S	
E1	
E2	
For Office Use Only	

Masters Project Final Report

(MCS)

March 2018

Project Title	**Enhanced Model To Detect Phishing Web Sites Based On Fuzzy Logic And Heuristic Approach**
Student Name	**E. W. M. Nimalika Ekanayake**
Registration No. & Index No.	**2015/MCS/028** **15440284**
Supervisor's Name	**Dr Kasun de Zoysa**

For Office Use ONLY

Enhanced Model To Detect Phishing Web Sites Based On Fuzzy Logic And Heuristic Approach

A dissertation submitted for the Degree of Master of Computer Science

E. W. M. Nimalika Ekanayake
University of Colombo School of Computing
2018

Declaration

The thesis is my original work and has not been submitted previously for a degree at this or any other university/institute.

To the best of my knowledge it does not contain any material published or written by another person, except as acknowledged in the text.

Student Name: **E. W. M. Nimalika Ekanayake**

Registration Number: **2015/MCS/028**

Index Number: **15440284**

_____

Signature: Date:

This is to certify that this thesis is based on the work of

Ms. **E. W. M. Nimalika Ekanayake**

under my supervision. The thesis has been prepared according to the format stipulated and is of acceptable standard.

Certified by:

Supervisor Name: **Dr Kasun de Zoysa**

_____

Signature: Date:

Abstract

With the rapid growth of Internet usage and online transactions web attacks also have increased during recent years. Among them phishing attacks are common and the damage is higher. Phishing is the attempt to steal sensitive information like usernames, passwords, credit card numbers etc from users. Most of the time this is done by deceiving the users by making website similar to legitimate sites. Users are unable to identify the difference. Therefore they use their confidential information in these sites. So phishers collect these data and misuses them. According to Anti Phishing Working Group, compared to 2015 there is 65% increase in number of phishing attacks in 2016. Researches have proposed different solutions to this problem. But none of these have been able to solve the issue completely due to the dynamic and complex nature of the problem. Therefore a better solution to fight these phishing attacks is an urgent need nowadays.

In this research we proposed and developed an enhanced model to detect phishing websites based on fuzzy logic and heuristic approach. 10 features have been identified which distinguishes a phishing website from a legitimate website depending on the URL of the website. The values for these features are the input to the model. It is a fuzzy model which is capable of deciding whether a given URL is a phishing website or a legitimate website. To decide this, the model uses fuzzy rules, which are derived, based on data mining classification process. The model was developed using Java programming language together with fuzzy control language.

The final model was evaluated based on confusion matrix and the model was able to show 82.56 % overall accuracy rate with higher true positive rate. A chrome browser extension was developed and it is able to detect phishing sites in real time. This helps the users to protect their sensitive information from phishers.

Acknowledgements

First and foremost I would like to express my sincere gratitude to my project supervisor Dr Kasun De Zoysa who has supported and guided me throughout the research. His advices and suggestions helped me to complete my research successfully.

Also I would like to thank my colleagues and friends who helped me in numerous ways from the beginning of the research till the end.

My sincere gratitude offers to my office staff including the country manager who supported me during the entire year to balance office work and academic work. And I would like to thank project coordinators, academic staff and the library staff who provided me with necessary information whenever I needed.

Finally I would like to express my gratitude to my family who supported me and encouraged me in completing the research. Without them I would not have been able to achieve my objectives.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

DNS	Domain Name System
APWG	Anti-Phishing Working Group
URL	Uniform Resource Locator
IP	Internet Protocol
SVM	Support Vector Machine
KNN	K-Nearest Neighbor
HTTP	Hypertext Transfer Protocol
SSL	Secure Sockets Layer
IDCP	Intelligent Detection For Cyber Phishing
WEKA	Waikato Environment for Knowledge Analysis
API	Application Programming Interface
FCL	Fuzzy control language
IDE	Integrated Development Environment
LM	Left Most Max
FIS	Fuzzy Inference System
TP	True Positive
TN	True Negative
FP	False Positive
FN	False Negative
ACC	Accuracy
ARFF	Attribute-Relation File Format

Chapter 1

Introduction

1.1 Motivation

Use of Internet is increasing rapidly. People tend to use online services due to fast and easy access. Internet has become a useful part of day-to-day life. Most of social and financial activities are now online. Almost every person depends on online activities like online banking, online shopping, online booking and many more. Due to heavy use of Internet, effect of web attacks also increasing. Among these, phishing attacks plays a major role. Also phishing web sites can cause the loss of thousands of dollars and leads to the damage of the brand image of organizations. Phishing is a cyber-security threat that steals personal information from users by posing as a trustworthy organization or entity. This is a type of social engineering attack that loots confidential information such as username, credit card details, passwords, account details, social security number and other important data. In these attacks attacker makes fake pages. They design the pages by making a little change in the legitimate page or copying it. Therefore the Internet user will not be able to differentiate between legitimate and phishing webpages. There are different types of phishing attacks. Data Theft, Search Engine Phishing, DNS-Based Phishing etc. [1].

According to the report released by APWG (Anti Phishing Working Group) the total number of phishing attacks in 2016 were 1,220,523 and it is a 65% increase over 2015. APWS has stated that in 4th quarter of 2004, phishing attacks recorded per month is 1609. But in 2016 an average of 92,564 per month is recorded. It's an increase of 5753% over 12 years [2]. These statistics clearly shows that there's a rapid growth in the phishing attacks.

Detecting and identifying phishing websites in real-time is a complex and dynamic problem. Although there are various phishing detection techniques that have been proposed to prevent phishing, there is still a lack of accuracy due to the dynamic nature. So it is really important to develop a phishing detection model with higher accuracy rate and fast access time to overcome this problem.

1.2 Statement of the problem

Nowadays Internet has become an essential part of our daily lives. We are engaging with online banking, online shopping and many more through Internet. With the recent growth of Internet, web attacks also have increased rapidly. Among them phishing attacks have affected users a lot. There's no single solution to stop this fraudulent activity. According to the reports released by Anti Phishing Working Group statistics clearly shows that there's a rapid increase of phishing attacks over past years. Damage caused by these phishing attacks also increasing. Those attacks have caused huge financial loses and therefore affected the world economy. People have lost the trust of online transactions due to these reasons. In response to the increase in phishing attacks, different phishing detection techniques have been the focus of considerable research. Researchers are proposing different solutions to this problem. Due to the complexity and dynamic nature of this problem still there's no proper solution. Therefore, effective way to detect phishing attacks is really important nowadays.

1.3 Aims and Objectives

The aim of this project is to develop an enhanced model to detect phishing web sites based on fuzzy logic and heuristic approach.

To achieve the aim, one of the main objectives is to study the existing phishing attack detection systems. Researchers have proposed different solutions for detecting phishing web sites. But these solutions have limitations. Therefore it is important to identify these limitations to produce a better solution.

Next objective is to develop a new model, which addresses the limitations of the current approaches. Producing a model with higher accuracy rate and a fast classification time is the main focus.

Then the evaluation of the developed model needs to be done to ensure the effectiveness and accuracy of the results produced by the system.

Finally, develop a sample web browser extension that uses the developed model to alert the user about suspected the phishing websites.

1.4 Scope

This project will consist of developing an enhanced model to detect phishing web sites based on fuzzy logic and heuristic approach. This project will use both URL analysis and web site ranking to detect phishing web sites. URL analysis is based on the identified URL features like length, IP address, domain, subdomain, URL characters etc. Alexa web page ranking and WHOIS data analysis also will be used in this approach. Also a fuzzy system will be used for the prediction process.

The required data to develop and evaluate the model will be collected from PhishTank website and Yahoo directory. PhishTank is an online repository, which contains information about phishing web sites [3]. This get updated daily. Yahoo directory provides large data sets of legitimate web sites for researches [4]. Output of the project is an enhanced model which is capable of identifying whether a given website is a phishing web site or a legitimate website. Main focus of the model is to obtain a higher accuracy rate and produce the classification result within an acceptable time.

Evaluation process will be done based on the updated phishing web sites list in PhishTank and legitimate web sites list in Yahoo directory. Sample web browser extension will also be developed to alert user about the phishing web sites in this project. Project will be completed by the end of the academic year

1.5 Contribution

Phishing attack detection is a dynamic and a complex problem. The reason for this is the new techniques and methodologies used by the phishers. And also most of the phishing web sites do not have a long lifetime. Some may be up and running only for few hours. Therefore phishing detection methods like blacklisting where, a list of suspicious web sites is recorded may not be appropriate. Due to the rapid increase in phishing attacks and the monetary loses causing to the world economy, many anti phishing techniques have been developed and many researches have proposed solutions. But still there is no proper methodology to detect phishing web sites. Some solutions lack the ability to maintain high accuracy rate. Some may have the problem of detecting the website in real time. Some solutions have the limitation of classifying within acceptable time duration. The proposed solution focus on both URL analysis and web page ranking methodologies together with fuzzy logic to detect the phishing web sites with a higher accuracy rate and with real time fast classification.

Chapter 2

Literature Review

Phishing attack detection is a major focus these days due to the increase in the attacks during past few years. APWG statistics also shows that phishing attacks have been increased rapidly. Many researchers have proposed different solutions to address this problem. Those can be categorized as Black List/White List based approaches, Heuristic based approaches, Fuzzy logic based approaches and Fuzzy logic and classification based approaches. But still none of them were able to fix the issue completely due to the dynamic nature of the problem and the complexity. Different proposed solutions have different limitations. Major limitations are low accuracy rate and slow classification in real environment.

2.1 Black List / White List based approach

List based approach is the easiest and fastest approach to detect phishing web sites. This approach is basically maintaining two lists. Black list contains phishing web sites and white list contains legitimate web sites. But this approach may lack the property of higher accuracy rate. Researchers have proposed different solutions based on this approach. Ankit Kumar Jain and B. B. Gupta have proposed a method-using auto updated white list of legitimate web sites. Also their approach uses the hyperlink features that are extracted from the source code of a webpage to make the decision. The proposed model has 2 major modules, domain and IP address matching module that matches the present domain and IP address in the white list and hyperlink module which examines the features extracted from hyperlinks to take the decision. In domain and IP address matching module they maintain a whitelist, which contains domain name and corresponding IP address. In hyperlink module it checks the links in phishing web page, which has links to legitimate web pages. To verify the hyperlink relationship they have used 1120 web pages taken from Phishtank web site. Second module will be used if the web page that user tries to access does not present in the whitelist. This increases the access time. [5].

List based approaches are easy to implement. Also it is easy to use. But the accuracy rate may

be low. Most of the phishing web sites are hosted only for few hours. Therefore maintaining a huge blacklist is very hard. It will require more memory usage.

2.2 Heuristic Based Approach

Blacklist based phishing detection techniques are not effective. Therefore researchers focus on more advanced techniques. Jin-Lee Lee et al. proposed a heuristic based phishing detection technique. It uses uniform locator (URL) features. They have used 3000 phishing site URLS and 3000 legitimate site URLS in developing the model. The main focus was the URL features. To develop the proposed model they have used 26 URL based features that were used in previous studies as well as new features. Some of them include Google page rank, IP address, Length of URL, Suspicious character, Port number matching, Number of subdomains, Length of URL, Prefix and suffix, Primary domain spelling mistake and Phishing words in URL. Proposed approach includes two phases training and detection. In the training phase a classifier is generated. They have used several machine learning algorithms to determine the classifier. Those are support vector machine (SVM), Naive Bayes, Decision tree, k-nearest neighbour (KNN), Random tree and Random forest. The results were compared and they have identified Random forest as the best machine learning algorithm to identify phishing web sites based on URL features. Since they have use many URL features the classifier generation and detection process was slower than expected [6].

