PHISHING WEBSITE DETECTION USING INTELLIGENT DATA MINING TECHNIQUES

Design and Development of an Intelligent Association Classification Mining Fuzzy Based Scheme for Phishing Website Detection with an Emphasis on E-Banking

by

Maher Ragheb Mohammed Abur-rous

Submitted for the degree of Doctor of Philosophy

Department of Computing

University of Bradford

2010

PHISHING WEBSITE DETECTION USING INTELLIGENT DATA MINING TECHNIQUES

Thesis submitted for the degree of Doctor of Philosophy
by

Maher Ragheb Mohammed Abur-rous

ABSTRACT

***Keywords*:** Phishing, e-banking, fuzzy logic, association, classification, machine learning, data mining

Phishing techniques have not only grown in number, but also in sophistication. Phishers might have a lot of approaches and tactics to conduct a well-designed phishing attack. The targets of the phishing attacks, which are mainly on-line banking consumers and payment service providers, are facing substantial financial loss and lack of trust in Internet-based services. In order to overcome these, there is an urgent need to find solutions to combat phishing attacks. Detecting phishing website is a complex task which requires significant expert knowledge and experience. So far, various solutions have been proposed and developed to address these problems. Most of these approaches are not able to make a decision dynamically on whether the site is in fact phished, giving rise to a large number of false positives. This is mainly due to limitation of the previously proposed approaches, for example depending only on fixed black and white listing database, missing of human intelligence and experts, poor scalability and their timeliness.

In this research we investigated and developed the application of an intelligent fuzzy-based classification system for e-banking phishing website detection. The main aim of the proposed system is to provide protection to users from phishers deception tricks, giving them the ability to detect the legitimacy of the websites. The proposed intelligent phishing detection system employed Fuzzy Logic (FL) model with association classification mining algorithms. The approach combined the capabilities of fuzzy reasoning in measuring imprecise and dynamic phishing features, with the capability to classify the phishing fuzzy rules.

Different phishing experiments which cover all phishing attacks, motivations and deception behaviour techniques have been conducted to cover all phishing concerns. A layered fuzzy structure has been constructed for all gathered and extracted phishing website features and patterns. These have been divided into 6 criteria and distributed to 3 layers, based on their attack type. To reduce human knowledge intervention, Different classification and association algorithms have been implemented to generate fuzzy phishing rules automatically, to be integrated inside the fuzzy inference engine for the final phishing detection.

Experimental results demonstrated that the ability of the learning approach to identify all relevant fuzzy rules from the training data set. A comparative study and analysis showed that the proposed learning approach has a higher degree of predictive and detective capability than existing models. Experiments also showed significance of some important phishing criteria like URL & Domain Identity, Security & Encryption to the final phishing detection rate.

Finally, our proposed intelligent phishing website detection system was developed, tested and validated by incorporating the scheme as a web based plug-ins phishing toolbar. The results obtained are promising and showed that our intelligent fuzzy based classification detection system can provide an effective help for real-time phishing website detection. The toolbar successfully recognized and detected approximately 92% of the phishing websites selected from our test data set, avoiding many miss-classified websites and false phishing alarms.

ACKNOWLEDGMENT

First of all, I would like to thank god, ALLAH, whose grace has led me to this important moment of my life.

I would like to thank Dr. Alamgir Hossain, Dr. Keshav Dahal and Dr. Fadi Thabtah for their direction, assistance, and guidance. Their recommendations and suggestions have been invaluable for my research work. Our meeting and discussions have been extremely useful and valuable; your comments and advices have helped me a lot to follow the right direction in my research work. Thank you very much for your help and support.

Special thanks should be given to Mr. Nafi-Ul Karim for his help regarding the technical programming part of my model implementation

Also, I offer my regards and blessings to my colleagues who helped me in many ways during my research work

Finally, words alone cannot express the thanks and gratitude I owe to my wife (Heyam) for her encouragement, support and endless love, through the duration of my PhD research. Grateful thanks to my mother for all here prayers and my family for their ongoing support and love.

List of Acronyms and Abbreviations

AC	Associative Classification
AI	Artificial Intelligence
APWG	Anti-phishing Work Group
AURL	Abnormal URL
CAR	Class Association Rule
CBA	Classification based on Association Rule
DNS	Domain Name Service
DoS	Denial of Service
FL	Fuzzy Logic
HTML	Hyper Text Mark-up Language
HTTPS	Hyper Text Transfer Protocol Secured
IP	Internet Protocol
IREP	Incremental Reduced Error Pruning
ISP	Internet Service Provider
NN	Neural Network
RIPPER	Repeated Incremental Pruning to Produce Error Reduction
RURL	Abnormal Request URL
SFH	Server Form Handler
SSL	Security Socket Layer
URL	Uniform Resource Locater
WEKA	Waikato Environment for Knowledge Analysis
XML	User Interface Language

Contents

List of Figures

List of Tables

Publications

The research work carried out as part of this thesis has been published in international scientific refereed conference proceedings and journals. The list of published papers is provided below.

Conference Papers

1. Aburrous, M., Hossain, M.A., Thabtah, F., Dahal, K., "Intelligent Quality Performance Assessment for E-Banking Security using Fuzzy Logic," *Proceeding of the Fifth International Conference on Information Technology: New Generations (ITNG 2008)*, pp. 420-425, IEEE Computer Society Press , Las Vegas, Nevada, USA,2008.
2. Aburrous, M., Hossain, M.A., Thabtah, F., Dahal, K., "Intelligent Phishing Website Detection System using Fuzzy Techniques", *Proceeding of the 3rd International Conference on Information and Communication Technologies: From Theory to Applications, (ICTTA 2008)*, pp. 1-6, IEEE , Damascus, Syria, Apr 2008.
3. Aburrous, M., Hossain, M.A., Dahal, K., Thabtah, F., "Modelling Intelligent Phishing Detection System for E-banking Using Fuzzy Data Mining"**,** *Proceeding of the International Conference on CyberWorlds (CW '09)*, pp. 265-272, IEEE Computer Society Press*,* Bradford, UK, 2009.
4. Aburrous, M., Hossain, M.A., Thabtah, F., Dahal, K., "Classification Techniques for predicting e-Banking Phishing Websites"**,** *Proceedings of the*

*International Conference on Multimedia Computing and Information Technology (MCIT-2010)***,** pp. 9-12, IEEE**,** University of Sharjah, U.A.E, 2010.

5. Aburrous, M., Hossain, M.A., Dahal, K., Thabtah, F.,"Predicting Phishing Websites using Classification Mining Techniques with Experimental Case Studies", *Proceedings of the 7th Int'l Conf. on Information Technology: New Generations ITNG 2010*,pp. 176-181, IEEE Computer Society Press, Las Vegas, USA.

Journal Papers

1. Aburrous, M., Hossain, M.A., Dahal, K., Thabtah, F., "Experimental Case Studies for Investigating E-Banking Phishing Techniques and Attack Strategies ", *Journal of Cognitive Computation,* DOI 10.1007/s12559-010-9042-7, ISSN 1866-9956 (Print) 1866-9964 (Online)*,* Springer, April 2010.
2. Aburrous, M., Hossain, M.A., Dahal, K., Thabtah, F., "Intelligent phishing detection system for e-banking using fuzzy data mining", *Journal of Expert Systems with Applications*, (In Press) DOI:10.1016/j.eswa.2010.04.044, Elsevier, 20 May 2010.

Chapter 1

Introduction

1.1 Background

1.1.1 Internet Banking (e-banking)

Internet banking (e-banking) is defined as the automated delivery of new and traditional banking products and services directly to customers through interactive electronic communication channels. E-Banking includes the systems that enable customers, individuals or businesses, to access accounts, transact business, or obtain information on products and services through a public or private network, including the Internet (FFIEC, 2003). Commercial banking is undergoing rapid changes, as the international economy expands and advances towards institutional and market completeness. A major force behind these developments is technology, which is breaching geographical, industrial and regulatory barriers, creating new products, services and market opportunities, and developing more information and systems-oriented business and

management processes (Liao and Cheung, 2002). Banks across the world are motivated to implement e-banking to achieve either top-line or bottom-line benefits. This is achieved through increased market share due to product delivery convenience and product innovation (Jaleshgari, 1999; Orr, 1999). The Internet is the fastest growing banking channel today, both in the fields of corporate and retail banking. The development is no longer just driven by the banks' desire to save money: first and foremost it is a manifestation of customers' demand to access bank services on-line at any time and from any terminal.

Internet banking is rapidly becoming more and more popular as customers recognize the advantages Internet banking has to offer. It offers a cost effective alternative to telephone and branch banking services. Visiting a local branch bank costs around 68 times more than using Internet banking, and using the telephone would cost around 7 times more (Cryptomathic, 2004). For one most banks charge fewer fees when their customers take advantage of their online banking services. Customers can conduct 95% of their business over the Web, accessing their account and information, making payments and reconciling statements using computers rather than paper or phone to complete transactions. Instead of going down to the local branch bank office, Internet banking customers can accomplish multiple tasks at once with the click of a button. It can be accessed at any time from any Internet connection, and does not require any human interaction at the bank's end.

There are many advantages of Internet banking for customers who can use their computer from home or any site where they have regular access to a computer. The services are available 7 days a week, 24 hours a day and transactions are executed and confirmed almost instantaneously. The range of transactions available is normally fairly

broad. Customers can do anything from checking on an account balance to applying for a mortgage.

Banks also see advantages to offering their services on the Internet as follows:

- Opportunity to provide 24/7 client services
- Potential to offer more services
- Increased customer loyalty
- Ability to attract new customers
- Increased customer satisfaction
- Reduction in the need for data entry
- Reduction in costs, as the need for physical branches is reduced (Sukkar and Hasan, 2005).

