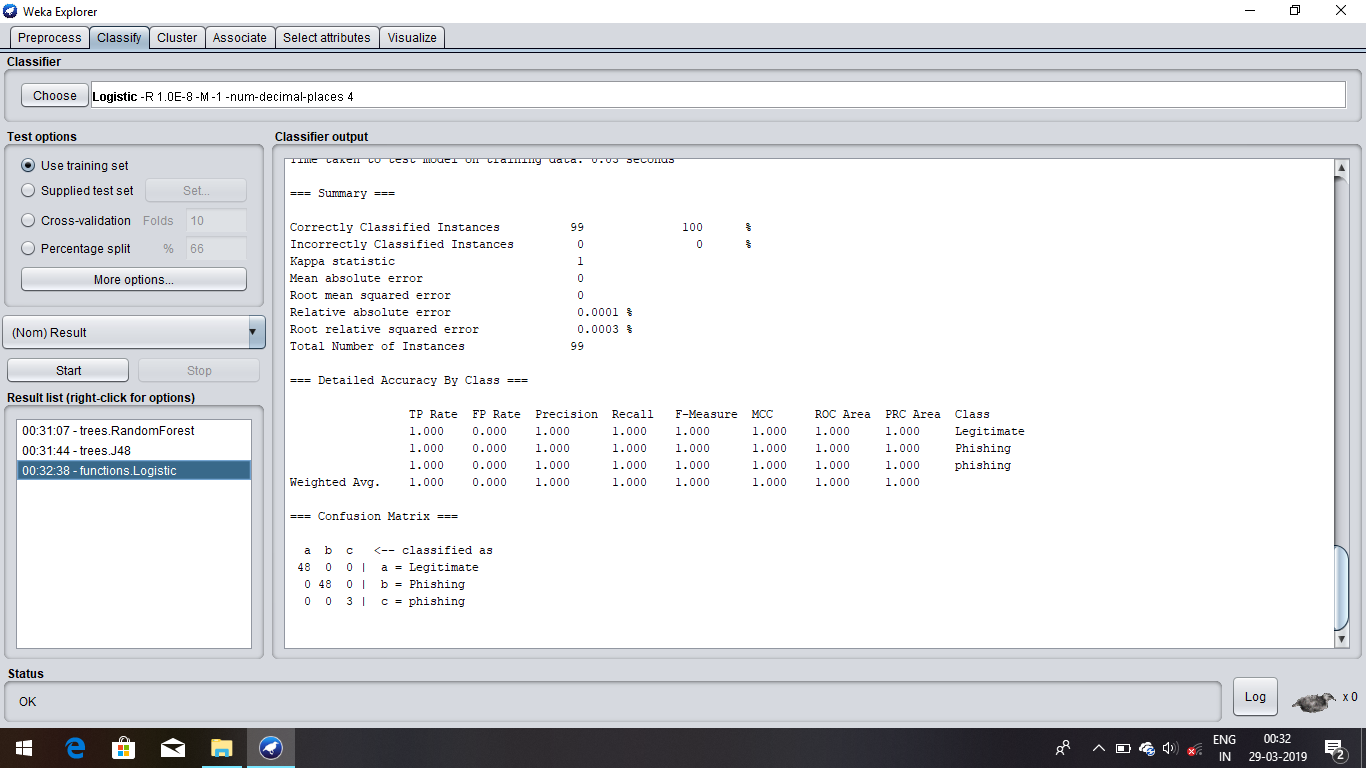


Fig 6.4 Logistic accuracy

**( v ) j48**

90.9091% of web page phishing data instances are correctly classified and remaining 9.0909% of instances are incorrectly classified. The percentage of correctly classified instances is often called accuracy or sample accuracy. So this data set consists of 90.9% accurate instances.