There are several features that are helpful to clearly distinguish a phishing website from a legitimate one. Some of these features are URL and domain identity, Page style and contents, Security and encryption, Source code and JavaScript and Web address bar. Researchers use different number of features and different combinations in their models. Jaydeep Solanki and Rupesh G. Vaishnav has used ‘URL & domain name’ and ‘host based’ in their model. They check URL features like the length of the URL, number of dots and slashes in the url, special characters, HTTP and SSL, IP address etc. After feature extraction they have used a machine-learning algorithm as the classifier. Support Vector Machines and Decision tree algorithms are used to develop this model [7].

Heuristic based approaches are more complex to implement than list based approaches. But these approaches can be used to detect phishing web sites that may not present when the model was developed. Since heuristic based use URL features, the accuracy depends on the features selected. If more features are selected then the computation time will be higher. If

less features are selected accuracy may not be up to required state. Also accuracy depends on the algorithm used for the classification process.

2.3 Fuzzy Logic Based Approach

Phishing attacks harm the victims by plundering the identity and money. Due to this people lose their trust in Internet transactions. Detecting phishing is complex because every time phishers approach with new technologies and methodologies. Therefore people propose different solutions. One approach is based on fuzzy logic. Fuzzy logic has been using from many decades for researches to embed the inputs into computer model for many applications. In Boolean logic input is represented as true or false. But in fuzzy logic it allows representation of partial membership in sets to calculate result. The importance of fuzzy logic in phishing detection is the use of the linguistic variables to represent the phishing indicators.

K. N. Manoj Kumar and K. Alekhya proposed a solution based on fuzzy logic to deal with fraudulent websites. It is implemented through fuzzy logic technique. This approach categorizes the given URL input of the website into five categories. They are highly legitimate, legitimate, suspicious, phishy and highly phishy. Phishing attack detection model uses 4 main phases Fuzzification, Evaluating rule, aggregating the rule outputs and Defuzzification. After going through this 4 main phases this model classifies a given URL input to one of the phishing categories [8].

P. Barraclough and G. Fehringer also proposed a model based on fuzzy logic. They have constructed a fuzzy rule model utilizing combined features based on a fuzzy inference system. They proposed intelligent detection for cyber phishing (IDCP) model classifies between phishing, suspicious and legitimate characters. IDCP consists of four main components. They are feature base, feature sources, current websites and Inference engine. Feature base contains phishing features. The feature sources are phishing websites from PhishTank archive where phishing websites are maintained for public access. Current website carry 3 possible characteristics phishing, suspicious and legitimate characteristics. Fuzzy Inference Engine is a Sugeno systematic approach that carries out the reasoning in which the detection system searches for a solution [9].

Fuzzy logic based approaches have the advantage of low memory consumption and also inference speed is high. But the implementation is more complex compared to heuristic based approach.

2.4 Fuzzy Logic and Classification Based Approach

Fuzzy logic based approach was unable to provide the expected accuracy rate. But this has strengths like low memory consumption and higher inference speed. Therefore researchers started focusing on using fuzzy logic together with classification. Data mining techniques play a major role in this approach.

To discriminate between legitimate and phishing URLs, fuzzy and binary matrix construction method can be used. S. Nivedha et al. proposed a fuzzy based logic association rule-mining algorithm. In this approach they have used different URL features. They have used fourteen features such as dots in host name of the URL, IP address, top-level domain, sub domain URL length, Unicode in URL and special characters. Then these extracted features are converted to fuzzy membership values as 'Low', 'Medium' and 'High'. Rules were generated to detect phishing web sites by applying association rule. Apriori is the algorithm used to generate binary matrix method. In this approach they have used only URL based features [10].

Automatic filtering of phishing websites has become a necessity as these can cause the loss of thousands of dollars and leads to brand image damage. Shireen Riaty et al. proposed a model to Enhance detecting phishing websites based on machine learning techniques of fuzzy logic with associative rules. Based on rules the proposed phishing detection model converts input features of the web sites into an output that reveals the nature of the web site whether it's a phishing site. A new set of features is constructing from the input data. These features are transferred into different forms of continuous values using clustering, value mapping process and frequent pattern mining. Prediction process happens when newly constructed features are used with a fuzzy system that is learned by optimizing membership function and a set of rules. The proposed work discovers patterns of the phishing websites properties using association rules that inserted into a fuzzy logic classifier. A fuzzy system is commonly applied to control variables of continuous nature. Therefore pre-processing the input features prior to applying fuzzy system has been done. But changing the shape and the nature of the

input feature will influence the accuracy of the phishing detection technique positively or negatively [11].

Sathish .S and Thirunavukarasu .A proposed a Phishing website detection model for secure online transactions. The proposed model has three main layers. First layer used Google page rank and IP address in the URL. In the second layer domain name characteristics are used. Characteristics of web page content are used in layer three. Fuzzy logic is used to classify the web pages according to their rank. Development has been done in three stages. First is the data collection. Second stage is fuzzy rule base. Final stage is classification. Classification tool WEKA is used for the classification of web sites using the determined parameters. Fuzzy inference system is employed to identify and report the end user of the risk factors [12].

Phishing detection is a complex and dynamic problem because of subjective considerations and the ambiguity involved in detection. Fuzzy data mining is an effective tool to handle this problem since it offers more natural way of dealing with quality factors. Maher Aburrous et al. have proposed an intelligent phishing detection system for e-banking using fuzzy data mining. They have mainly focused on e-banking and identified 27 characteristics which stamp the forged website. Fuzzy is used to represent key phishing characteristics indicators in a linguistic manner. This system has four main steps. First is fuzzification where linguistic descriptors such as high, low, medium are assigned for each key phishing characteristic indicator. Next is the Rule generation using classification algorithms. In this step they have used data mining classification and association rule. Then Aggregation of the rule outputs happens. This is where unifying the outputs of all discovered rules is done. Final step is the defuzzification. This is the process of transforming a fuzzy output of a fuzzy inference system into a crisp output. This step was done using Centroid technique. The output is e-banking phishing website risk rate and is defined in fuzzy sets like ‘very phishy’ to ‘Very legitimate’[13].

The online banking consumers and payment service providers are the main targets of the phishing attacks. Therefore people focus more on detecting phishing websites. But it is a complex task, which requires significant expert knowledge and experience. Rajeev Kumar Shah et al. proposed an intelligent fuzzy-based classification system for e-banking phishing website detection. Providing protection to the users from phishers’ deception tricks, giving them the ability to detect the legitimacy of websites is the main aim of the proposed

solution. The proposed intelligent phishing detection system combines Fuzzy Logic model with association classification mining algorithms. This aggregates the capabilities of fuzzy reasoning in measuring imprecise and dynamic phishing features, with the capability to classify the phishing fuzzy rules. To develop the proposed model they have used 27 phishing website characteristics and factors. The model consists of four phases. They are fuzzification, rule evaluation, aggregation of the rule outputs and defuzzification [14].

Combining both fuzzy logic and classification rules helps to produce better quality solutions. But the development of the model is complex compared to other approaches. Accuracy depends on the quality of the selected feature and also the number of features. Comparison of strengths and limitations of discussed approaches is given in Table 2.1.

Table 2.1 Comparison of phishing detection approaches

Approach	Strengths	Limitations
Black List / White List based approach	• Easy to implement • Easy to use • Less computational cost	• Memory overhead • Low accuracy rate • Cannot detect zero hour phishing websites
Heuristic Based Approach	• High accuracy rate • Can detect zero hour phishing web sites	• Higher cost • Higher memory requirement
Fuzzy Logic Based Approach	• Can detect zero hour phishing web sites • Requires less memory • Inference speed is very high	• Complex to design • Accuracy is less • Accuracy depends on the selected features
Fuzzy Logic and Classification Based Approach	• Higher accuracy rate • Can detect zero hour phishing web sites • Inference speed is very high	• Complex to design • Accuracy depends on the selected features • Requires more domain specific knowledge

Currently available solutions have the problem of maintain high accuracy rate due to the dynamic nature of the phishing web sites. Some solutions have the limitation of detecting websites in real time. Detecting a given website whether a phishing website or a legitimate

website within an acceptable time is also a challenge. In the proposed model fuzzy logic and classification-based approach is used. Currently available approaches use URL features like length of URL, Suspicious characters, spelling mistakes in URL etc. In the proposed solution most appropriate selected URL features were used. Because using many features may slow down the classification process. Fuzzy logic is used because it has the power of representing the input variables in linguistic form. Because in phishing context it is not appropriate to define clear boundaries like true or false in URL features. Not only URL features, web site ranking and WHOIS data [15] also will be considered when developing the model. Current approaches use URL features. But using website ranking, WHOIS data together with URL analysis increases the accuracy. By studying the characteristics of phishing web site URL unique features can be identified to differentiate those with legitimate web sites. Therefore using URL features together with website ranking and WHOIS data helps to produce a better solution in detecting phishing websites.

Chapter 3

Analysis and Design

The proposed model is a phishing attack detection model based on fuzzy logic and heuristic approach. This model is capable of distinguishing phishing web sites from legitimate web sites. The Overall Architecture and the proposed methodology to develop phishing attack detection model is described in this chapter.

3.1 Overview of the methodology

The proposed phishing detection model uses fuzzy logic and data mining algorithms to detect phishing web sites based on identified web site characteristics and factors. The model uses 10 identified URL features such as length of URL, Alexa ranking [16], number of special characters presents etc. as the main input. Depending on the values of these features fuzzy model membership classes will be identified. The advantage of using fuzzy logic is that it's capability to define linguistic variables to represent Key Phishing characteristic indicators. Fuzzy representation more closely matches human condition. Also this approach enables processing of vaguely defined variables and the variables that cannot be defined by mathematical relationships. When developing the fuzzy model to identify the fuzzy rules data mining techniques are used. Data mining is used to extract implicit, previously unknown and potentially useful information from large data set. Data mining tools predict trends that can be used in detecting phishing web pages. The phishing detection model uses both fuzzy logic and data mining.

The propose methodology can be categorized into 5 major processes.

- Feature Extraction
- Fuzzification
- Rule generation using classification algorithms
- Aggregation of rules output
- Defuzzification

3.2 Feature Extraction

Extracting features from a given URL is one of the main processes in developing phishing detection model. These features are important to distinguish a phishing web site from a legitimate web site. The model is developed depending on these extracted features. Combination of these features helps to detect phishing web sites.

The model uses 10 identified features in the feature extraction process. These features are extracted from the literature for better analysis of phishing URLs.

1. WHOIS year of the web site domain
2. Alexa Rank of the web site
3. Length of the URL
4. Length of the host
5. Whether '@' sign present in the URL
6. Use of IP address in the domain
7. Number of dashes (-) in the URL
8. Number of underscores (_) in the URL
9. Number of dots (.) in the URL
10. Whether URL contains identified suspicious words

3.2.1 WHOIS year of the website domain

WHOIS databases store contact info for the owners of all registered domains. Name, address, phone number and email address are all listed. WHOIS also contains domain availability status, registration/expiration dates and related info. Using this database domain registration date is extracted.

3.2.2 Alexa Rank of the website

Alexa Traffic Rank is a measure of a website's popularity, compared with all of the other sites on the internet, taking into account both the number of visitors and the number of pages viewed on each visit. This can be interpreted as the website's position in a massive league table based on both visitor numbers and the number of pages viewed by each visitor. The 'most popular' site is given a rank of 1. Most of the phishing sites have very low alexa rank which indicates least popularity.