Transactional websites provide customers with the ability to conduct transactions through the financial institution's website by initiating banking transactions or buying products and services. Since transactional websites typically enable the electronic exchange of confidential customer information and the transfer of funds, services provided through these websites expose a financial institution to higher risk than basic informational websites. Wholesale e-banking systems typically expose financial institutions to the highest risk per transaction, since commercial transactions usually involve larger amounts (FFIEC, 2003).

In developed countries, banking customers are increasingly taking advantage of on-line services, and this phenomenon is regularly studied by researchers. The willingness of consumers to adopt on-line banking usually depends on how Internet-aware they are.

Electronic banking is still young, although the acceleration in its adoption has been enormous and, at the current time, many banks offer their services through the Internet.

However, in the not-too-distant future, it is expected that every big bank will offer this service through the Internet in one way or another, with an increase in quality of service and performance (Sukkar and Hasan, 2005).

1.1.2 Phishing Websites

Phishing is a relatively new internet crime in comparison with other forms, e.g., virus and hacking. More and more phishing web pages have been found in recent years in an accelerative way (Fu, et al., 2006). Its impact is the breach of information security through the compromise of confidential data and the victims may finally suffer losses of money or other kinds. A phishing website as shown in Figure1.1 is a broadly launched social engineering attack that attempts to defraud people of their personal information including credit card number, bank account information, social security number, and their personal credentials in order to use these details fraudulently against them (James, 2006). Phishing has a huge negative impact on organisations' revenues, customer relationships, marketing efforts, and overall corporate image. Phishing attacks can cost companies tens to hundreds of thousands of dollars per attack in fraud-related losses and personnel time. Even worse, costs associated with the damage to brand image and consumer confidence can run into the millions of dollars (Brooks, 2006).

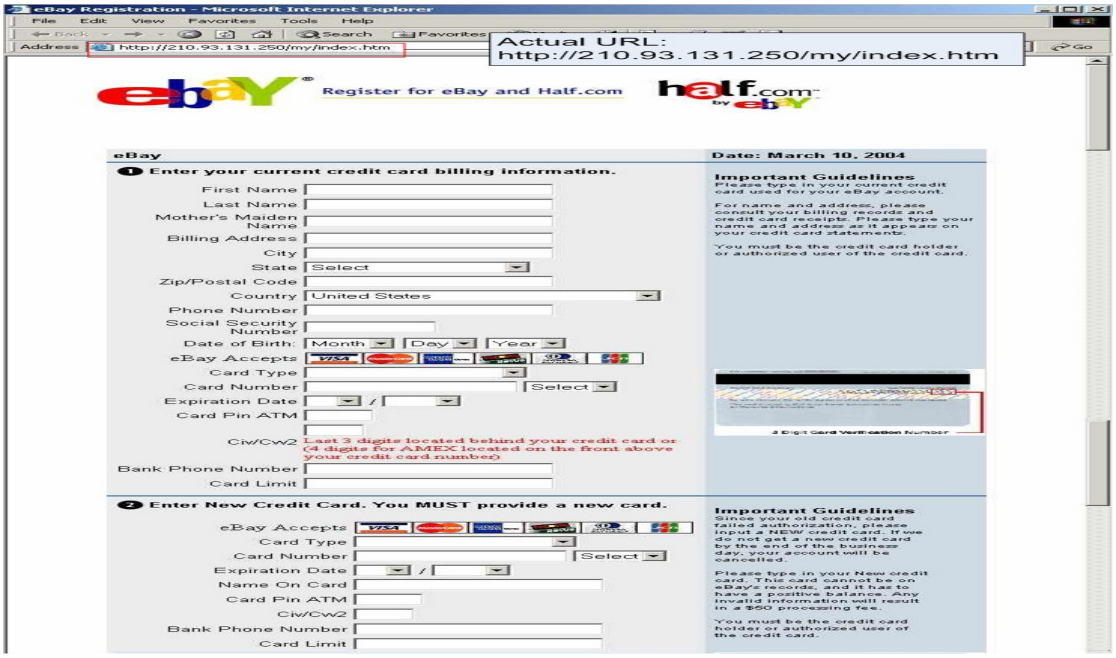


Figure1.1: Screenshot of a phishing website

Definition of Phishing Website

There are many definitions of phishing website; we want to be very careful how we define the term, since it is constantly evolving. One of these definitions comes according to the Anti-Phishing Working Group (APWG)'s definition (APWG, 2005), "Phishing attacks use both social engineering and technical subterfuge to steal consumers' personal identity data and financial account credentials". Typically a phishing attack is a combination of fraudulent emails, spoofed websites, and identity theft. Internet users or customers of many banks and financial institutions are the targets of phishing attacks (Ding and Li, 2006).

Nevertheless, there are lots of definitions of a phishing website from different perspectives. Hereunder we mention some of these definitions to get better understanding of its features and attack tactics.

Phishing is a particular type of spam which reflects social engineering. Phishing frauds are characterized by attempts to masquerade as a trustworthy person or emulate an established and reputed business in an electronic communication such as email or website (James, 2006). The objective is to trick recipients into divulging sensitive information such as bank account numbers, passwords, and credit card details. A person engaged in phishing activities is called a phisher (Chhabra, 2005). Phishing website attacks use web sites designed to look as if they come from a known and legitimate organisation, in order to deceive users into disclosing personal, financial, or computer account information. The attacker can then use this information for criminal purposes, such as identity theft, larceny, or fraud. Users are tricked into disclosing their information like bank accounts, credit cards etc., either by providing it through a web form or by downloading and installing hostile software (Wu, 2006). A phishing website is an attempt to commit fraud via social engineering. The impact is the breach of information security through the compromise of confidential data. A phishing website is a style of offence that network fishermen tempt victim with pseudo website to surrender important information voluntarily (Ming and Chaobo, 2006). The phishers usually camouflage themselves as a known bank, tradesman on line, a credit card corporation and so on (Qi and Yang, 2006). Phishing website is a form of electronic online identity theft in which the attackers use a combination of social engineering and web site spoofing techniques to trick a user into revealing confidential information. This information is typically used to make an illegal economic profit (e.g., by online banking transactions, purchase of goods using stolen credentials, etc.) (Ludl, et al., 2007). This is accomplished primarily by crafting a faux online presence to masquerade as a legitimate institution and soliciting information from unsuspecting customers (Seker, 2006). Phishing attacks involving websites are among the most commonplace and effective

types of online fraud, having the potential to cost both victims and targeted organisations in privacy, reputation, and monetarily (Zdziarski, et al., 2003).

We can summaries all these different definitions of phishing website in just one sentence, "Phishing website is the practice of creating a replica of an existing web page to fool a user into submitting personal, financial, or password data" (MAAWG and APWG, 2006).

Phishing websites use a number of different techniques to hide the fact that they are not authentic including overwriting or disguising the true URL shown in the browser, overlaying the genuine web site with a crafted pop-up window, drawing fake padlock images on top of the browser window to give the impression that SSL is enabled, and registering SSL certificates for domain names similar to the real organisation etc. In practice, these tricks make it extremely difficult for the average user to distinguish a phishing site from a genuine one (Gundel, 2005). Following the rapid development of online financial services and e-commerce, phishing website attacks have become one of the most dangerous and prevalent threats on Internet, causing inestimable damage. More and more phishing web pages have been found in recent years in an accelerative way (Fu, et al., 2006). To avoid phishing websites, both online financial organisations and their consumers have to understand phishing and anti-phishing technologies and take security actions. The scope and complexity of phishing activities are increasing very rapidly as phishing turns into an organized crime from a low budget amateur activity. Phishing website attacks not only cause significant financial damage to both individuals and companies and financial organisations, but also damage users' confidence in e–commerce and e-banking as a whole (Dong, et al., 2008). While most phishing attacks are relatively unsophisticated, there is a very clear trend towards them becoming more and more clever, both in terms of the psychological aspects and the technology

deployed. As this is occurring, the organisations concerned with preventing phishing attempts are also developing improved countermeasures. Without any definitive attack or countermeasure in sight, this is likely to remain a cat-and-mouse race where each party keeps trying to anticipate the other's next move (Jakobsson and Young, 2005).

1.1.3 Evolution of Phishing

At the beginning of phishing history, phishers were usually acting alone or in small, unsophisticated groups. Literature often portrays early phishers as adolescents desiring account data to cause mischief and to make long-distance phone calls, usually with a low level of organisation or malice (The Honeynet Project & Research Alliance, 2005). As financial organisations have increased their on-line presence and investment, the economic value of compromising on-line account information has increased dramatically. Phishing attacks became more and more professional, organized and systematic.

From the 1990s, following the popularity of Internet, America OnLine (AOL) became the first target of the phishing attacks. The first attempts at hacking into AOL were aimed at legitimate AOL accounts, and the phishing attacks were connected with the wares community which exchanges pirated software. There were programs (like AOHell) that automated the process of phishing for accounts and credit card information. Back then, phishing wasn't used as much in e-mail compared to Internet Relay Chat (IRC) or the messaging alert system that AOL used. Phishers usually pretended to be an AOL staff member and sent instant messages to the customers. They created messages such as "verify your account" or "confirm billing information" to lure

victims into revealing passwords or other sensitive information. The information they obtained would be used to trade in the wares community.

With the increasing growth of online financial services and e-commerce, the focus of phishing attacks turned to consumers of on-line banks, on-line retailers and other on-line service providers such as eBay or PayPal. The media of phishing are usually on-line forums of e-banks, Internet Relay Chatting (IRC), Instant Messaging (IM), and Email. Typically, the phisher poses as an employee of an on-line organisation, gains trust from the consumers of the organisation, and then deceives the consumers into sending out their sensitive information.