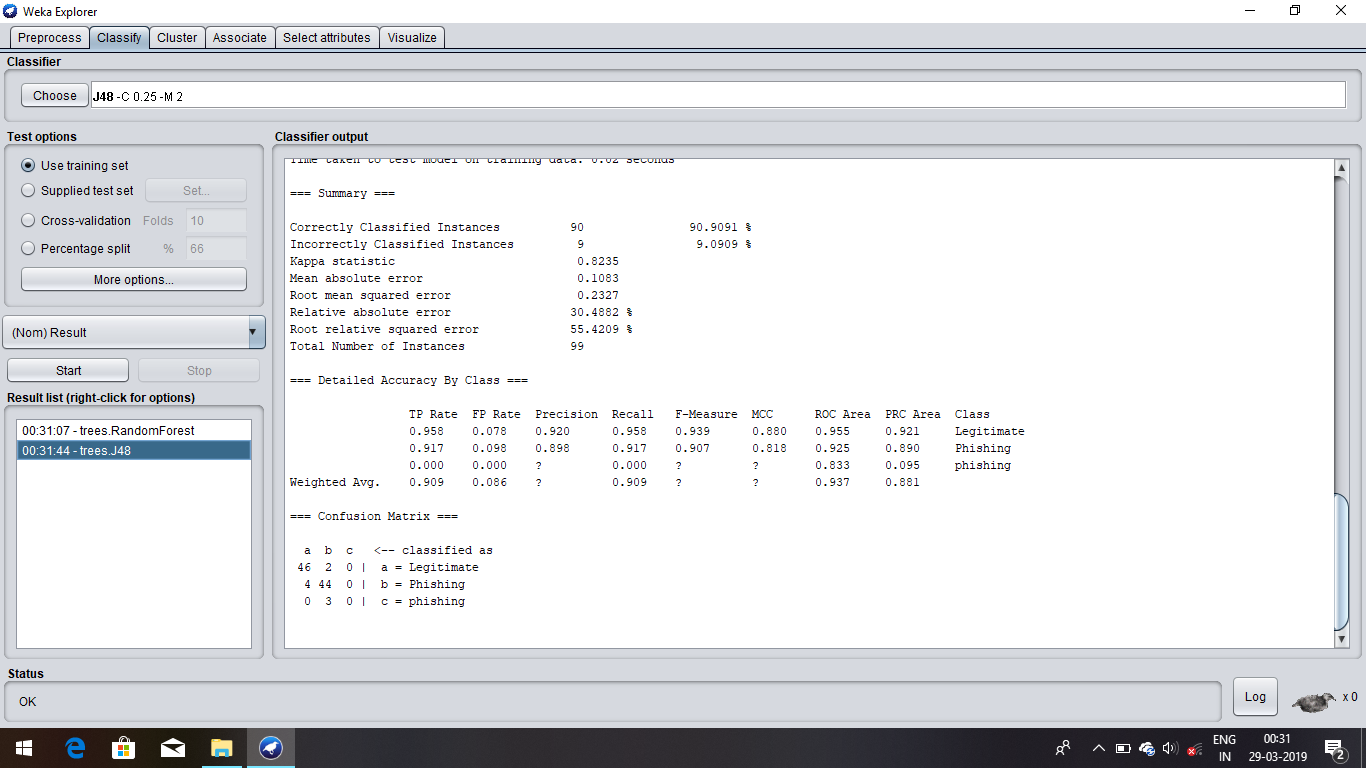


Fig 6.5 j48 accuracy

**7.Results**

Table7.1 accuracy

|  |  |
| --- | --- |
| **Classification algorithm** | **Correctly Classified Instances** |
| Naïve Bayes | 87.87% |
| Logistic Regression | 97.56% |
| Random Forest | 99.89% |
| J48 | 90.90% |

Fig 7.2 Accuracy of classification algorithm

**Chapter 8**

**Conclusion and Future Work**

**Conclusion**

Phishing is the one of the most serious threats in the cyber world. In this attack, the user inputs credential to a bogus website which looks like a genuine one. In this work, presented a survey on phishing website detection based on Classification algorithm. These approaches use various features of a web page to detect phishing attacks, such as text, URL, website traffic, login form, certificate, etc. Some of these approaches still have limitations in terms of accuracy, embedded objects, zero hours attack, etc. Detection of phishing site which uses the embedded object is still an open challenge. Moreover, machine learning based techniques require a high computational power to extract and compute the features in a real-time environment.

**Future work**

Future work will aim to develop a system that can learn by itself about new types of phishing attacks by adding a more enhanced feature to the detection process.

**9.References**

[1] G. Canbek and ù. Carolus, “A Review on Information, Information Security and Security Processes,” Politek. Derg., vol. 9, no. 3, pp. 165– 174, 2006.

[2] L. McCluskey, F. Thabtah, and R. M. Mohammad, “Intelligent rulebased phishing websites classification,” IET Inf. Secur., vol. 8, no. 3, pp. 153–160, 2014.

[3] R. M. Mohammad, F. Thabtah, and L. McCluskey, “Predicting phishing websites based on self-structuring neural network,” Neural Comput. Appl., vol. 25, no. 2, pp. 443–458, 2014.

[4] R. M. Mohammad, F. Thabtah, and L. McCluskey, “An assessment of features related to phishing websites using an automated technique,” Internet Technology, pp. 492–497, 2012.

[5] W. Hadi, F. Aburub, and S. Alhawari, “A new fast associative classification algorithm for detecting phishing websites,” Appl. Soft Comput. J., vol. 48, pp. 729–734, 2016.

[6] N. Abdelhamid, “Multi-label rules for phishing classification,” Appl. Comput. Informatics, vol. 11, no. 1, pp. 29–46, 2017.

[7] N. Sanglerdsinlapachai and A. Rungsawang, “Using domain top-page similarity feature in machine learning-based web phishing detection,” in 3rd International Conference on Knowledge Discovery and Data Mining, WKDD 2010, 2010, pp. 187–190.