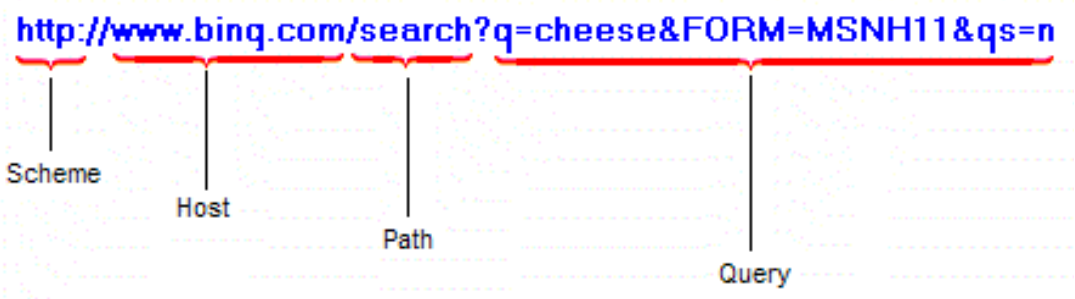
3.2.3 Length of the URL

This feature is the length of the URL. After doing the literature review it is understood that the length of the URL is a good feature to classify phishing and legitimate websites.

3.2.4 Length of the host

As previous studies suggests length of the host is also a good indicator to identify phishing and legitimate URLs. Parts of a URL is show in Figure 3.1

Figure 3.1 Parts of URL



3.2.5 Whether @ sign present in the URL

This feature is about presence of @ sign in the URL. An @ symbol in a URL causes the string to the left to be disregarded. Right side string is treated as the actual URL for retrieving the page. Because of the limited size of the browser address bar, this makes it possible to write URLs that appear legitimate within the address bar, but the real URL will retrieve a different page.

3.2.6 Use of IP address in the domain

If a page's domain name is an IP address there's a high probability that it being a phishing site. Legitimate web sites most of the time use a domain name instead of an IP address.

e.g. http://**23.251.146.223**/AZUL/Azul52276apc/index.php

3.2.7 Number of dashes (-) in the URL

This feature counts the number of dashes(-) present in the URL. Normally legitimate site do not have dashes present in the URL

e.g. http://info-fb-confirmation-2017.16mb.com/revery/ contains 3 dashes

3.2.8 Number of underscores (_) in the URL

This feature counts the number of underscores(_) present in the URL. Phishing site tend to use more than one underscores.

e.g. http://bestimpex.in/rs_plugin/index_1.html contains 2 underscores

3.2.9 Number of dots (.) in the URL

This feature counts the number of dots(.) present in the URL. Compared to legitimate sites phishing sites use many dots in the URL. To make links appear legitimate phishers use sub domains.

e.g. http://odyometre.org/aol.com/www.aol.com/ contains 4 dots

3.2.10 Whether URL contains identified suspicious words

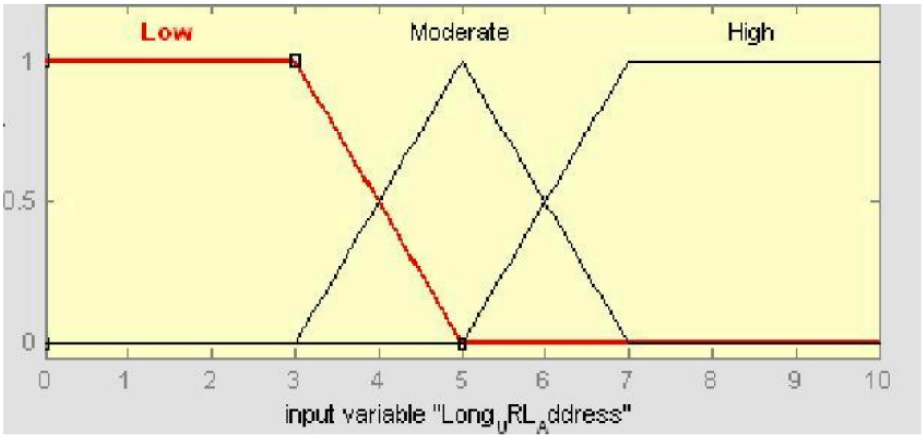
This feature checks whether a URL contains words like "secure", "account", "webscr", "login", "ebayisapi", "signin", "banking" and "confirm". Based on studies it is found that these words appear in phishing website frequently.

3.3 Fuzzification

First identifying the key phishing characteristics happens then in fuzzification step linguistic descriptions such as Low, Medium, High are assigned for each of these phishing characteristics. For example phishing characteristic indicator 'URL length' can range from input class 'Low' to 'High'. But the mapping is not one to one. We cannot specify clear boundaries between classes. Therefore membership function is designed to get the degree of membership. Membership function is a curve. It defines how each point in the input space is mapped to a membership value between 0 and 1. Triangular and trapezoidal membership functions are used. Similar to this for other phishing characteristic also linguistic descriptors are assigned. Example of membership functions for URL length feature is given in Figure 3.2 Output function is mapped to Phishing website risk rate. It is defined as phishing or legitimate.

Eg.

3.2 Input variable for Long URL Address component



3.4 Rule generation using classification algorithm

Next step is rule generation using classification algorithms. Firstly key phishing characteristic indicators and risk of the website are defined. Specifying how phishing website probability varies is the next step. In the proposed solution data mining classification and association rule approach is used instead of using expert system. Expert systems use expert knowledge to provide fuzzy rules. In the proposed solution the model automatically detect significant patterns of phishing characteristics in archive data. In phishing attack detection model two web access archives are used. One from Phishtank archive which contains a list of phishing website URLs and the other one from yahoo directory which contains a list of legitimate URLs. Using this data different feature sets are defined and many important rules are derived. This helped in fuzzy rule phase.

3.5 Aggregation of rules output

Next step is aggregation of the rule outputs. In this step output of all discovered rules are unified. In this step combining the membership functions of all rules into a single fuzzy set happens.

3.6 Defuzzification

Final step is defuzzification. In this step transforming the fuzzy output of the fuzzy inference system into a crisp output happens. Final output needs to be a crisp output. The input to the defuzzification process is the aggregate output fuzzy set. This is done using a defuzzifaction method. Developed phishing detection model's output is defined in fuzzy sets as 'phishing' or 'legitimate'. Finally fuzzy output set is defuzzified to arrive at a scalar value.

Overall architectural design is graphically described in Figure 3.3 It contains the main components of the proposed system and Figure 3.4 shows the flow in fuzzy inference sytem.

Figure 1.3 Main Components

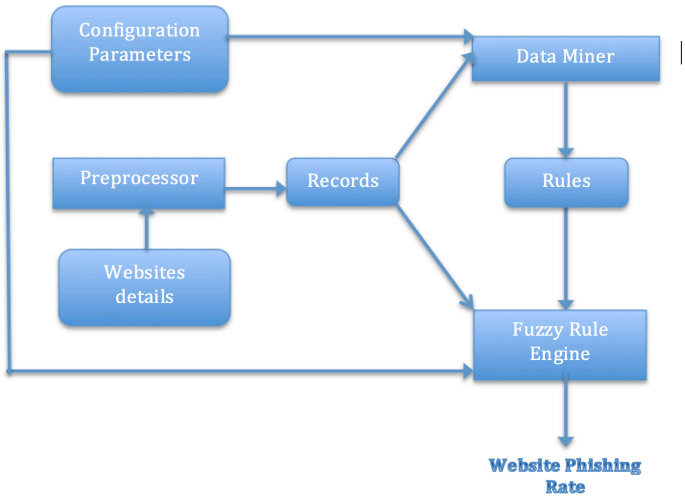
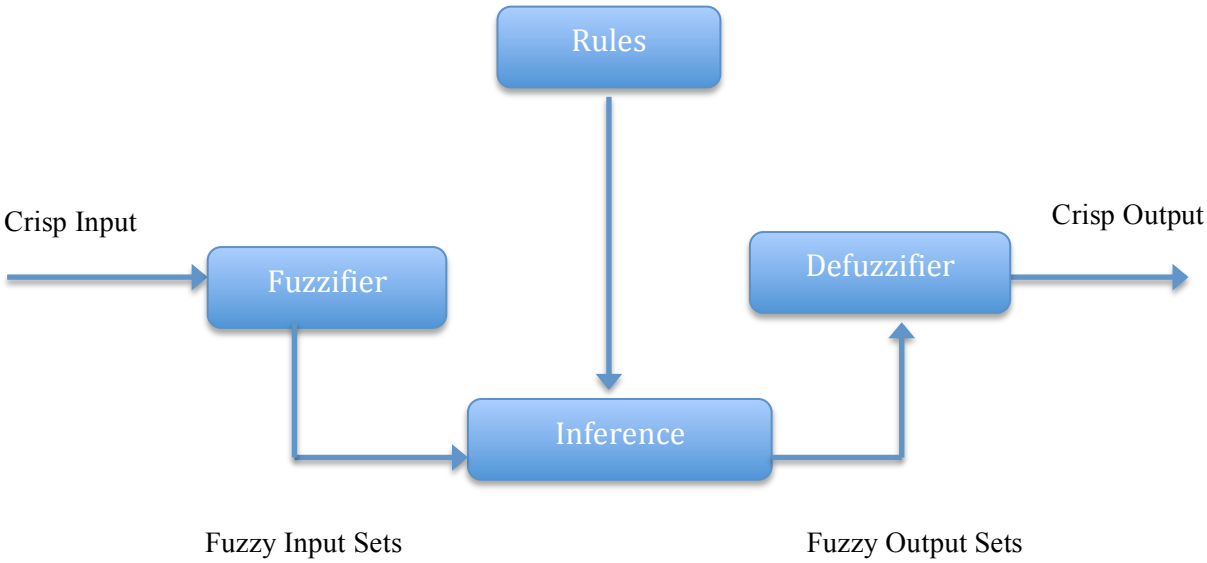


Figure 3.4 Fuzzy inference system



Chapter 4

Implementation

Implementation of phishing detection model has 6 main phases

1. Data collection
2. Feature Extraction
3. Defining membership functions for URL features
4. Fuzzy rule generation using heuristic approach
5. Implementing fuzzy model
6. Developing chrome browser extension to warn user about phishing sites

4.1 Data Collection

In this phase data is collected for training the classification model and to test the final fuzzy model. Data set contains two different sets of website URLs. First set contains phishing site URLs and the other set contains legitimate URLs. Phishing URLs are downloaded from well-known phishTank public data repository. It is a community driven database. When people submit URLs the validity is checked by the registered users to classify them under phishing sites. Legitimate URL set was collected from Yahoo directory. They maintain large list of legitimate URLs and they provide those data sets free of charge for University research students. From those two data sets 5000 phishing URLs and 5000 legitimate URLs were used in training phase. Another 5000 phishing URLs and 5000 legitimate URLs were used to test the final phishing detection model.

4.2 Feature Extraction

In feature extraction phase 10 features for each URL is identified. These methods are implemented using Java programming language using eclipse IDE. The extracted features are

1. whoisYear - Age of the domain
2. alexaRank - alexa rank of the domain
3. urlLength - length of the URL
4. hostLength - length of the host name
5. atPresent - whether @ sign present in the URL

6. ipPresent - whether IP address present in the URL
7. noOfdash - number of dashes(-) present in the URL
8. noOfUnderscore - number of underscores(_) present in the URL
9. noOfDots - number of dots(.) present in the URL
10. containsWords - whether URL contains words "secure", "account", "webscr", "login", "ebayisapi", "signin", "banking", "confirm"

WHOIS database stores details about web domains. By using Java WHOIS client we access these information. By submitting each URL in the data set the domain created date was extracted from the response.

By connecting to the Alexa API we can get the Alexa rank for each domain. By using java-programming language, for each URL in the dataset Alexa rank was extracted for the training and testing the phishing model.

Length of the URL and also the length of the host were calculated in the implemented java-program.