The sudden onslaught of phishing against financial institutions was first reported in July 2003. According to the Great Spam Archive, the targets were primarily e-loan, e-gold, Wells Fargo, and Citibank. The most remarkable twist about the phishing phenomenon is that it introduced a new class of attack vectors that was overlooked in almost every financial institution's security budget: the human element. All the expensive firewalls, SSL certificates, IPS rules, and patch management could not stop the exploitation of on-line trust that not only compromises confidential user information but has had a major impact on consumer confidence regarding telecommunications between an establishment and its clients.

Phishing started as e-mails written to convince the target to reply with the information asked for. This is still the most common method of initiating phishing attacks, but today phishers use several different ways to collect the information they require. Copied websites, Trojans, key-loggers and screen captures are just a number of different methods they are currently using (Jakobsson, et al., 2007).

Phishers began to create fake websites to increase the successful rate of phishing. For instance, phishers register dozens of domain names that look like a famous brand, such as “www.cit1bank.com” or “www.citi-bank.com”. Victims, who enter one of these websites by making mistakes in typing or by falling for the phisher’s ruse, may believe that the website is the real one, and operate their account on the website. Phishers embed website designs into the emails, completing them with stolen logos and trademarks from the targeted organisation, and forge the return address so that the address appears to come from the legitimate organisation (Jagatic, et al., 2005).

Many new attacks include a link to a legitimate banking website in the background, but a fake "login" box placed in front of the real site. Obviously it is more convincing because the legitimate site and the pop-up appear to be from the same source. After giving up personal financial information on a phishing site, the victim is redirected to the real home page of the company being targeted. Thus, the victim will not suspect the website of being false. Two user studies were conducted and the researchers found that actively interrupting a user with a pop-up message during a phishing attack is more effective than just a passive warning displayed in the browser toolbar (Hernandez and Leggio, 2006).

Phishing website attacks are growing at a torrid pace. The numbers of phishing attacks and reported phishing sites are increasing every year, even every month. Damage caused by phishing is severe. The APWG (Anti-Phishing Working Group) is an industry association focused on eliminating identity theft and fraud that result from the growing problem of phishing and email spoofing. This voluntary-based organisation provides a forum to discuss phishing issues, trials and evaluations of potential technology solutions, and access to a centralized repository of reports on phishing attacks (Zin and

Yunos, 2005). The number of unique phishing websites detected by this organisation showed that there has been a huge increase in unique phishing sites all over the world. In December 2005, the forged phishing site alone exceeded 7,000 (Brooks, 2006). APWG has also recently released a new report containing statistics of phishing attacks during the first half of 2009. According to the APWG global phishing survey report (APWG, 2009) there were at least 55,698 phishing attacks, around 7 per cent higher than the previous year. Those attacks occurred on 30,131 unique domain names. APWG identified that 4,382 were registered by phishers, representing about 14.5% of the domain names involved in phishing. In addition, phishing was detected on 3,563 unique IP addresses. The Gartner study (Gartner, 2007) shows that phishing attacks escalated in 2007; more than $3 Billion was lost to these attacks. The survey found that 3.6 million adults lost money in phishing attacks in the 12 months ending in August 2007, as compared with the 2.3 million who did so the year before**.** And, in 2008, Gartner reported a 39.8 per cent increase over the number of victims a year earlier. Media outlets have reported that phishing website-related scams have resulted in more than $5 billion in fraudulent bank and financial charges to date (Microsoft Corporation, 2008).

Phishing techniques have a short history compared with other Internet threats, but there have emerged tens of thousands of variations in the evolution of phishing, which makes the research into anti-phishing very difficult.

1.1.4 Phishing and the Trust of e-Banking Business

Phishing websites can severely hurt Internet business, because people lose their trust in Internet transactions for fear that they will become victims of fraud. For example, many people believe that using on-line banking increases the likelihood that they will become

victims of phishing websites and identity theft, even though on-line banking provides more secure identity protection than paper- and mail-based systems.

The most harmful effect is that it will create “trust crises”. The trust will be eroded gradually without effective countermeasures to deal with the fraud, and everyone participating in network transactions will be harmed in the end. Trust is one of the most important determinants of successful e-banking (Suh and Han, 2002). Many researchers have argued that trust is essential for understanding interpersonal behaviour and is relevant to e-banking. Trust is not merely a short-term issue, but also the most significant long-term barrier to realizing the potential of BtoC e-commerce (Gefen, 2002). Falling victim to phishing websites could steal a customer’s proprietary information such as their account information and passwords, trade secrets, or other intellectual assets. Theft of a customer’s confidential information could have a disastrous effect on the companies or banks using electronic technology and could damage the trust between them and their clients.

Even in developed countries, many people are worried that their credit card details will be misused or hacked into, and are concerned about on-line fraud, such as phishing websites that offer imaginary services or items.

1.2 Motivation

Phishing websites are forged web pages that are created by malicious people to mimic web pages of real websites. Most of these kinds of web pages have high visual similarities to scam their victims. Some of these kinds of web pages look exactly like the real ones. Victims of phishing web pages may expose their bank account, password, credit card number, or other important information to the phishing website owners.

Phishing website is a very complicated and complex issue to understand and to analyze, since it is a combination of technical and social dynamics for which there is no known single silver bullet to solve it entirely.

Despite the great quantity of applications available for phishing website detection, there are only a few solutions that utilise machine learning mining techniques in detecting phishing websites. Moreover, most of these proposed and already implemented solutions are impractical, inaccurate and suffer from unacceptable levels of false positives or miss detection (Wu, et al., 2006; Cranor, et al., 2008).

The motivation behind the present study is to create a resilient and effective intelligent model to detect phishing websites and to discover whether phishing activity is taking place or not, in order to prevent all users from being deceived or hacked.

The methodology approach of this research is quantitative, and it investigates intelligent phishing website detection system, based on an artificial intelligence (AI) supervised machine learning approach. The technique uses fuzzy logic with simple data mining associative classification techniques and algorithms to process the phishing data features and patterns, for extracting classification rules into the data miner. The proposed phishing website system combines these techniques together to automate the fuzzy rules, produced by using the extracted classification rules to be implemented inside the fuzzy inference engine. These fuzzy rules allow us to construct if-then rules, which reflect the relations between the different phishing characteristics and features and their association with each other, to be used for the final phishing website detection rate.

From my position as an IT manager of one of the biggest banks in Jordan, I realized the serious effect of phishing websites towards consumers trust and confidence to online banking services. This derived me to do many investigations and research studies towards finding a solution to overcome this problem, especially for naive banking clients and consumers.

1.3 Aims & Objectives

Our aim is to build a hybrid system which combines and integrates fuzzy logic with a supervised machine learning mining technique using variations of associative classification algorithms to provide an efficient technique for classifying and indentifying phishing website with low false positive and false negative detection rate. This new mechanism reduces the need for human intervention and enhances the performance and the precision of detecting phishing websites rate**.**

We can summarize our objectives with the following points:

1. Thorough literature review in order to demonstrate the existing state-of-the-art of technology.
2. Building an intelligent dynamic phishing website detection system that combines association classification mining techniques and fuzzy logic to detect phishing websites. The resulting system has to be practical, adaptive and low in false alarms.
3. Demonstrating the applicability that by analyzing a large number of phishing websites datasets and page properties, we can utilise supervised machine learning techniques using associative classification mining algorithms and fuzzy logic for phishing detection.

4. Proving the validity and the applicability of applying fuzzy logic-based expert systems, that uses phishing fuzzy rules driven by human expert knowledge for building resilient and flexible phishing website detection system.
5. Automating the generation of the phishing fuzzy rules using intelligent classification mining algorithms, in order to reduce the human knowledge intervention, and shorten the development time of the phishing classifier to provide more accurate and efficient outputs.
6. Providing a solution that improves existing anti-phishing approaches using an AI heuristic search. The solution will provide installable web browser plug-ins, which should be effective, accurate and work in real time.

Quantitative research methodology has been developed and implemented to achieve all our abovementioned objectives, taking into consideration experimental case-studies analysis, data gathering, testing measures and comparing results.

1.4 Introducing Basic Terminologies and Technologies

1.4.1 Fuzzy Logic Model

Fuzzy Logic (FL) is a problem-solving control system methodology that lends itself to implementation in systems ranging from simple, small, embedded micro-controllers to large, networked, multi-channel PCs or workstation-based data acquisition and control systems. It can be implemented in hardware, software, or a combination of both. FL provides a simple way to arrive at a definite conclusion based upon vague, ambiguous, imprecise, noisy, or missing input information. FL's approach to control problems mimics how a person would make faster decision. FL incorporates a simple, rule-based

‘IF X AND Y THEN Z’ approach to a solving control problem rather than attempting to model a system mathematically. The FL model is empirically-based, relying on an operator's experience rather than their technical understanding of the system.

The fuzzy logic approach provides more information to help risk managers effectively manage assessing and identifying phishing website risk rates than the current qualitative approaches as the risks are quantified based on a combination of historical data and expert input. Modelling techniques that can accommodate a combination of data and expert input are better suited for modelling phishing operational risks. Fuzzy logic has been used for decades in the computer sciences to embed expert input into computer models for a broad range of applications. It offers a promising alternative for measuring operational risks (Samir, 2003). The advantage of the fuzzy approach is that it enables processing of vaguely defined variables, and variables whose relationships cannot be defined by mathematical relationships. Fuzzy logic can incorporate expert human judgment to define those variables and their relationships. The model can be closer to reality and be more site specific than some of the other methods (Mahant, 2004).

In contrast to the true or false world of Boolean logic, fuzzy logic techniques allow the use of degrees of truth to calculate results. They allow one to represent concepts that could be considered to be in more than one category. In other words, these techniques allow representation of overlapping and partial membership in sets or categories (Bridges and Vaughn, 2001).