In atPresent feature we check if a given URL contains the '@' symbol and In ipPresent feature we check if the URL has an IP address.

Using the given URL string number of dashes, underscores and number of dot per URL in the dataset was calculated.

In containsWord feature we check if the URL contains identified suspicious words like secure, account, webscr, login, ebayisapi, signin, banking and confirm.

Important Java code segments related to extracting URL features are listed in Appendix D.

4.3 Defining membership functions for URL features

Next step is to define membership functions for each URL features mentioned above. Membership functions are needed in the process of generating membership values for fuzzy variables. For each URL a crisp value for each feature is extracted and these values are used to determine the degree to which it belong to each appropriate fuzzy set. When defining membership functions the valid range of inputs are considered and divided into classes or fuzzy sets For example for the length of URL address classes are defined as 'short', 'medium' and 'long'. We cannot specify clear boundaries between classes. The degree of belongingness of the values of variables to any selected class is called the degree of membership. A membership function is designed for each phishing characteristic indicator, which is a curve

that defines how each point in the input space is mapped to a membership value (or degree of membership) between [0, 1]. All membership functions for the URL features are listed in Appendix A.

An example of the linguistic descriptors used to represent one of the key phishing characteristic indicators (URL Address Length) and a plot of the fuzzy membership functions are shown in Figure 4.1 below. Membership values are shown in Table 4.1. The x-axis in each plot represents the range of possible values for the corresponding key phishing characteristic indicators (Short, Medium and Long). The y-axis represents the degree to which a value for the key phishing characteristic indicators is represented by the linguistic descriptor.

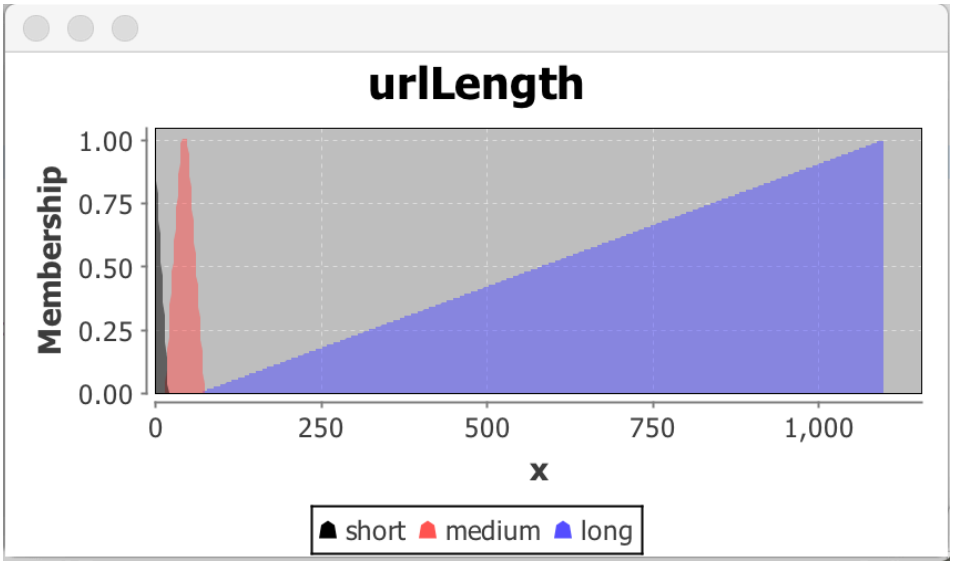
URL Address Length - Short, Medium and Long

Linguistic Variable : Length of URL

Table 4.1 URL length Membership Function values

Linguistic value	**Numeric Range**
Short	[0, 20]
Medium	[15, 40, 50, 75]
Long	[70, 1100]

Figure 2.1 URL length membership functions



In fuzzy model (.fcl) the URL length is defined as follows

```
FUZZIFY urlLength                    // Fuzzify input variable 'urlLength': {'short', 'medium' , 'long'}
        TERM short := (0, 1) (20, 0) ;
        TERM medium := (15, 0) (40,1) (50,1) (75,0);
        TERM long := (70, 0) (1100, 1);
END_FUZZIFY
```

4.4 Fuzzy rule generation using heuristic approach

Fuzzy rules were generated in this step. When developing fuzzy logic models, the experts define fuzzy rules. Therefore the accuracy of the model depends on their knowledge. In our approach to eliminate this and to automate the rule generation process data mining classification based approach was used. In this step 5000 phishing URLs and 5000 legitimate URLs were used. For each of these URLs previously described 10 features were extracted. Then by using the defined fuzzy membership functions for each of these features fuzzy membership class was defined. Then the data set was converted to .arff format. This file was input to WEKA data mining tool [17].

```
@relation phishingtest

@attribute urlLength {short, medium, long}
@attribute hostLength {short, medium, long}
@attribute noOfdash {high, low}

@data
medium, medium, low, low, low, no, no, no, old, high, false
long, medium, low, low, low, no, no, no, none, none, true
 …
```

Appendix C contains a sample .arff file.

Using available classification algorithms such as JRip, J48, PART rules were identified. Outputs of WEKA tool for each algorithm are shown in Appendix E.

JRip

JRip (RIPPER) is one of the most popular and basic algorithms. Classes are examined in growing size and an initial set of rules for the class is generated using incremental reduced error. JRip proceeds by treating all the examples of a particular decision in the training data as a class, and finding a set of rules that cover all the members of that class. Then it proceeds to the next class and does the same. This algorithm was used to identify rules to input to fuzzy inference process.

J48

J48 is an implementation of C4.5 algorithm. There two methods in pruning support by J48 first are known as subtree replacement, it work by replacing nodes in decision tree with leaf. It works with the process of starting from leaves that overall formed tree and do a backward toward the root. The second type implemented in J48 is subtree raising by moved nodes upwards toward the root of tree and also replacing other nodes on the same way.
C4.5 algorithm produce decision tree classification for a given dataset by recursive division of the data and the decision tree is grown using Depth-first strategy. J48 also used in WEKA data mining tool to identify the rules.

PART

PART is a separate-and-conquer rule learner. The algorithm produce sets of rules called "decision lists" which are planned set of rules. A new data is compared to each rule in the list in turn, and the item is assigned the class of the first matching rule. PART builds a partial C4.5 decision tree in each iteration and makes the "best" leaf into a rule. Therefore PART algorithm is also used to identify the Rules.

Using the above algorithms following 18 rules were identified based on the accuracy of the final fuzzy model. Numbers of evaluation processes were carried out to identify the best rules for the model. Experimental approach was used and different rule combinations were tested to identify the highest accuracy model.

RULE 1 : IF alexaRank IS high AND containsWords IS no THEN phishing IS legitimate;
RULE 2 : IF hostLength IS medium AND whoisYear IS old THEN phishing IS legitimate;
RULE 3 : IF whoisYear IS new THEN phishing IS phish;
RULE 4 : IF urlLength IS short THEN phishing IS legitimate;

RULE 5 : IF alexaRank IS low AND noOfUnderscore IS low AND urlLength IS medium THEN phishing IS phish;
RULE 6 : IF alexaRank IS none AND urlLength IS medium THEN phishing IS phish;
RULE 7 : IF alexaRank IS none AND containsWords IS yes THEN phishing IS phish;
RULE 8 : IF containsWords IS no AND noOfdash IS low AND urlLength IS medium AND whoisYear IS none THEN phishing IS legitimate;
RULE 9 : IF alexaRank IS none AND urlLength IS medium THEN phishing IS phish;
RULE 10 : IF urlLength IS long AND alexaRank IS none THEN phishing IS phish;
RULE 11 : IF urlLength IS long THEN phishing IS phish;
RULE 12 : IF alexaRank IS none AND urlLength IS long THEN phishing IS phish;
RULE 13 : IF alexaRank IS none AND whoisYear IS new AND urlLength IS medium THEN phishing IS phish;
RULE 14 : IF alexaRank IS none AND containsWords IS yes THEN phishing IS phish;
RULE 15 : IF alexaRank IS none AND whoisYear IS old THEN phishing IS phish;
RULE 16 : IF alexaRank IS none AND urlLength IS medium THEN phishing IS phish;
RULE 17 : IF urlLength IS long THEN phishing IS phish;
RULE 18 : IF alexaRank IS low AND hostLength IS long AND containsWords IS yes THEN phishing IS phish;

4.5 Implementing fuzzy model

Fuzzy model was developed using jFuzzyLogic library [18]. It is an open source java library implementing industry standards to develop fuzzy systems. jFuzzyLogic implements Fuzzy control language (FCL) [19] specification IEC 61131 part 7. FCL is defined as a 'Control language', so the main concept is a 'control block' which has some input and output variables.

When developing the fuzzy model first FUNCTION BLOCK is defined. Then input and output variables are defined. Next we define how each input variable is fuzzified is defined in FUZZIFY block. In each block we define one or more TERMs(also called LinguisticTerms). Each term is composed by a name and a membership function. Output variables then defuzzified to get a 'real' output number. Defuzzifiers are defined in DEFUZZIFY blocks. Linguistic terms are defined just like in FUZZIFY blocks. For the defuzzification Left Most Max (LM) method was used. Final section of the fuzzy model is the RULEBLOCK. All the fuzzy rules are defined here.

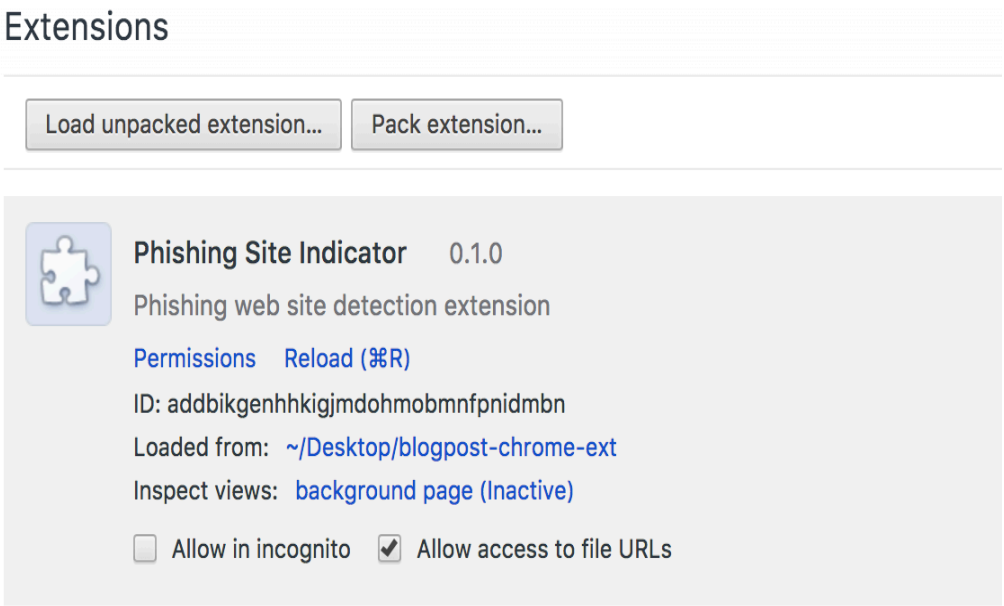
To use the implemented fuzzy model in Java programming language. First we need to load the fcl file that we created. Then we need to set values for each URL feature for a given URL. Finally we need to call the evaluate method to get the final crisp value. Depending on the final result of the fuzzy model we can identify if a given URL is a phishing URL or a legitimate URL.

Complete fcl file, which contains all described sections, is added in Appendix B.

4.6 Developing chrome browser extension to warn user about phishing sites

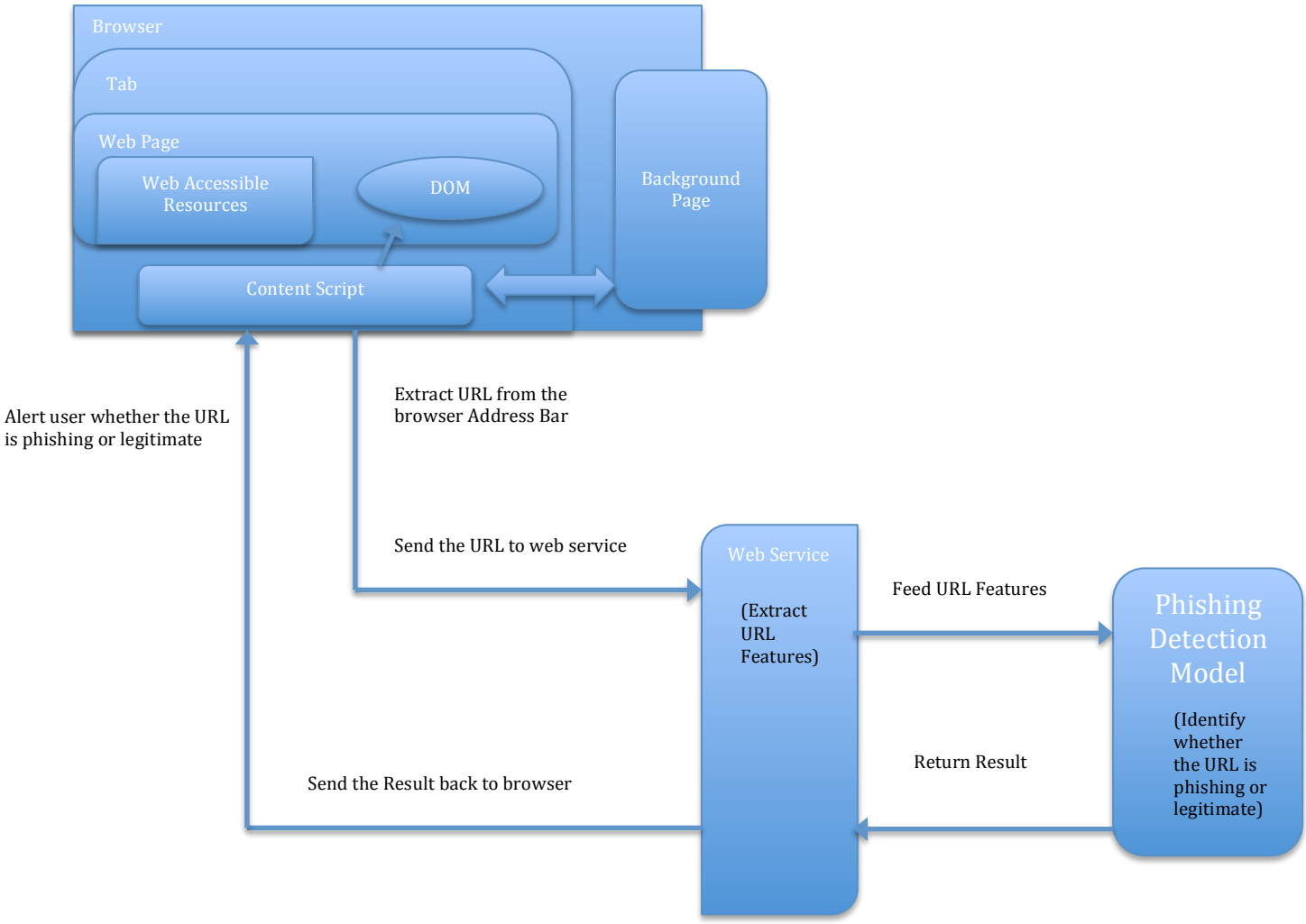
A Chrome Web Browser Extension (Figure 4.5) [20] was developed to use the implemented phishing site detection model. When user enters the URL the developed model will extract 10 URL features that described above and those values will be input to the developed phishing detection model. Depending on the extracted value the fuzzy model will decide whether the URL is a phishing URL or a legitimate URL. If the URL is legitimate then Browser Plugin Icon will indicate that by turning into green (Figure 4.6). If the URL is detected as phishing Browser plugin icon will indicate that by turning into red. And also a warning banner will be shown in the browser (Figure 4.7). So that user is alerted about the phishing site. Then user can be more careful not to enter usernames, passwords, credit card numbers etc. like personal information in these websites.

Figure 4.2 Chrome extension



Browser extension is a collection of files. It contains manifest.json, content.js, background.js, styles.css and jquery-3.2.1.min.js files. manifes.json file contains the main information about the extension such as name, version, scripts, default icons etc. Web server call to the phishing detection programs happens in content.js file. styles.css file contains the basic styles for the extension.

Figure 4.3 Architecture of implemented chrome web extension



In the content script of the chrome extension URL of the site is extracted from the browser address bar. Then it will be passed to the web service. In the web service URL features will be extracted. Then the feature will be fed to the Phishing detection model. Model will decide the given URL is a phishing URL or a legitimate URL. Phishing detection model will then return the result to the web service and web service will return the result back to the web browser. Depending on the returned value the chrome extension will alert the user about the status of the URL. If the URL is legitimate phishing indication icon will turn into green and if the URL is phishing icon will turn into red and also warning banner will be displayed.

Figure 4.4 Phishing extension icon indicator for legitimate sites

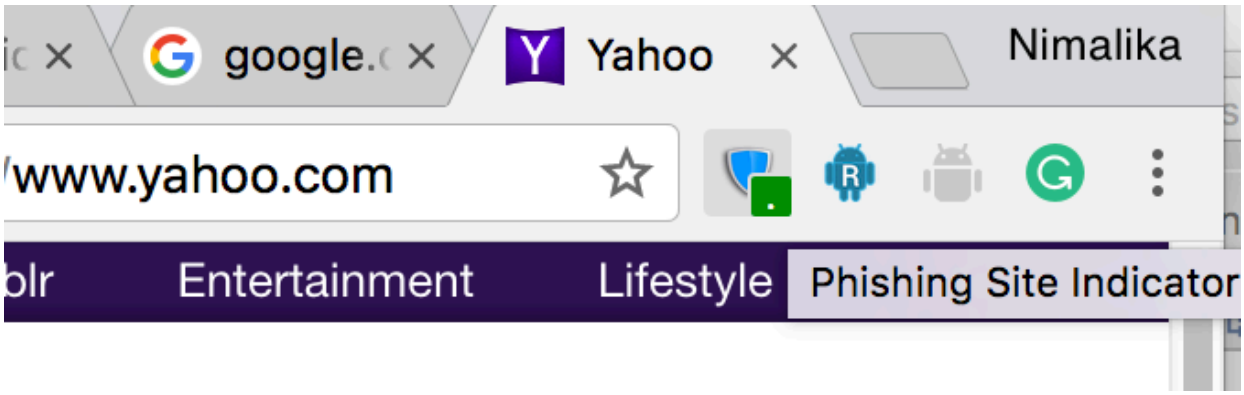
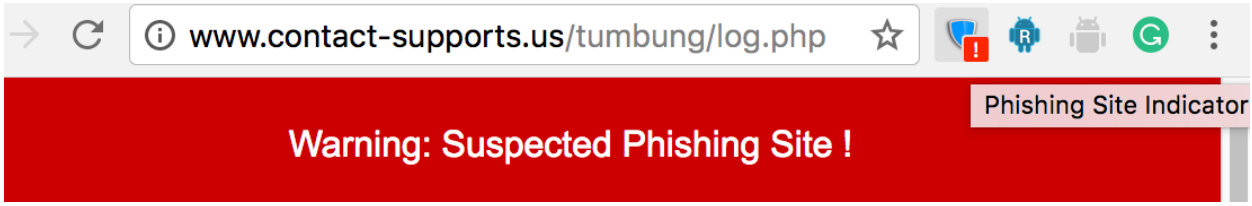


Figure 4.5 Phishing extension icon indicator and warning banner for phishing sites



Chapter 5

Evaluation And Testing

The outcome of the research is a phishing website detection model developed using heuristic based approach and fuzzy logic. Research was carried out to identify features of phishing web sites and legitimate websites so that the model is capable of identifying a given URL is phishing website or a legitimate website. The main focus was to develop a model with accepted accuracy level and accepted time.

Development of the model was done based on two main stages. In the stage one rules for the fuzzy model was identified using heuristic based approach. For that data mining classification approach was used. Main algorithms used to identify the rules are PART, JRip, J48. In fuzzy based approaches normally fuzzy rules are defined based on expert knowledge. Therefore the accuracy of the model depends on the knowledge of the expert. In this approach to eliminate this and to automate the rule generation process this classification-based approach was used. In this stage a phishing URL dataset and a legitimate URL dataset was used. Phishing URL dataset was gathered from the well-known 'phishtank' public data source. Public users can submit URLs to phishtank. Once submitted these URLs are verified by the registered users to confirm it as a phishing site. Legitimate URL dataset was gathered from well-known 'yahoo' directory. Yahoo maintains these data sources and they provide these datasets for research students free of charge. From these datasets 5000 phishing URLs and 5000 legitimate URLs were extracted for the training set (Sample phishing URLs and legitimate URLs are listed in Appendix F and Appendix G). Classification process was done using WEKA data mining tool.

The second stage of developing the model is applying fuzzy logic. The fuzzy model was developed using Jfuzzylogic library with java programming language. Eclipse tool was used to develop the program.

When developing the model, 10 features were used depending on the URLs that were extracted from the datasets. They are

- urlLength - length of the URL
- hostLength - length of the host name
- noOfdash - number of dashes(-) present in the URL

- noOfUnderscore - number of underscores(_) present in the URL
- noOfDots - number of dots(.) present in the URL
- atPresent - whether @ sign present in the URL
- containsWords - whether URL contains words "secure", "account", "webscr", "login", "ebayisapi", "signin", "banking", "confirm"
- ipPresent - whether IP address present in the URL
- whoisYear - Age of the domain
- alexaRank - alexa rank of the domain

5.1 Feature evaluation

CorrelationAttributeEval technique with Ranker search method in WEKA data mining tool was used to identify the correlation between each attribute and the output variable. It is a popular technique for selecting most relevant attributes in dataset. This method evaluates the worth of an attribute by measuring the correlation (Pearson's) between it and the class. Following is the result of CorrelationAttributeEval method (Figure 4.8). From the below result obtained from this method noOfHash and noOfTild attributes were removed as the correlation value of these two attributes are very low.

Figure 5.1 Correlation Attribute Eval method results

```
=== Run information ===

Evaluator:    weka.attributeSelection.CorrelationAttributeEval
Search:       weka.attributeSelection.Ranker -T -1.7976931348623157E308 -N -1
Relation:     phishingtest
Instances:    10000
Attributes:   13
              urlLength
              hostLength
              noOfdash
              noOfUnderscore
              noOfDots
              noOfHash
              noOfTild
              atPresent
              containsWords
              ipPresent
              whoisYear
              alexaRank
              phishing
Evaluation mode:    evaluate on all training data
```

```
=== Attribute Selection on all input data ===

Search Method:
        Attribute ranking.

Attribute Evaluator (supervised, Class (nominal): 13 phishing):
        Correlation Ranking Filter
Ranked attributes:
 0.552   12 alexaRank
 0.426    1 urlLength
 0.3215   9 containsWords
 0.2454  11 whoisYear
 0.2029   2 hostLength
 0.1964   5 noOfDots
 0.1666   8 atPresent
 0.0888   3 noOfdash
 0.0783  10 ipPresent
 0.0565   4 noOfUnderscore
 0.01     6 noOfHash
 0        7 noOfTild

Selected attributes: 12,1,9,11,2,5,8,3,10,4,6,7 : 12
```

Since the model was developed in two main stages the evaluation of the model is also done accordingly. Firstly the evaluation of the rules generated in the classification process was done and then the evaluation of the final fuzzy based model was done. Evaluation process was carried out based on experimental approach. For that two main datasets were used. One from phishtank database and the other from yahoo directory.