Fuzzy logic can be justified in our work since it can tolerate imprecisely-defined data, can model non-linear functions of arbitrary complexity and can build on the experience of experts (Mahant, 2004).

1.4.2 Data Mining

Data Mining is the automated extraction of previously unrealized information from large data sources for the purpose of supporting actions. The rapid development of data mining has made available a wide variety of algorithms, drawn from the field of statistics, pattern recognition, machine learning and databases. Fayyad, et al. (1998) defines data mining as one of the main phases in Knowledge Discovery from Databases (KDD), which extracts useful patterns from data. The availability of high speed computers, automated data collection tools and large memory capacities has made the process of collecting and storing huge quantities of information possible. The process of extracting this useful knowledge is accomplished using data mining techniques (Fayyad, et al., 1998; Elmasri and Navathe, 1999).

Consider a retail store with a large collection of sales transactions and customer information. The marketing division at the store is promoting a new credit card in a new geographical area. Typical business decisions have to be made such as how credit card limits are decided for each customer and how each customer's total purchases contribute to the decision process, etc. Finding associations between customer's different features can help the managers in making business decisions. These associations are known as association rules, an example of which is: "55% of customers who buy crisps are likely to buy a soft drink as well; 4% of all database transactions contain crisps and a soft drink".

"Customers who buy crisps" is known as rule antecedent, and "buy a soft drink as well" is known as rule consequent. The antecedent and consequent of an association rule contain at least one item. The 55% of the association rule mentioned above represents the strength of the rule and is known as rule's confidence, whereas the 4% is a statistical

significance measure, known as the rule's support. In a credit card application, the store's management is only interested in one class of association rules where the rules consequent is related to whether a credit card should be offered. They would like to develop an automated computer system, which analyses the customer's different attributes in a certain geographical area to come up with a set of rules. These rules are then used to assess credit card applications for new customers by predicting the credit card attribute. The subset of the association rules, which consider the credit attribute as the class attribute is known as **C**lass Association Rules (CARs) (Liu, et al., 1998). An example of a CAR is: "60% of rows that contain incomes which exceed 25*k* have been granted a credit card; 4% of all rows contain incomes exceeding 25*k*". Similar to the association rule approach, the 60% of the above CAR represents the confidence and the 4% denotes the support. The main significant difference between a CAR and an association rule is that the consequent of the CAR is only the class attribute, whereas in an association rule, the consequent could be multiple items (Freitas, 2000).

1.4.3 Association Rule Mining

Association rule algorithms find correlations between features or attributes used to describe a data set. Association rule mining can be decomposed into two sub-tasks (Agrawal, et al., 1993; Agrawal and Srikant, 1994): (1) The discovery of all frequent itemsets (those whose support is above the *minsupp* threshold) and (2) for each frequent itemset found, *Z*, produce rules of the form $X \rightarrow (Z - X)$, $X \subset Z$ whose confidence is above the *minconf* threshold. The support of an itemset in association rule mining is defined as the proportion of transactions in the database that contain that itemset and the confidence of a rule $X \rightarrow Z$, defined as support($X \cup Z$)/support(Z).

1.4.4 Traditional Classification Rule Mining

Given a training data set of historical transactions, the problem is to discover the CARs with significant supports and high confidences (attribute values that have frequencies above user specified minimum support and minimum confidence thresholds). One subset of the generated CARs is chosen to build an automatic model (classifier) that could be used to predict the classes of previously unseen data. This approach, which uses association rule mining to build classifiers, is called "associative classification" (AC) (Liu, et al., 1998, Li, et al., 2001). Unlike the classic classification approaches such as rule induction and decision trees which usually construct small sized classifiers, AC explores all associations between attribute values and their classes in the training data set, aiming to construct larger sized classifiers. This is because AC methods aim to produce useful knowledge missed by traditional methods, which therefore should improve the predictive accuracy within applications.

1.4.5 Associative Classification Rule Mining

The AC approach was introduced in 1997 by Ali, et al. (1997) to produce rules for describing relationships between attribute values and the class attributes and not for prediction, which is the ultimate goal for classification. In 1998, AC has been successfully employed to build classifiers by Liu, et al. (1998) and later attracted many researchers (e.g. Li, et al., 2000; Dong, et al., 1999; Yin and Han, 2003) from data mining and machine learning communities.

AC is a special case of association rule mining in which only the class attribute is considered in the rule's consequent (Liu et al., 1998). For example in a rule such as $X \rightarrow Y$, Y must be a class attribute. Let us define the AC problem, where training data set T has m distinct attributes $A1$, $A2 \ldots Am$ and C is a list of class labels. The number of

rows in T is denoted $|T|$. Attributes could be categorical (meaning they take a value from a finite set of possible values) or continuous (where they are real or integer). In the case of categorical attributes, all possible values are mapped to a set of positive integers. For continuous attributes, a discretisation method is first used to transform these attributes into categorical ones.

An AC task is different from association rule mining. The most obvious difference between association rule mining and AC is that the latter considers only the class attribute in the rules consequent. However, the former allows multiple attribute values in the rules consequent.

1.5 Contributions of this Research and Investigation

There are different topics and issues that arise in a phishing website case, including: Extraction techniques of phishing website main features and characteristics, Phishing datasets execution, Performing special phishing experiments and case studies for analyzing and collecting phishing factors and relation rules., Utilisation of fuzzy logic and data mining techniques for building intelligent phishing website detection model and practical plug-ins phishing toolbar implementation for testing and validating.
The main contributions of this research and investigations are introduced below.

1.5.1 Empirical Phishing Experimental Case-Studies

We have performed a number of experiments to cover all phishing concerns related to its approaches, motivations and deception behaviour techniques. We have implemented two phishing experiments which cover website phishing attack techniques and a survey scenario of phishing website detection procedure. We analyzed all these phishing

experiments and covered all their reactions. These different experiments helped us greatly in gathering and analyzing many of the different social engineering phishing features, characteristics and factors with their mutual relationships'.

1.5.2 Extracting Phishing Features

In this thesis, and by analyzing a large number of phishing pages, conducting different phishing experimental case studies, reviewing different phishing investigations, technical reports, research papers and implementing many phishing questionnaires and surveys, we managed to extract 27 features and factors which can characterize and signature any phishing website case or incident. We divide these features into 6 criteria or categories distributed into 3 layers, depending on its attack type. We give special weight to every phishing criteria and layer, based on different strategies and attack tactics, to be implemented on our detection model for more precision final output. We use these collected phishing website features and patterns on our experimental archive datasets for analyzing and testing, using specific mining association classification algorithms and techniques for automating classification rule generation. Layer 1 has one criterion (URL & Domain Identity), layer 2 has 2 criteria (Security & Encryption, Source Code & Java Script) and last layer has 3 criteria (Page Style & Contents, Web Address Bar, Social Human Factor). All criteria have different numbers of related phishing website features and patterns.

1.5.3 Fuzzy-based Association Classification Mining Model for Phishing Website Detection

We propose a dynamic intelligent phishing website detection system, based on a specific AI supervised machine learning approach. The technique used utilises fuzzy logic with simple data mining associative classification techniques and algorithms to process the phishing data features and patterns, for extracting classification rules into the data miner. The proposed phishing website system combines these techniques together to automate the fuzzy rules production by using the extracted classification rules to be implemented inside the fuzzy inference engine. These fuzzy rules, allow us to construct if-then rules that reflect the relations between the different phishing characteristics and features and its association with each other, to be used for the final phishing website detection rate.

In this thesis, we want to prove that by analyzing a large number of phishing websites datasets and page properties, we can use supervised machine learning techniques using associative classification mining algorithms and fuzzy logic to design and develop an intelligent efficient system that can predict and detect whether phishing activity is taking place on a website or not. That is why we believe that a hybrid system which combines and integrates fuzzy logic with supervised machine learning data mining technique, using variation of associative classification algorithms (CBA, JRip, PART, PRISM, C4.5) implemented into the Data Miner, allows for valuable phishing feature extraction and rule processing, providing efficient techniques for classifying and indentifying phishing website with low false positive and false negative detection rate. Finding association rules between phishing website different features and patterns which distinguish legitimate websites from phishing website can greatly help the performance and precision of the detection system, by automating a rule generation process using data miner classification rules as an alternative to human expert

knowledge to be implemented and processed by the fuzzy inference engine for the final output.

1.5.4 Implementation of Intelligent Phishing Website Detection Model (Plug-ins phishing toolbar)

We designed a plug-ins phishing website detection toolbar for testing and validation using our integrated association classification mining fuzzy model to show and prove its feasibility, reliability and accuracy. The implementation was programmed using Java language, and it successfully recognized and detected approximately 92% of the phishing websites selected from our test data subset, avoiding many miss-classified websites and false phishing alarms. Further, we show from this practical plug-in toolbar implementation that data mining fuzzy-based solutions are actually quite effective in protecting users against phishing websites attacks, and also we believe it can be used to improve existing anti-phishing approaches which use AI heuristics search.

1.6 Thesis Road Map

In this thesis, we introduce an extensive literature review concerning the phishing website problem and all its related work. Then we show from quantitative point of view our implementation of empirical phishing experiments and case studies to gather and analyze all the different social engineering phishing website features, characteristics and patterns with all their mutual relationships. Further, we present a fuzzy logic model for building a phishing website detection system using its four phases (Fuzzification, Rule Evaluation, Aggregation and Defuzzification). Fuzzy logic is used to characterize the phishing website features and patterns as fuzzy variables with specific fuzzy sets. Fuzzy

set operations are then used to generate all the phishing fuzzy rules which come from previous human expert knowledge, to be processed into the fuzzy inference engine for the final calculation of the phishing website risk rate. We then enhance the phishing fuzzy rule generation by combining fuzzy logic with intelligent mining association classification algorithms and techniques (CBA, JRip, PART, PRISM and C4.5). Finding association rules between phishing websites' different features which classify legitimate websites from phishing website can greatly help the performance and precision of the detection system. Automating the rule generation process using data miner association classification algorithms can be used as an alternative to human expert knowledge, to be processed by the fuzzy inference engine for the final output.