5.2 Rules evaluation

10 - fold cross validation is used to evaluate the rules. Dataset is divided into 10 separate groups and 9 out of the 10 parts are used to train the classifier. Then the information gathered in the training phase is used to test the 10th group. This is done for 10 times. At the end of the training and testing phase, each of the groups would have been used as either training or testing data. This method ensures that the training data is different from the test data.

Accuracy of the rules generated from the classification model (Weka tool) shows in Table 5.1

Table 5.1 - Accuracy of generated classification rules

Algorithm	**Accuracy %**
PART	86.68
JRip	86.29
J48	86.69

5.3 Fuzzy model evaluation

The final phishing detection model was evaluated based on following measurements. For that another 5000 phishing URLs from 'Phishtank' datasource and 5000 legitimate URLs from yahoo directory was used. Therefor the training URL dataset is different from test URL dataset.

Accuracy Measurement Methods

True Positive (TP)

Number of correctly classified Phishing URLs

True Negative (TN)

Number of correctly classified Legitimate URLs

False Positive (FP)

Number of Phishing URLs classified as Legitimate

False Negative (FN)

Number of Legitimate URLs classified as Phishing

Sensitivity and Specificity

Sensitivity measures the proportion of positives that are correctly classified as such.

Sensitivity (True Positive Rate) = TP / (TP + FN)

Specificity measures the proportion of negatives that are correctly classified as such.

Specificity (True Negative Rate) = TN / (TN + FP)

Precision and Recall

Precision is the ratio of the number of relevant records retrieved to the total number of relevant records in the dataset

Precision (Positive Predictive Value) = TP / (TP + FP)

Recall is the ratio of the number of relevant records retrieved to the total number of irrelevant and relevant records retrieved.

Recall (True Positive Rate) = TP / (TP + FN)

Accuracy

ACC = TP + TN / (TP + FP + FN + TN)

Following are the recorded values for above measurements of the developed phishing detection model. (5000 phishing URLs and 5000 legitimate URLs were used in testing)

True Positive = 4360 (87.2%**)**

True Negative = 3896 (77.92 %**)**

False Positive = 640 (12.8%**)**

False Negative = 1104 (22.08%**)**

Sensitivity = Recall = 0.7979

Specificity = 0.8589

Precision = 0.872

Accuracy = 0.8256

Confusion Matrix (Figure 5.2) [21].

A confusion matrix is a technique for summarizing the performance of a classification model. It contains information about actual and predicted classifications done by the system. Calculating a confusion matrix give a better idea of what the model is getting right and also about the errors.

Figure 5.2 Confusion Matrix

		Actual Value (As confirmed by experiment)	
		True	**False**
Predicted Value (Predicted by the test)	**True**	**(TP) 4360**	**(FP) 640**
	False	**(FN) 1104**	**(TN) 3896**

Research was carried out in different ways to identify the best model. First the classification model was developed only applying data mining technique. Different data mining algorithms were used to identify the highest accuracy model. When applying those algorithms initial model was developed from the URL features without giving a label depending on the membership function. By applying the above algorithms a classification model was develop. To further enhance the model fuzzy logic was applied. Initial fuzzy model was developed without applying the fuzzy rules from classification model. Later dataset of URL features were generated by applying membership function from the fuzzy model. Those generated result was used again in classification model. Then using that result classification rules were identified. These identified rules were then used in fuzzy model. By applying these steps phishing detection model was developed. The recorded accuracy rate of the model is 82.56 % and the average time to identify a given URL as phishing or legitimate is 1416 milliseconds.

Chapter 6

Conclusion and Future Work

6.1 Conclusions

When developing the phishing website detection model fuzzy logic has been combined with heuristic approach which use classification data mining technique. Previous studies related to phishing detection have been thoroughly examined to identify the limitations of current approaches. This approach has been proposed and implemented to develop a model with accepted accuracy level and reduced time for detection.

The fuzzy logic based detection model has been proposed using its four standard phases (Fuzzification, Rule Evaluation, Aggregation and Defuzzification). Phishing website features and patterns are characterized as fuzzy variables with specific fuzzy sets. Experts define fuzzy rules when developing fuzzy models. To eliminate this and to automate the rule generation process data mining classification process has been used. For that classification algorithms JRip (RIPPER), PART, and Decision Tree (J48) were used in WEKA data mining tool. To generate these rules 5000 phishing URLs from well-known Phish Tank website and 5000 legitimate URLs from Yahoo directory were used. To get the generated rules with fuzzy membership class labels first the data set was converted to values with fuzzy membership classes using the membership functions that were defined in fuzzy model. To evaluate the rules generated in classification, 10-fold cross validation was used. Mining association classification rules were then combined with the fuzzy logic inference engine to provide efficient and competent techniques for phishing website detection.

The results shows that data mining associative classification fuzzy-based solutions are actually quite effective in building detection solutions for protecting users against phishing websites attacks. We believe our model can be used to improve existing anti-phishing approaches. Using this approach will automate the fuzzy rule generation process and reduce the human intervention in building an effective phishing detection intelligent model.

Chrome extension was developed to use in Chrome browser so that the user is notified if user access a phishing web page. Developed chrome extension uses the implemented fuzzy based

phishing detection model to identify whether the current URL is a phishing URL or a legitimate URL. The intelligent phishing detection extension reduces the requirement of human knowledge intervention for detection of a phishing website. This approach has been provided as an alternative solution of depending only on the black-list or white-list approach, by adopting a new fuzzy-based classification mining technique to detect phishing website.

The results of our testing and validation show that the proposed approach is able to produce a phishing detection model with higher accuracy rate. It managed to classify correctly approximately 82% of all tested websites.

Following are summary of the main contributions:

• A study has been carried out to identify the limitation of existing phishing detection approaches.

• 10 phishing features were identified which shows the highest impact to categorize a given URL as phishing or legitimate.

• Enhanced model has been proposed to detect phishing websites which combines fuzzy logic and heuristic approach which includes data mining classification algorithms.

• A web-based chrome extension has been designed for notifying the users about the phishing websites.

The model was developed in Java programming language using JFuzzy logic library together with FCL type to define the fuzzy model. Model was able to show 82.56% accuracy rate.

6.2 Future Work

A fuzzy logic and data mining based approach has been introduced for building an intelligent phishing website detection system. This kind of supervised machine learning technique that combined the fuzzy logic model with the associated classification technique for detecting phishing websites verified lots of potential for its validity and usability throughout the research investigation.

As future work, we want to extend our work by integrating our phishing website detection model to all other standard browsers for example FireFox, Internet Explorer, Safari etc.

Also the model can integrate other supervised machine learning techniques like Neural Network. Also we can use other data mining classification algorithms to find the rule set for the fuzzy model. In our approach we have used only 10 features to input to the system. The accuracy depends on the selected features. We can add few more features and test the accuracy level. Also we can incorporate deep learning techniques to better understand the relationships among features to produce better result.

References

[1] V. Suganya, “A Review on Phishing Attacks and Various Anti Phishing Techniques,” *International Journal of Computer Applications*, vol. 139, Apr. 2016.

[2] Anti Phishing Working Group. (2016. February.) APWG Phishing Activity Trends Report 4th Quarter 2016. [Online]. Available: http://docs.apwg.org/reports/apwg_trends_report_q4_2016.pdf

[3] “PhishTank,” PhishTank. [Online]. Available: https://www.phishtank.com

[4] “Yahoo,” Yahoo. [Online]. Available: https://webscope.sandbox.yahoo.com/#datasets.

[5] A. Jain and B. Gupta, "A novel approach to protect against phishing attacks at client side using auto-updated white-list", *EURASIP Journal on Information Security*, 2016.

[6] Lee, D. Kim and C. Hoon, "Heuristic-based Approach for Phishing Site Detection Using URL Features", *International Journal of Advances in Computer Networks and Its Security–IJCNS*, vol. 5, no. 2, 2015.

[7] J. Solanki and R. G. Vaishnav, "Website Phishing Detection using Heuristic Based Approach", *International Research Journal of Engineering and Technology (IRJET)*, vol. 3, no. 5, 2016.

[8] K. Kumar and K. Alekhya, "Detecting Phishing Websites using Fuzzy Logic", *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)*, vol. 5, no. 10, 2016.

[9] P. Barraclough and G. Fehringer, "Intelligent Detection for Cyber Phishing Attacks using Fuzzy Rule-Based Systems", *International Journal of Innovative Research in Computer and Communication Engineering*, vol. 5, no. 6, 2017.

[10] S. Nivedha, S. Gokulan, C. Karthik, R. Gopinath and R. Gowshik, "Improving Phishing URL Detection Using Fuzzy Association Mining", *The International Journal of Engineering and Science (IJES)*, vol. 6, no. 4, pp. 21-31, 2017.

[11] A. Sharieh and R. Jabri, "Enhance Detecting Phishing Websites Based on Machine

Learning Techniques of Fuzzy Logic with Associative Rules", The University of Jordan Amman, Jordan, 2017.

[12] S. Sathish and A. Thirunavukarasu, "Phishing Webpage Detection for Secure Online Transactions", *IJCSNS International Journal of Computer Science and Network Security*, vol. 15, no. 3, 2015.

[13] M. Aburrous, M. Hossain, K. Dahal and F. Thabtah, "Intelligent phishing detection system for e-banking using fuzzy data mining", *Expert Systems with Applications*, pp. 7913–7921, 2017.

[14] R. Shah, A. Hossin and A. Khan, "Intelligent Phishing Possibility Detector", *International Journal of Computer Applications*, vol. 148, no. 7, 2016.

[15] "WHOIS | Lookup Domain Name Availability - GoDaddy SG", *GoDaddy*, 2018. [Online]. Available: https://sg.godaddy.com/whois.

[16] "Alexa Internet - About Us", *Alexa.com*, 2018. [Online]. Available: https://www.alexa.com/about.

[17] "WEKA Approach for Comparative Study of Classification Algorithm", *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 2, no. 4, 2013.

[18] P. Cingolani and J. Alcala´-Fdez, "jFuzzyLogic: a Java Library to Design Fuzzy Logic Controllers According to the Standard for Fuzzy Control Programming", *International Journal of Computational Intelligence Systems*, vol. 6, pp. 61-75, 2013.

[19] "Fuzzy Control Language (FCL)", Ffll.sourceforge.net. [Online]. Available: http://ffll.sourceforge.net/fcl.htm.

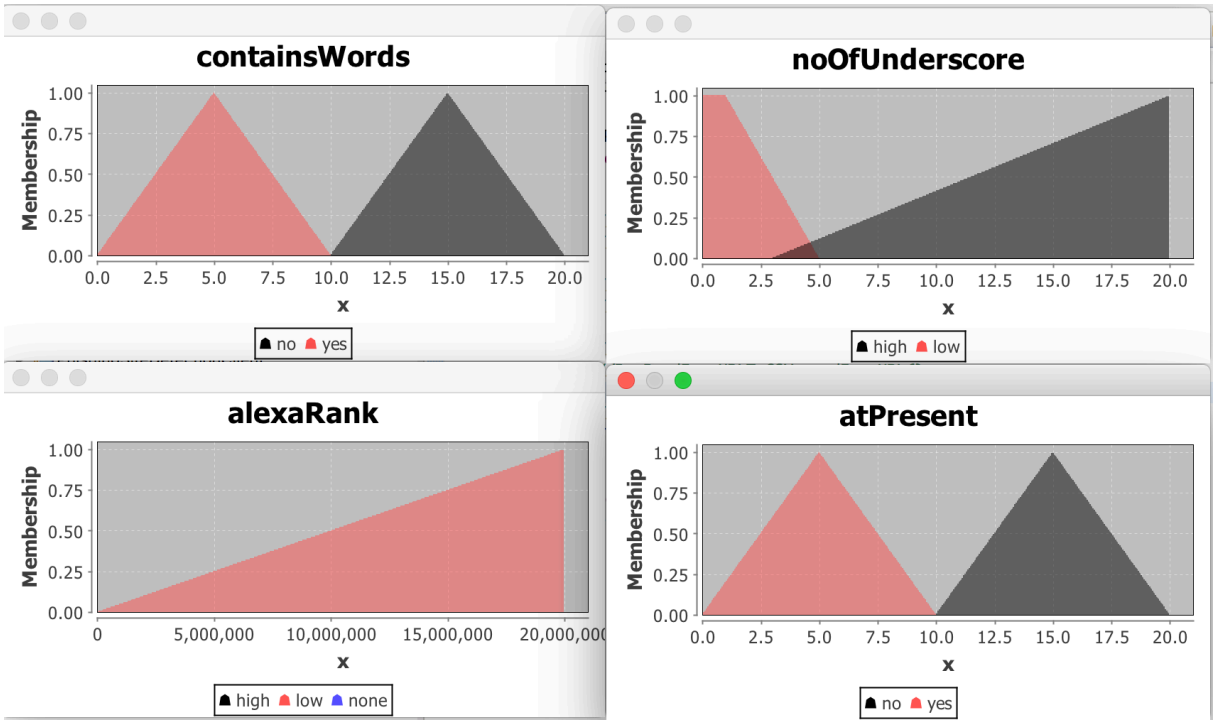
[20] "What are extensions? - Google Chrome", Developer.chrome.com, 2018. [Online]. Available: https://developer.chrome.com/extensions.

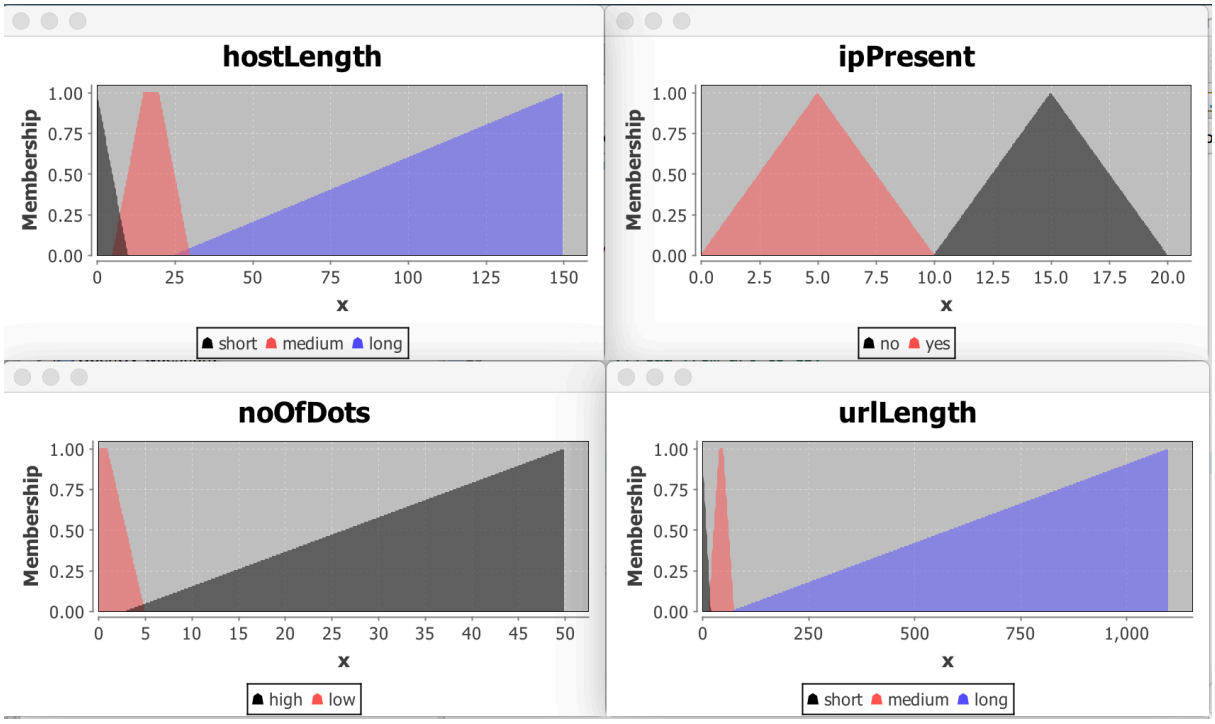
[21] U. DBD, "Confusion Matrix", Www2.cs.uregina.ca, 2018. [Online]. Available: http://www2.cs.uregina.ca/~dbd/cs831/notes/confusion_matrix/confusion_matrix.html.

Appendices

Appendix A

Membership functions






phishing
Membership
1.00
0.75
0.50
0.25
0.00
0.0 2.5 5.0 7.5 10.0 12.5 15.0 17.5 20.0
x
phishing:0.00 (LeftMostMax) legitimate phish
whoisYear
Membership
1.00
0.75
0.50
0.25
0.00
0 250 500 750 1,000 1,250 1,500 1,750 2,000
x
new old none
noOfdash
Membership
1.00
0.75
0.50
0.25
0.00
0 5 10 15 20 25 30 35 40 45 50 55 60
x
high low

Appendix B

Phishing.fcl file

```
/*
        Identify phishing web sites
*/

FUNCTION_BLOCK tipper          // Block definition (there may be more than one block per file)

VAR_INPUT                      // Define input variables

        urlLength: REAL;
        hostLength: REAL;
        noOfdash : REAL;
        noOfUnderscore : REAL;
        noOfDots : REAL;
        atPresent : REAL;
        containsWords : REAL;
        ipPresent : REAL;
        whoisYear : REAL;
        alexaRank : REAL;
END_VAR

VAR_OUTPUT                     // Define output variable
        phishing : REAL;
END_VAR

FUZZIFY urlLength              // Fuzzify input variable 'urlLength': {'short', 'medium' , 'long'}
        TERM short := (0, 1) (20, 0) ;
        TERM medium := (15, 0) (40,1) (50,1) (75,0);
        TERM long := (70, 0) (1100, 1);
END_FUZZIFY

FUZZIFY hostLength             // Fuzzify input variable 'hostLength': {'short', 'medium' , 'long'}
        TERM short := (0, 1) (10, 0) ;
        TERM medium := (5, 0) (15,1) (20,1) (30,0);
        TERM long := (25, 0) (150, 1);
END_FUZZIFY

FUZZIFY noOfdash               // Fuzzify input variable 'noOfdash': { 'low', 'high' }
        TERM low := (0, 1) (1, 1) (5,0) ;
        TERM high := (3,0) (60,1);
END_FUZZIFY

FUZZIFY noOfUnderscore         // Fuzzify input variable 'noOfUnderscore': { 'low', 'high' }
        TERM low := (0, 1) (1, 1) (5,0) ;
        TERM high := (3,0) (20,1);
END_FUZZIFY

FUZZIFY noOfDots               // Fuzzify input variable 'noOfDots': { 'low', 'high' }
        TERM low := (0, 1) (1, 1) (5,0) ;
        TERM high := (3,0) (50,1);
END_FUZZIFY

FUZZIFY atPresent              // Fuzzify input variable 'atPresent': { 'yes', 'no' }
        TERM yes := (0,0) (5,1) (10,0);
        TERM no := (10,0) (15,1) (20,0);
END_FUZZIFY

FUZZIFY containsWords          // Fuzzify input variable 'containsWords': { 'yes', 'no' }
        TERM yes := (0,0) (5,1) (10,0);
        TERM no := (10,0) (15,1) (20,0);
END_FUZZIFY

FUZZIFY ipPresent              // Fuzzify input variable 'ipPresent': { 'yes', 'no' }
        TERM yes := (0,0) (5,1) (10,0);
        TERM no := (10,0) (15,1) (20,0);
END_FUZZIFY
```

```
FUZZIFY whoisYear                       // Fuzzify input variable 'whoisYear': {none', 'old' , 'new'}
        TERM none := (0, 1) (250, 0) ;
        TERM old := (200, 0) (2000,1) (2010,1) (2014,0);
        TERM new := (2013, 0) (2050, 1);
END_FUZZIFY

FUZZIFY alexaRank                       // Fuzzify input variable 'alexaRank': {none', 'high' , 'low'}
        TERM none := (0, 1) (1, 0) ;
        TERM high := (1, 0) (2000,1) (10000,1) (55000,0);
        TERM low := (50000, 0) (20000000, 1);
END_FUZZIFY

DEFUZZIFY phishing                      // Defzzzify output variable 'phishing' : {'phish', 'legitimate' }
        TERM phish := (0,0) (5,1) (10,0);
        TERM legitimate := (10,0) (15,1) (20,0);
        METHOD : LM;            // Use 'Last of Maxima Method' defuzzification method
        DEFAULT := 10;          // Default value is 0 (if no rule activates defuzzifier)
END_DEFUZZIFY

RULEBLOCK No1
        AND : MIN;                      // Use 'min' for 'and' (also implicit use 'max' for 'or' to fulfill DeMorgan's
Law)
        ACT : MIN;                      // Use 'min' activation method
        ACCU : MAX;                     // Use 'max' accumulation method

        RULE 1 : IF alexaRank IS high AND containsWords IS no THEN phishing IS legitimate;
        RULE 2 : IF hostLength IS medium AND whoisYear IS old THEN phishing IS legitimate;
        RULE 3 : IF whoisYear IS new THEN phishing IS phish;
        RULE 4 : IF urlLength IS short THEN phishing IS legitimate;
        RULE 5 : IF alexaRank IS low AND noOfUnderscore IS low AND urlLength IS medium THEN phishing IS
phish;
        RULE 6 : IF alexaRank IS none AND urlLength IS medium THEN phishing IS phish;
        RULE 7 : IF alexaRank IS none AND containsWords IS yes THEN phishing IS phish;
        RULE 8 : IF containsWords IS no AND noOfdash IS low AND urlLength IS medium AND whoisYear IS none
THEN phishing IS legitimate;
        RULE 9 : IF alexaRank IS none AND urlLength IS medium THEN phishing IS phish;
        RULE 10 : IF urlLength IS long AND alexaRank IS none THEN phishing IS phish;
        RULE 11 : IF urlLength IS long THEN phishing IS phish;
        RULE 12 : IF alexaRank IS none AND urlLength IS long THEN phishing IS phish;
        RULE 13 : IF alexaRank IS none AND whoisYear IS new AND urlLength IS medium THEN phishing IS phish;
        RULE 14 : IF alexaRank IS none AND containsWords IS yes  THEN phishing IS phish;
        RULE 15 : IF alexaRank IS none AND whoisYear IS old  THEN phishing IS phish;
        RULE 16 : IF alexaRank IS none AND urlLength IS medium  THEN phishing IS phish;
        RULE 17 : IF urlLength IS long  THEN phishing IS phish;
        RULE 18 : IF alexaRank IS low AND hostLength IS long AND containsWords IS yes  THEN phishing IS phish;


END_RULEBLOCK

END_FUNCTION_BLOCK
```