Finally, we show the practical implementation of our model for testing and validation. We design an intelligent plug-ins phishing website detection toolbar using our integrated association classification mining fuzzy model to show and prove its feasibility, reliability and accuracy.

1.7 Outline of the Thesis

The thesis consists of 7 chapters. Chapter 2 introduces a general literature review of phishing website problem, types, classification and different anti-phishing approaches and technology. Chapter 3 defines the concept of social engineering attack and presents our case studies: Website Phishing Experiment and Phishing Website Survey Scenario Experiment, followed by users' reaction analysis. Chapter 3 also states all the phishing website characteristics and features extracted from these different phishing experiments and case studies. Chapter 4 presents our intelligent fuzzy logic phishing detection model with its system design implementation and fuzzy rule-base for all model phases.

Chapter 5 introduces and covers all implemented supervised machine learning associative and classification mining algorithms and techniques used on our intelligent phishing website detection model, to combine and integrate with fuzzy logic inference engine. In this chapter we present our five association and classification algorithms and approaches (JRip, PART, PRISM, C4.5, CBA) with all their parameters and conditions, to be used to automate the classification rule generation for enhancing the performance and precision of the final phishing website detection rate. Followed by the presentation of our intelligent fuzzy-based association classification mining model for phishing website detection which combines fuzzy logic with association and classification algorithms and techniques to automate fuzzy rule generation and reduce the role of human intervention. This chapter shows our intelligence heuristic webpage analysis, our experiential setup for phishing website dataset and extracted features, utilisation of different DM classification algorithms and its rules generation for all criteria and layers related to our intelligent detection model. Chapter 6 is devoted to the practical implementation of the phishing detection model for testing and comparing, demonstrating and analyzing the developed plug-ins intelligent phishing detection toolbar to prove its feasibility, reliability and accuracy comparing to other phishing detection toolbars. The last chapter, Chapter 7, summarizes the main achievements of this thesis, presents the general conclusions and suggests further research directions.

Chapter 2

Literature Review

2.1 Introduction

Phishing websites are a recent problem. Nevertheless, due to their huge impact on the financial and on-line retailing sectors and since preventing such attacks is an important step towards defending against website phishing attacks, there are several promising approaches to this problem and a comprehensive collection of related works. In this section, we briefly survey existing anti-phishing solutions and a list of the related works.

Dhamija and Tygar's (2005) approach involves the use of a so-called dynamic security skin on the user's browser. This technique uses a shared secret image that allows a remote server to prove its identity to a user in a way that supports easy verification by humans but which is difficult for the phishers to spoof. The disadvantage of this approach is that it requires effort by the user. That is, the user needs to be aware of the phishing threat and check for signs that the site he/she is visiting is being spoofed. The proposal approach requires changes to the entire web infrastructure (both servers and clients), so it can succeed only if the entire industry supports it. Also this technique does not provide security for situations where the user login is from a public terminal. More recently, Dhamija et al. (2006) analyzed 200 phishing attacks from the Anti-Phishing

Work Group database and identified several factors, ranging from pure lack of computer system knowledge, to visual deception tricks used by adversaries, due to which users fall for phishing attacks. They further conducted a usability study with 22 participants. The participants were asked to study 20 different websites to see if they could tell whether they were fraudulent or authentic. The result of this study showed that age, sex and computer habits didn't make much difference. They even noticed that pop-up warnings of invalid signature of the sites and visual signs of SSL (Secure Sockets Layer), padlocks etc. were very inefficient and were overlooked. They found that 23% of the participants failed to look at security indicators warning about phishing attacks and, as a result, 40% of the time they were susceptible to a phishing attack. Based on their analysis, the authors suggest that it is important to re-think the design of security systems, particularly by taking usability issues into consideration. Wu et al. (2006a) proposed methods that require web page creators to follow certain rules to create web pages, by adding sensitive information location attributes to HTML code. However, it is difficult to persuade all web page creators to follow the rules.

Liu et al. (2005) analyzed and compared legitimate and phishing web pages to define metrics that can be used to detect a phishing page on visual similarity (i.e. block level similarity, layout similarity and overall style similarity). The DOM -based (Wood, 2005) visual similarity of web pages is oriented, and the concept of visual approach to phishing detection was first introduced. Through this approach, a phishing web page can be detected and reported in an automatic way rather than involving too many human efforts. Their method first decomposes the web pages (in HTML) into salient (visually distinguishable) block regions. The visual similarity between two web pages is then evaluated in three metrics: block level similarity, layout similarity, and overall style similarity, which are based on the matching of the salient block regions. A web page is

classified as a phishing page if its visual similarity value is above a predefined threshold. Fu, et al. (2006) proposed a phishing web page detection method using the EMD-based visual similarity assessment. This approach works at the pixel level of web pages rather than at the text level, which can detect phishing web pages only if they are "visually similar" to the protected ones without considering the similarity of the source codes.

The phishing filter in IE8 is a toolbar approach with more features such as blocking the user's activity on a detected phishing site. The most popular and widely-deployed techniques, however, are based on the use of blacklists of phishing domains that the browser refuses to visit. For example, Microsoft has recently integrated a blacklist-based anti-phishing solution into its Internet Explorer (IE8). The browser queries lists of blacklisted and whitelisted domains from Microsoft servers and makes sure that the user is not accessing any phishing sites. Microsoft's solution is also known to use some heuristics to detect phishing symptoms in web pages (Sharif, 2005). Obviously, to date, the company has not released any detailed public information on how its anti-phishing techniques function.

Chandrasekaran et al. (2006) proposed an approach to classify phishing based on phishing emails' structural properties. 25 features, comprising style markers (e.g. the words suspended, account, and security) and structural attributes, such as the structure of the subject line of the email and the structure of the greeting in the body, were used in the study. 200 emails (100 phishing and 100 legitimate) were tested. Simulated annealing was applied as an algorithm for feature selection. After a feature set was chosen, information gain (IG) was used to rank these features based on their relevance. Thus, they applied one-class SVM to classify phishing emails based on the selected

features. The results demonstrated a detection rate of 95% of phishing emails with a low false positive rate.

Fette et al. (2007) compared a number of commonly-used learning methods through their performance in phishing detection on a past phishing data set, and finally Random Forests were implemented in their algorithm PILFER. The authors claim that the methods can be used in the detection of phishing websites as well. 860 phishing emails and 6950 legitimate emails were tested. The proposed method correctly detected 96% of the phishing emails with a false positive rate of 0.1%. Ten handpicked features were selected for training using a phishing dataset that was collected in 2002 and 2003. As pointed out by the authors themselves, their implementation is not optimal and further work in this area is warranted.

Abu-Nimeh et al. (2007) compared six machine-learning techniques to classify phishing emails. Their phishing corpus consisted of a total of 2889 emails and they used 43 features (variables). They used a bag-of-words as their feature set and the results demonstrated that merely using a spam detection mechanism, i.e. bag-of-words only, achieves high predictive accuracy. However, relying on textual features results in high false positive rates, as phishing emails are very similar to legitimate ones. The studied classifiers could successfully predict more than 92% of the phishing emails.

Pan and Ding (2006) examined the anomalies in web pages, in particular, the discrepancy between a web site's identity and its structural features and HTTP transactions. Herzberg and Gbara (2004) proposed a solution to combine the technique of standard certificates with a visual indication of correct certification; a site-dependent logo indicating that the certificate was valid would be displayed in a trusted credentials area of the browser (Olsen, 2004), (Perez, 2003). Another approach detects certain

common attack instances, such as attacks in which the images are supplied from one domain while the text resides with another domain, and attacks corresponding to misspellings of URLs of common targets (Jakobsson, 2005).

Previous research works on duplicated document detection approaches focus on plain text documents and use pure text features in similarity measure, such as collection statistics (Chowdhury, et al., 2002), syntactic analysis (Broder, et al., 1997), displaying structure (Chen, et al., 2003), (Nanno, et al., 2003), (Yu, et al., 2003), visual-based understanding (Gu, et al., 2002), vector space model (Salton, et al., 1975). Hoad and Zobel have surveyed various methods on plagiarized document detection in (Hoad and Zobel, 2003). However, as (Liu, et al., 2005) demonstrated, pure text features are not sufficient for phishing web page detection since phishing web pages mainly employ visual similarity to scam users (Fu, et al., 2006).

"The Phishing Guide" by Ollmann (2004) gives a detailed understanding of the different techniques often included in phishing attacks. The phenomenon that started as simple emails persuading the receiver to reply with the information the attacker required has evolved into more advanced ways to deceive the victim. Links in email and false advertisements sends the victim to more and more advanced fraudulent websites designed to persuade the victim to type in the information the attacker wants, for example to log into the fraudulent site mimicking the company's original. Ollmann also presents different ways to check whether websites are fraudulent or not. Apart from inspecting whether the visited site really is secure through SSL (Secure Sockets Layer), the user should also check that the certificate added to the website really is from the company it claims to be from and that it is signed by a trusted third party. Focusing more attention on the URL can also often reveal fraudulent sites. There are a number of

ways for the attackers to manipulate the URL to look like the original, and if the users are aware of this they can more easily check the authentication of the visited site. Watson et al. (2005) describe in their *White Paper,* "Know your enemy: Phishing", different real-world phishing attacks collected in German and United Kingdom honeynets. Honeynets are open computer networks designed to collect information about different attacks out in the real world, for further forensic analysis. They noticed that phishing attacks using vulnerable web servers as hosts for predesigned phishing sites are by far the most common, compared to using self-compiled servers. A compromised server is often host for several different phishing sites. These sites are often only active for a few hours or days after being downloaded to the server. PassMark (2005) includes a personalized image in a web page to indicate that the user has set up an account with the site. This approach places the burden on *users* to notice the visual differences between a good site and a phishing site and then to correctly infer that a phishing attack is underway. However, this requires user awareness and prior knowledge. Another approach is two-factor authentication, which ensures that the user not only knows a secret but also presents a security token (FDIC, 2004). However, this approach is a server-side solution. Phishing can still happen on sites that do not support two-factor authentication. Sensitive information that is not related to a specific site, *e.g.*, credit card information and SSN, cannot be protected by this approach either. The PRIME project (Pettersson, et al., 2005) helps users to manage their on-line identity in a more natural and intuitive way using three UI paradigms. It supports drag-and-drop actions for personal information submission. It does not specifically target the phishing problem but its improved user interface could help users correctly manage their on-line information. One potential problem with the PRIME interface is its "Just-In-Time-Click-Through Agreements" (JITCTAs) that is used to generate "small agreements that

are easier for the user to read and process". Users could still ignore the agreements by directly clicking through the "I Agree" button.

APWG provides a solution directory (APWG, 2005) which contains most of the major anti-phishing companies in the world. However, an automatic anti-phishing method is seldom reported. Cyveillance Fraud Management (Kirda and Kruegel, 2005a) uses proprietary Internet monitoring technology to identify phishing-related activity such as suspicious domain registrations, phishing lures, spoofed sites and the post-attack sale of compromised credentials. Others include Internet Identity's Domain Security Audit (Liu, et al., 2005). These approaches are mainly motivated to protect corporations' interests. Nonetheless, they do not directly defend against phishing attacks for users.

Other browser-integrated anti-phishing tools include Google Safe Browsing (Schneider, et al., 2007) and McAfee SiteAdvisor (McAfee SiteAdvisor, 2007). Similar to the Microsoft IE 8 anti-phishing protection, Google Safe Browsing uses blacklists of phishing URLs to identify phishing sites. The disadvantage of the approach is that non-blacklisted phishing sites are not recognized. The success of a blacklist relies on massive amounts of data being collected at frequent intervals. In contrast, SiteAdvisor is a database-backed solution that is, however, mainly designed for protection against malware-based attacks (e.g., Spyware, Trojan horses, etc.). It includes automated crawlers that browse web sites, perform tests and create threat ratings for each visited site. Unfortunately, just like other blacklist or database-based solutions, SiteAdvisor cannot recognize new threats that are unknown and not in the database (Zhang, et al., 2006). Verisign (2005) has also been providing a commercial anti-phishing service. The company is crawling millions of web pages to identify "clones" in order to detect phishing web sites. Furthermore, just like other large companies such as Microsoft,

McAfee and Google, blacklists of phishing websites are maintained. Note that one problem with crawling and blacklists proposals could be that the anti-phishing organisations will find themselves in a race against the attackers. This problem is analogous to the problems faced by anti-virus and anti-spam companies. Obviously, there is always a window of vulnerability during which users are susceptible to attacks. Furthermore, listing approaches are only as effective as the quality of the lists that are maintained. Gabber et al. (1999), present a tool that tries to protect a client's identity and password information. They define client personality in terms of username, password and email address and introduce a function which provides clients with different personalities for the different servers they visit. Chandrasekaran, (2005) proposed inserting intelligent chip to sign as anti-phishing new fighting technique. Chinchani and Upadhyaya (2005) introduced new procedure by stemming software's flaws and improving vigilance with psychological defence, using different logon passwords and payment passwords. Emigh (2006) discussed a wide variety of phishing attacks and countermeasures for the attacks. He also discussed why users are fooled by phishing attacks and the effectiveness of anti-phishing toolbars.

Jakobsson introduced a new model, called a *phishing graph*, to visualize the flow of information in a phishing attack (Jakobsson, 2005). While this model is not, in essence, a defensive technique, it is the first step towards developing an abstract model for visualizing phishing. A phishing graph enhances the ability to analyze and understand the course of a phishing attack. TrustedBrowser (Ye and Smith, 2005) uses a synchronized random coloured boundary to secure the path from users to their browser. The trusted status content is marked in the trusted window whereas the server content is shown in the distrusted window. Anti-Phish (Kirda and Kruegel, 2005b) compares the domains for the same sensitive information in web pages to the domains in the caches.

That is, if it detects that confidential information such as a password is being entered into a form on a distrusted website, a warning is generated and the pending operation is cancelled. PhishHook (Stepp, 2005) converts a web page to “normal form” through text, images and hyperlinks transformations.

PwdHash (Ross, et al., 2005), in contrast, creates domain-specific passwords that are rendered useless if they are submitted to another domain (e.g., a password for www.gmail.com will be different if submitted to www.attacker.com).

The limitation of browser-based schemes is that they require prior knowledge of the target site, which is unfortunately not always available. More importantly, since phishing attackers are able to update the inducement techniques to get around those schemes, the effectiveness of these schemes is not convincing. In a proactive manner, a set of techniques are designed to capture phishing sites on the Internet.

Several commercial and open-source toolbars have been proposed to protect the users from phishing attacks. Most of these techniques perform static checking of the visited web pages and URLs to detect the phishing attacks, as shown in Figure 2.1.

Figure 2.1: Existing security toolbars

Spoofstick (Spoofstick, 2005) is a widely-used tool that performs reverse DNS lookup on the visited website, for the purpose of displaying the IP address of the visited site on the browser's toolbar. Although this information can be used to separate legitimate and masqueraded websites, it still requires a 'human-in-the-loop' to make the actual decision.

NetCraft anti-phishing toolbar (Netcraft toolbar, 2006) employs distributed decision mechanisms that rely on its client's majority vote to infer a website's validity. The websites tagged malicious by its subscribed clients are scrutinized, and the result is disseminated among other subscribing members in the form of blacklists. The approach partially uses a database of sites that are maintained by the company. As this technique relies on users' feedback for its decision-making, it may be subject to increased false positives and denial-of-service (DoS) attacks, since the new phishing sites that are not in the database may not be recognized, especially in cases where a group of hackers maliciously frame a legitimate website as malicious. Also, since the masqueraded websites are short-lived, it is highly unlikely that such responses will be propagated to the clients before their lifetime. The weakness of this approach is its poor scalability and its timeliness.

One of the popular methods of detection is using add-in toolbars for the browser. Chou *et al.* introduced one such tool, SpoofGuard (Chou, et al., 2004), that determines if a web page is legitimate based on a series of domain and URL-based tests. It uses domain names, URLs, links, and images to measure the similarity between a given page and the pages in the caches or histories. It looks for phishing symptoms (e.g., obfuscated URLS) in web pages and raises alerts. The technique examines the downloaded website using

various stateful and stateless evaluations like checking for invalid links, URL obfuscation attempts etc. The major disadvantage with these approaches is that they are susceptible to attacks launched from the compromised legitimate website. Also, in many web-hosting domains the attacker could create a user account with the name *login* and launch a successful phishing attack by hosting the masqueraded page in his domain space, which would typically appear as www.domain.com/login, thereby circumventing the aforementioned approaches. Herzberg and Gbara (2004) proposed TrustBar, a third-party certification solution to phishing. The authors propose creating a Trusted Credentials Area (TCA). The TCA controls a significant area, located at the top of every browser window, and large enough to contain highly visible logos and other graphical icons for credentials identifying a legitimate page. While their solution does not rely on complex security factors, it does not prevent spoofing attacks. Specifically, since the logos of websites do not change, they can be used by an attacker to create a look-alike TCA in a distrusted web page.

It should be emphasized that none of the above defence techniques – blacklist, spoofing detection, password-scrambling, anti-phishing toolbars or spam filters – will completely make phishing attacks impossible to perpetrate. Instead, they provide valuable but scattered roadblocks impeding the attacker.

2.2 Anti-Phishing Technology

2.2.1 Anti-Phishing Overview

Anti-phishing tools provide consumers with a dynamic system of warning and protection against potential phishing attacks, and they also defend the brands of legitimate ISPs and web commerce site developers from being “spoofed” to propagate

scams. Of course, the most important role of an anti-phishing tool is to identify phishing websites in a very accurate way and within an acceptable timescale. Some of these tools provide binary indicators which show whether that site is phishing or not, and that can be implemented by using coloured indicators (green represents a legitimate site, and red represents a positively-identified phishing site). Other tools use a ternary system which means that the site can be phishing, legitimate, or unknown (suspicious), and that can also be implemented by using coloured indicators (green represents a legitimate site, red represents a positively-identified phishing site and a yellow or gray indicator represents an unknown or suspicious site).

Phishing techniques have not only grown in number, but also in sophistication. Phishers might have a lot of approaches and tactics to conduct a well-designed phishing attack. The target of the phishing attacks - consumers of on-line banking and payment services providers - are facing a large amount of financial loss and loss of trust in Internet-based services. There is an urgent need to find solutions to combat phishing attacks. So far, various solutions have been proposed and developed in response to phishing. These solutions target both non-technical and technical problem areas.

2.2.2 Non-Technical Anti-Phishing Solutions

Legislation

Legislation is obviously a direct way to minimize phishing by tracing and arresting phishing criminals. Followed the lead of the US, many countries have enacted laws against suspected phishers, and many phishers have been arrested and prosecuted. There are some problems that reduce the effectiveness of the excising laws Firstly, the phisher

is always hard to trace and catch; a phishing attack can be perpetrated very quickly and, afterwards, the perpetrator can vanish into cyberspace. In addition, the fake websites typically migrate rapidly from one server to another. The average phishing website is online for only about 54 hours (Garera, et al., 2006). The other problem is that many laws are of use only after the damage is done, when a consumer has already been defrauded as a result of the phishing. And before the phishing is perpetrated, it is difficult to define which class of fraud is a crime.