Appendix C

Sample phishing.arff file

@relation phishingtest

@attribute urlLength {short, medium, long}
@attribute hostLength {short, medium, long}
@attribute noOfdash {high, low}
@attribute noOfUnderscore {high, low}
@attribute noOfDots {high, low}
@attribute atPresent {yes, no}
@attribute containsWords {yes, no}
@attribute ipPresent {yes, no}
@attribute whoisYear {none, old, new}
@attribute alexaRank {none, high, low}
@attribute phishing {true, false}

@data
medium, medium, low, low, low, no, no, no, old, high, false
long, medium, low, low, low, no, no, no, none, none, true
medium, medium, low, low, low, no, no, no, none, low, false
medium, long, low, low, low, no, yes, no, none, none, true
medium, medium, low, low, low, no, no, no, old, none, false
long, long, low, low, low, no, yes, no, none, none, true
medium, medium, low, low, low, no, no, no, none, high, false
short, short, low, low, low, no, no, no, new, none, false
medium, medium, low, low, low, no, no, no, none, none, false
long, medium, low, low, low, no, no, no, none, none, false
medium, medium, low, low, low, no, no, no, none, high, false
long, medium, low, low, low, no, yes, no, none, low, false
medium, medium, low, low, low, no, no, no, none, low, false
short, short, low, low, low, no, no, no, new, none, false
medium, medium, low, low, low, no, no, no, none, high, false
medium, medium, low, low, low, no, no, no, none, low, false
long, medium, low, low, low, no, no, no, none, none, false
medium, medium, low, low, low, no, no, no, none, low, false
long, medium, low, low, low, no, no, no, none, high, false
medium, medium, low, low, low, no, no, no, none, low, false
medium, medium, low, low, low, no, no, no, none, low, false

Appendix D

Important Java code segments

a. URL feature WHOIS year extraction

```
whois.connect(WhoisClient.DEFAULT_HOST);
String whoisData1 = whois.query("=" + domainName);

Pattern pattern = Pattern.compile("Creation Date: (.*?)T");
Matcher matcher = pattern.matcher(whoisData1);

while (matcher.find())
  {
   result.append(matcher.group(1));
  }
whois.disconnect();
```

b. *URL feature Alexa rank identification*

```
String url = "http://data.alexa.com/data?cli=10&url=" + domain;

URLConnection conn = new URL(url).openConnection();
InputStream is = conn.getInputStream();

DocumentBuilder dBuilder = DocumentBuilderFactory.newInstance()
                                        .newDocumentBuilder();
Document doc = dBuilder.parse(is);
Element element = doc.getDocumentElement();

NodeList nodeList = element.getElementsByTagName("POPULARITY");
if (nodeList.getLength() > 0)
{
        elementAttribute = (Element) nodeList.item(0);
```

```
        String ranking = elementAttribute.getAttribute("TEXT");
        if(!"".equals(ranking))
  {
        result = Integer.valueOf(ranking);
        }
}
```

c. *URL length and Host length extraction*

```
return URLString.length();

URL url = new URL(urlString);
return url.getHost().length();
```

d. *URL feature IP Present extraction*

```
String IPADDRESS_PATTERN =
                "(?:(?:25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?)\\.){3}(?:25[0-5]|2[0-4][0-
9]|[01]?[0-9][0-9]?)";

boolean ipPresent = false;

Pattern pattern = Pattern.compile(IPADDRESS_PATTERN);
Matcher matcher = pattern.matcher(text);
if (matcher.find()) {
        ipPresent = true;
} else{
        ipPresent = false;
}
```

e. *Count dashes, underscores and number of dot in the URL*

```
int count = 0;
        for (int i=0; i < url.length(); i++)
            {
                if (url.charAt(i) == character)
                {
                    count++;
                }
            }
return count;
```

f. *Check if URL contains suspicious words*

```
String[] words = {"secure", "account", "webscr", "login", "ebayisapi", "signin", "banking",
"confirm"};

return Arrays.stream(words).parallel().anyMatch(urlString::contains);
```

Appendix E

Rules generated from JRip, J48 and PART algorithms

JRip Output

```
JRIP rules:
===========

(alexaRank = none) and (urlLength = long) => phishing=true (1508.0/11.0)
(alexaRank = none) and (whoisYear = new) and (urlLength = medium) => phishing=true (531.0/2.0)
(alexaRank = none) and (containsWords = yes) => phishing=true (341.0/2.0)
(alexaRank = none) and (whoisYear = old) => phishing=true (402.0/56.0)
(alexaRank = none) and (urlLength = medium) => phishing=true (2186.0/699.0)
(urlLength = long) => phishing=true (244.0/32.0)
(alexaRank = low) and (hostLength = long) and (containsWords = yes) => phishing=true (6.0/0.0)
 => phishing=false (4782.0/584.0)

Number of Rules : 8
```

J48 Output

```
J48 pruned tree
------------------

alexaRank = none
|   urlLength = short: false (29.0)
|   urlLength = medium: true (3460.0/759.0)
|   urlLength = long: true (1508.0/11.0)
alexaRank = high
|   containsWords = yes: true (8.0/2.0)
|   containsWords = no: false (1424.0/100.0)
alexaRank = low
|   urlLength = short: false (4.0)
|   urlLength = medium
|   |   containsWords = yes: true (42.0/4.0)
|   |   containsWords = no
|   |   |   whoisYear = none
|   |   |   |   hostLength = short: true (4.0)
|   |   |   |   hostLength = medium: false (2864.0/290.0)
|   |   |   |   hostLength = long
|   |   |   |   |   noOfdash = high: true (2.0)
|   |   |   |   |   noOfdash = low: false (130.0/46.0)
|   |   |   whoisYear = old: false (283.0/93.0)
|   |   |   whoisYear = new: true (32.0/1.0)
|   urlLength = long
|   |   noOfUnderscore = high
|   |   |   noOfdash = high: true (22.0)
|   |   |   noOfdash = low: false (6.0)
|   |   noOfUnderscore = low: true (182.0/12.0)

Number of Leaves  :     16
```

PART Output

```
PART decision list
------------------

alexaRank = high AND
containsWords = no: false (1424.0/100.0)

alexaRank = none AND
urlLength = medium: true (3460.0/759.0)

urlLength = long AND
alexaRank = none: true (1508.0/11.0)

urlLength = long AND
whoisYear = none AND
noOfUnderscore = low: true (129.0/12.0)

containsWords = no AND
noOfdash = low AND
urlLength = medium AND
whoisYear = none: false (2998.0/340.0)

urlLength = long AND
whoisYear = old: true (62.0)

hostLength = medium AND
whoisYear = old: false (285.0/94.0)

alexaRank = low AND
noOfUnderscore = low AND
urlLength = medium: true (75.0/5.0)

urlLength = short: false (32.0)

whoisYear = new: true (14.0)

noOfdash = low AND
urlLength = long: false (6.0)

: true (7.0/2.0)

Number of Rules  :      12
```

Appendix F

Sample phishing URLs

- http://googlechroml.com.br/
- http://www.zonesegura-viabcp.com/
- http://wwwunicredmobilebr.com/index1.php
- http://100234566987.at.ua/3689/incorrect_email.html
- http://100234566987.at.ua/3689/confirmation.html
- http://paypal-info.729amca92.net/signin/mad5/websrc?country.x=US
- http://armazemdolores.com/bootstrap/css/adri/etapa1.html
- http://armazemdolores.com/bootstrap/css/adri/etapa1.html?PHPSSID=004347409749948&safe=736358885395740864214731535823451841811637053631661319
- http://armazemdolores.com/bootstrap/css/adri/
- http://bcpzonasegura.viabcp.net-per.com/OperacionesEnLinea/
- http://100354454896.at.ua/3257/incorrect_email.html
- http://100354454896.at.ua/3257/confirmation.html
- http://confirmesion012.support20.ga/
- http://bit.ly/2tiCcdp
- http://adm808.ucoz.ro/confirmation_now.html
- http://adm807.ucoz.ro/reconfirmation_now.html
- http://protect-facebok-center.esy.es/recovery-chekpoint-login.html
- http://viaszonaseguras.000webhostapp.com/vias/netsegura/
- http://devel0per11.regisconfrim.cf/
- http://updatesusersinfos.com/
- http://bpc.servitos.net/bpc23/web/OperacionesEnLinea
- http://bcpzonasegure-viabcp.info/
- http://bcpzonasegure-viabcp.info/bcp/
- https://bitly.com/2uXsqj2
- http://bcpzonasegure-viabcp.info/Promocion/Participantes
- http://helpnotice.5gbfree.com/fb-secure/index.htm
- http://help-secure-notice.hol.es/fb-confirm/index.php
- https://umohammedarchitects.com/lanterm/login.php
- http://register356-21.helpfanspagea.cf/
- http://login-update-paypal-login-service.grangerjhsmusic.com/auth/7791a7703dc8634bacb24c30826b3c5bYmRjMDFkZDdmMTc2ZGExYWE3Mzk2NTNhNzZiYmI5MjQ=/resolution/websc_login/?country.x=US&locale.x=en_US
- http://23.251.146.223/AZUL/Azul52276apc/index.php
- http://zonaseguras-viabcepl.esy.es/bcpsvias99dsd898/
- http://13456414584.at.ua/1248/incorrect_email.html
- http://websitet7.com/qi/yira/?i=1038672
- http://paypal-support-solution.com/
- http://www.contact-supports.us/tumbung/log.php
- http://jpj.log-ww.com/update-payment-logid/
- http://ass.hub-web.com/proccesing-fbaccountss/login.php
- http://jpj.log-ww.com/accounts-login-confirmphp/
- http://ajhzxbebaja.at.ua/2548/incorrect_email.html

Appendix G

Sample legitimate URLs

- http://boingboing.net/
- http://slashdot.org/
- http://del.icio.us/
- http://www.michaelbach.de/ot/index.html
- http://memepool.com/
- http://news.bbc.co.uk/
- http://www.gwu.edu/~nsarchiv/
- http://mrl.nyu.edu/~perlin/
- http://www.scorecard.org/
- http://www.danga.com/memcached/
- http://mathworld.wolfram.com/
- http://www.tikouka.net/mailapp/
- http://www.jiwire.com/
- http://www.nytimes.com/
- http://www.ritsumei.ac.jp/~akitaoka/saishin-e.html
- http://www.ex-parrot.com/~chris/driftnet/
- http://www.climateprediction.net/index.php
- http://www.alistapart.com/stories/practicalcss/
- http://www.jmarshall.com/tools/cgiproxy/
- http://www.puzzlepirates.com/
- http://magnatune.com/
- http://www.lares.dti.ne.jp/~yugo/storage/monocrafts_ver3/29/bclock.html
- http://csrc.nist.gov/
- http://www.ragingmenace.com/software/sidetrack/index.html
- http://www.textfiles.com/
- http://special.lib.umn.edu/swha/IMAGES/home.html
- http://openguides.org/
- http://www.goproblems.com/
- http://www.gigamonkeys.com/book/
- http://www.multipledigression.com/type/
- http://free.abracode.com/cmworkshop/
- http://sun3.lib.uci.edu/~jsisson/john.htm
- http://www.batbox.org/wrt54g-linux.html
- http://unxutils.sourceforge.net/
- http://www.cs.rochester.edu/sosp2003/papers/p125-ghemawat.pdf
- http://www.remotecentral.com/
- http://www.squarefree.com/bookmarklets/webdevel.html
- http://www.inknoise.com/experimental/layoutomatic.php
- http://www.wannabegirl.org/firdamatic/
- http://antwrp.gsfc.nasa.gov/apod/astropix.html
- http://www.themorningnews.org/archives/how_to/how_to_write_a_thankyou_note.php
- http://fishbowl.pastiche.org/2002/10/21/http_conditional_get_for_rss_hackers