Public Education and Awareness

Generally speaking, the primary advantage for criminals conducting phishing attacks is the public's lack of education and awareness of both the existence of financial crimes targeting Internet users and the policies and procedures of online sites for contacting their customers regarding account information and maintenance issues. Thus, public education and awareness are important factors to counter phishing. As awareness of phishing grows among consumers, the incidences of phishing will shrink to a certain extent.

However, getting rid of phishing through education alone will be very difficult. First of all, there are always new or technology-naive Internet users who do not have any experience, and become victims of phishing. Another aspect is that phishers are getting better and better at mimicking genuine emails and websites; even the security expert may sometimes be fooled (Adida, et al., 2005). Finally, in order to be up to date with the latest phishing techniques, users have to spend a lot of time studying the phenomenon, which is impossible for the majority of Internet users (Binxing and Ruifeng, 2006).

2.2.3 Technical Anti-Phishing Solutions

To combat phishing, security organisations such as APWG and some of the world's leading security companies such as M acAfee and Symantec have proposed many technical anti-phishing solutions. Some desktop protection software already has built-in anti-phishing functions.

Since phishing attacks have many types and variations, and involve lots of attacking techniques, there is no 'silver bullet' to solve all phishing problems. the following sections, we will analyze some aspects of anti-phishing solutions based on the phishing technologies they use. All investigated anti-phishing applications create warnings to inform the user when a website appears to be fraudulent. Most anti-phishing applications are extensions or improvements of an existing web browser. Although a wide range of anti-phishing products are available, most of them are not able to make a decision dynamically on whether the site is in fact phished, giving rise to a large number of false positives.

The most popular techniques are described in detail below with some common anti-phishing examples described afterwards, which use one or more techniques combined with each other:

Blacklist Check

A "blacklist" is a dynamic list of known phishing-sites that is updated frequently with newly reported attacks. The suspicious URL is matched against a list of known phishing sites. This method is susceptible to "zero day attacks". Also, techniques like URL obfuscation and routing through alternate domain names can hinder this method ineffective.

Heuristics

This uses heuristics like domain registration information (owner, age, and country), the number of links to other known-good sites, image-hashing, third-party cookies and user reviews. Most of the heuristics used are subjective and produce a large number of false positives (Chinmay, et al., 2008).

User Rating/Polling

These techniques deem the URL as phished, based on user votes. However, it is ineffective against new phishing attacks and is very subjective.

Third Party Certification Authorities and Reputation Services

This requires an additional interface, which itself is susceptible to phishing.

Using Page Rank Methodology

Page rank can be used to detect a phishing URL (Garera, et al., 2006). However, false positives have been observed in these methods. Also, a website routed through a content distribution network (CDN) would create problems for domain-based checks (Chinmay, et al., 2008).

2.3 Anti-Phishing Security Toolbars

Many proposed anti-phishing solutions use toolbars that show different types of security messages and warnings in the web browser's interface to help users detect phishing sites, such as Spoofguard (Chou, et al., 2004), Trustbar (Herzberg and Gbara, 2004), SpoofStick (Spoofstick, 2005) and Netcraft (Netcraft toolbar, 2006) toolbars. Users are

advised to look at the existing browser security indicators, e.g., the URL displayed in the address bar and the lock icon displayed in the status bar when a connection is SSL-protected. However, controlled user studies have shown that these security indicators are ineffective against high-quality phishing attacks for several reasons (Wu, et al., 2006b):

First, warning indicators located in a peripheral area provide a much weaker signal than the centrally displayed web page and can be easily overwhelmed by convincing web content. Many users rely on the web content to decide if a site is authentic or phishing.

Second, the security-related information shown by the indicators is not really needed for the user's current task. Since security is rarely a user's primary goal, users fail to pay continuous attention to the indicators. Making security a separate task that users are required to remember is not an effective solution.

Third, sloppy but common web practices cause some users to rationalize the violation of the security rules that some indicators use to detect phishing attacks. For example, users are told to examine the hostname displayed in the address bar, to make sure that the hostname is the one they are expecting. But some legitimate websites use IP addresses instead of hostnames (*e.g.*, the Google cache) and some sites use domain names that are totally different from their brand names (Herzberg, 2005).

Fourth, some indicators deliver warnings without detailed, convincing explanations, which makes users think that the software is buggy and thus not treat the warning seriously.

Fifth, although users do notice the system model displayed by the toolbar under phishing attacks, most of them do not have the expertise to correctly interpret it. For

example, they cannot tell the difference between a lock icon displayed on a web page and the one displayed in the status bar. (e.g., amazon.com vs. amazon-department.com) are actually from the same organisation in the real world.

Finally, security indicators tend to show that something is wrong and advise users not to proceed, but they do not suggest good alternatives. This may encourage users to risk submitting their information anyway, since they don't see any other way to accomplish their goal.

2.4 Justification of the Proposed Research

Phishing website attacks are well-organized and financially motivated crimes which steal the user's confidential information and authentication credentials. It damages the confidence in e–commerce as a whole. It is obvious that phishing problems could be a stumbling block, impeding the development of on-line financial services. Current anti-phishing technologies have lots of limitations and constraints and will not completely stop phishing websites. An Artificial Intelligence (AI) heuristic-based search approach can be more appropriate and suitable for phishing website detection. We want to prove the applicability of using fuzzy-based classification mining techniques for building new phishing website detection. Extracted fuzzified phishing website features and patterns can be correctly classified and integrated in a supervised machine-learning solution to identify phishing websites effectively and dynamically.

The intelligent phishing detection system should reduce the requirement for human knowledge intervention for detection of phishing websites and be an alternative solution to the black-list or white-list dependency approach, by adopting new fuzzy-based classification mining technique to detect phishing websites. The proposed solution

should outperform the existing techniques in terms of accuracy, reliability and dependability.

Chapter 3

Social Engineering Phishing Attacks and Experimental Case-Studies

3.1 Introduction

The first step in fighting out phishing is to understand its technique and its methodology. That is why studying and knowing everything about social engineering is very crucial, since we know now that phishing is a social engineering technique employed to deceive users into giving away financial and personal information (Weider, et al., 2008). Implementing and conducting some empirical experiments and case-studies are also very important for studying and analyzing different social engineering phishing attacks in order to help us to design effective countermeasures.

The key element of a social engineering attack is trust - the target trusts the hacker. To resist this form of attack, we need to stimulate a healthy scepticism among staff of anything out of the ordinary and engender their trust in the company IT support infrastructure. We performed different case-study experiments to assess and to evaluate the accuracy and precision of phishing website factors to find the most common phishing clues and indicators that convey authenticity to our employees. Also, one of

the purposes of our experiments was to identify which malicious strategies and attack techniques are successful in deceiving general users, and why.

3.2 What is Social Engineering Phishing Attack?

Social engineering phishing attack is the act of manipulating people into doing what hackers want in order to gain access to information or resources. It's a collection of techniques used to deceive people into performing actions or divulging confidential information (Callow, 2009).

Social engineering phishing attack exploit the credulity, laziness, good manners, or even enthusiasm of people. Therefore it is difficult to defend against a socially-engineered attack, because the targets may not realize that they have been duped, or may prefer not to admit it to other people (Midsize Business Security Guidance, 2006).

It is being predicted that social engineering phishing attacks will be on the rise in the years to come. Billions of dollars are lost every year by corporations and internet users to social engineering attacks, in the process making participants in e-commerce increasingly distrustful. The problem of social engineering attack is that there is no single solution to eliminate it completely, since it deals largely with the human factor.

There are many types of social engineering attacks but, in this chapter, we will only concentrate on one popular type of these attacks which called ″Social Engineering Phishing Attack Using Internet Access".

3.3 The Goals of Social Engineering Phishing Attacks

A social engineering phishing attack attempts to persuade company staff to provide information that will enable him or her to use their systems or system resources to gain unauthorized access to a company's money, information, or IT resources. The social

engineering phisher persuades a staff member to provide information through a believable ruse, rather than infecting a computer with malware through a direct attack. An attack may provide information that will enable the hacker to make a subsequent malware attack (Midsize Business Security Guidance, 2006).

These phisher' goals are based on money, social advancement, and self-worth. Phishers want to take money or resources, they want to be recognized within society or their own peer group, and they want to feel good about themselves.

3.4 Social Engineering Phishing Attack Using Internet Access

Most employees browse the web for personal reasons, such as on-line shopping or research, at some time. Personal browsing may bring employees, and therefore the company computer systems, into contact with generic social engineers who will then use the staff in an effort to gain access to the company resources. The two most common methods of enticing a user to click a button inside a dialog box are by warning of a problem, such as displaying a realistic operating system or application error message, or by offering additional services.

The following Figure 3.1 shows how a hyperlink appears to link to a secure PayPal website (https), while the status bar does not show anything that indicates for sure that it will take the user to a hacker's site. A hacker can suppress or reformat the status bar information.

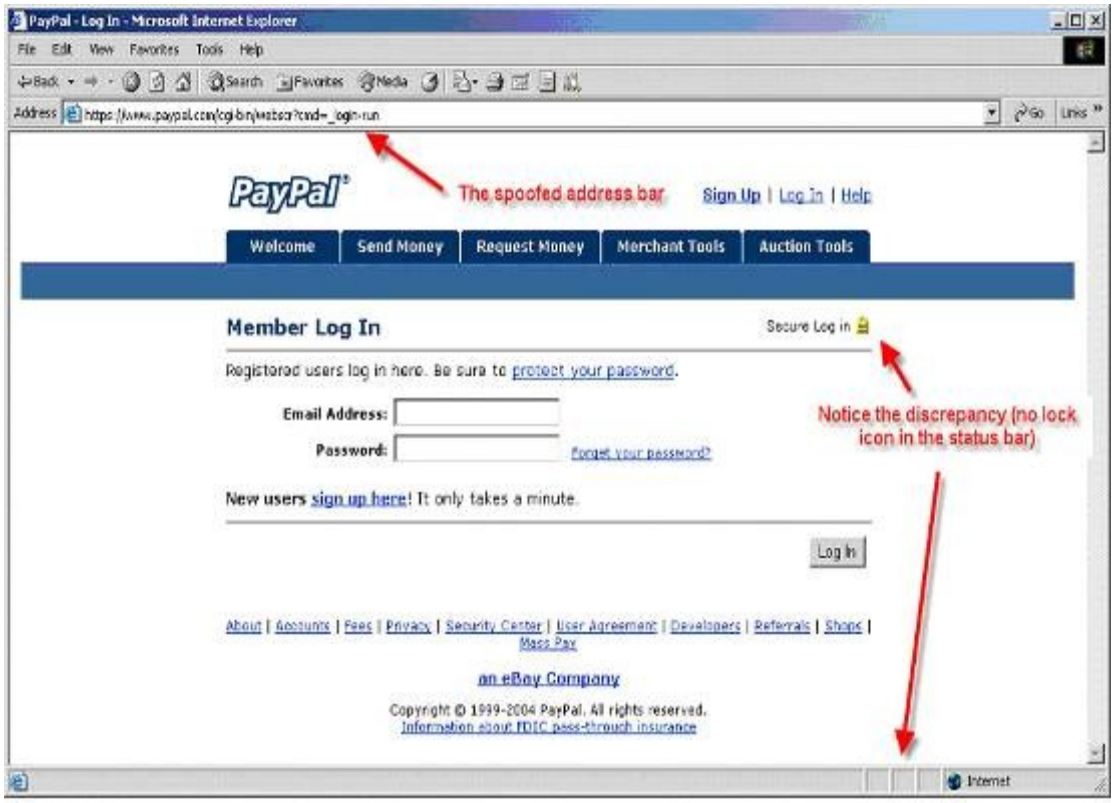


Figure 3.1: Web page phishing hyperlink

3.5 Empirical Phishing Experimental Case-Studies

Conducting different kinds of phishing experiments can shed some light on social engineering attacks, such as phone phishing and phishing website attacks, and can also help us in designing effective countermeasures and analyzing the efficiency of performing training and security awareness about phishing threats (Jakobsson, et al., 2007). The surprising percentages of victims who disclosed their credentials in our phishing experiments underscore the need to redouble our efforts in developing phishing prevention techniques.

3.5.1 Case-Study 1: Website Phishing Experiment

We engineered a website for phishing practice and study. The website was an exact replica of the original Jordan Ahli Bank website www.ahlionline.com.jo , designed to trap users and induce them by targeted phishing emails to submit their credentials (username and password). The specimen was inclusive of our colleagues at Jordan Ahli Bank after attaining the necessary authorizations from our management.

We targeted 120 employees with our deceptive phishing email, informing them that their e-banking accounts were at risk of being hacked and requesting them to log into their account through a fake link attached to our email using their usual customer ID and password to verify their balance and then log out normally.

Deceiving Phishing Email

E-banking Services BES

We have automatically reviewed your accounts recently and we suspect that they were tampered with by an unauthorized third party. Protecting the security of your account and our network is our primary concern. Therefore, as a preventative measure, we have deactivated the services in your account that are liable for breaching and we kindly ask you to thoroughly follow the hereunder procedures to ascertain that your account is intact.

- Login to your Internet Banking account.
- Enter your Customer ID and Password as usual.
- Review your recent account history for any unauthorized withdrawals or deposits. Report to us immediately if you suspect any unauthorized activity has taken place on your account.
- After checking, we will automatically update your account records and reconnect it with the main web server database. Confirmation

message will appear to you after successful update and reactivation of your account.

"Thank you,

Your record has been updated successfully"

- To get started, please click on the link below:

https://www.ahli.com/ahlionline

We apologize for any inconvenience this may cause, and appreciate your assistance in helping us maintain the integrity of the entire e-banking system. Thank you for your prompt attention to this matter.

Sincerely,

Banking Electronic Services Team

The web site successfully attracted 52 out of the 120 targeted employees, representing 44% who interacted positively by following the deceptive instructions and submitting their actual credentials (customer ID, Password).

Surprisingly, IT department employees and IT auditors constituted 8 out of the 120 victims representing 7%, which shocked me, since we expected them to be more alert than others. From other departments, 44 of the 120 targeted employee victims, representing 37%, fell into the trap and submitted their credentials without any hesitation.

The remaining 68 out of 120, representing 56%, were divided as follows: 28 employees (23%) supplied incorrect info, which seems to indicate a wary curiosity; and 40 employees, representing 33%, received the email, but did not respond at all, as shown in Table 3.1.

Table 3.1: Phishing website experiment

Response to Phishing Experiment	Number of Employees
Interacted positively (IT Department)	8
Interacted positively (Other Departments)	44
Interacted negatively (Incorrect info)	28
Interacted negatively (No response)	40
Total	**120**

The results clearly indicate, as shown in Figure 3.2, that the target phishing factor is extremely dangerous since almost half of the employees who responded were victimized, particularly trained employees such as those of the IT Department and IT Auditors.

Increasing the awareness of all users of e-banking regarding this risk factor is highly recommended; this includes customers and employees alike.

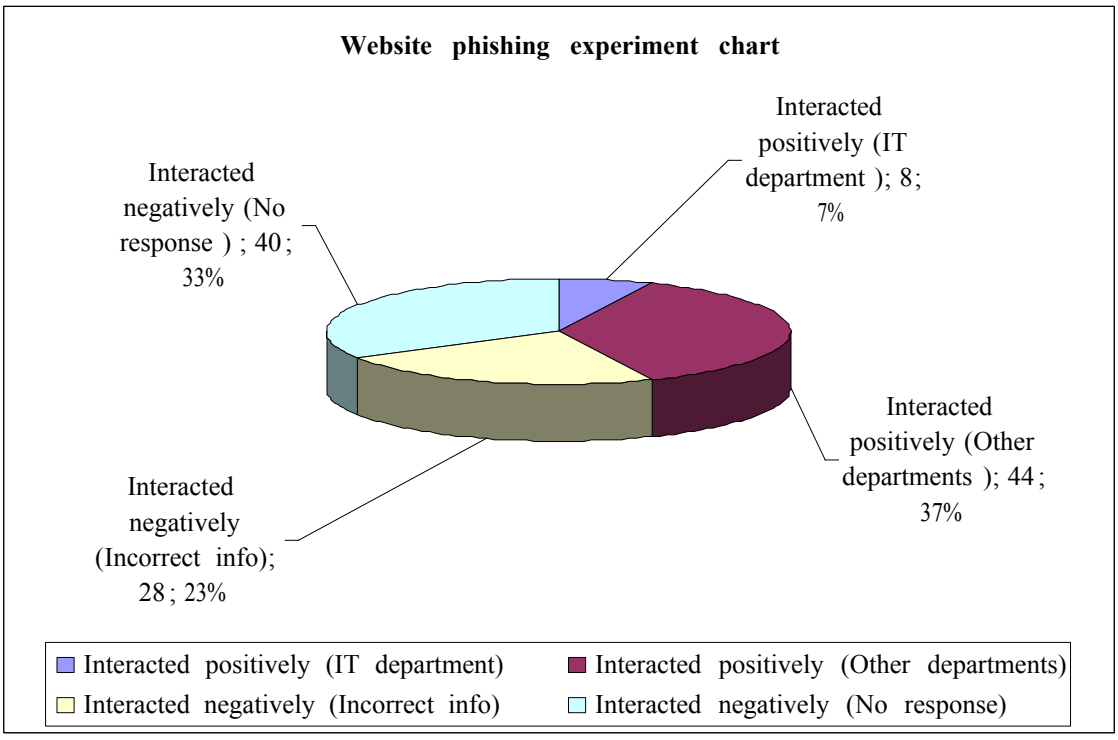


Figure 3.2: Website phishing response chart

3.5.2 Case-Study 2: Phishing Website Survey Scenario Experiment

After the success of our previous phishing website empirical experiment which was conducted at our bank, targeting a specific number of its employees (120), the bank was really interested in studying the vulnerability of their employees towards spear phishing e-banking websites, since targeted spear phishing attacks have always been more successful than generic phishing attacks in conning people and causing financial damage to companies and individuals. We found this a good opportunity to perform a new usability study experiment to assess and to evaluate the accuracy and the precision of our 27 phishing website factors and features, previously collected and analyzed as a result of our cognitive walkthrough of phishing websites' patterns and clues.

This time, we decided to create two groups from our bank employees, each group consisting of 50 participants. In the first group, the employees were totally naïve about the phishing threat and did not have any previous experience or training in dealing with this kind of social engineering phishing attack. Regarding the second group, we decided to choose the 50 employees from our previous 120 employee specimen who had participated in our previous phishing website experiment case, in order to measure and evaluate the effectiveness and the efficiency of prior phishing website awareness training, and past experience of dealing with phishing attack hacking incidents.

In total, our new specimen was 100 bank employees; half of them were untrained (First group) and the second half were trained (Second group).

We analyzed a set of phishing attacks and tricks to measure their effectiveness and influence, and developed 50 phishing and legitimate website survey scenarios which were collected from the APWG's archive (APWG, 2008), and Phishtank archive (Phishtank, 2008). The scenarios analyzed were carried out with the latest scenarios added to the archive by APWG and Phishtank experts. The scenarios were described

and explained in detail in their archives. From these different scenarios, 30 out of the 50 were phishing websites and the rest were legitimate.

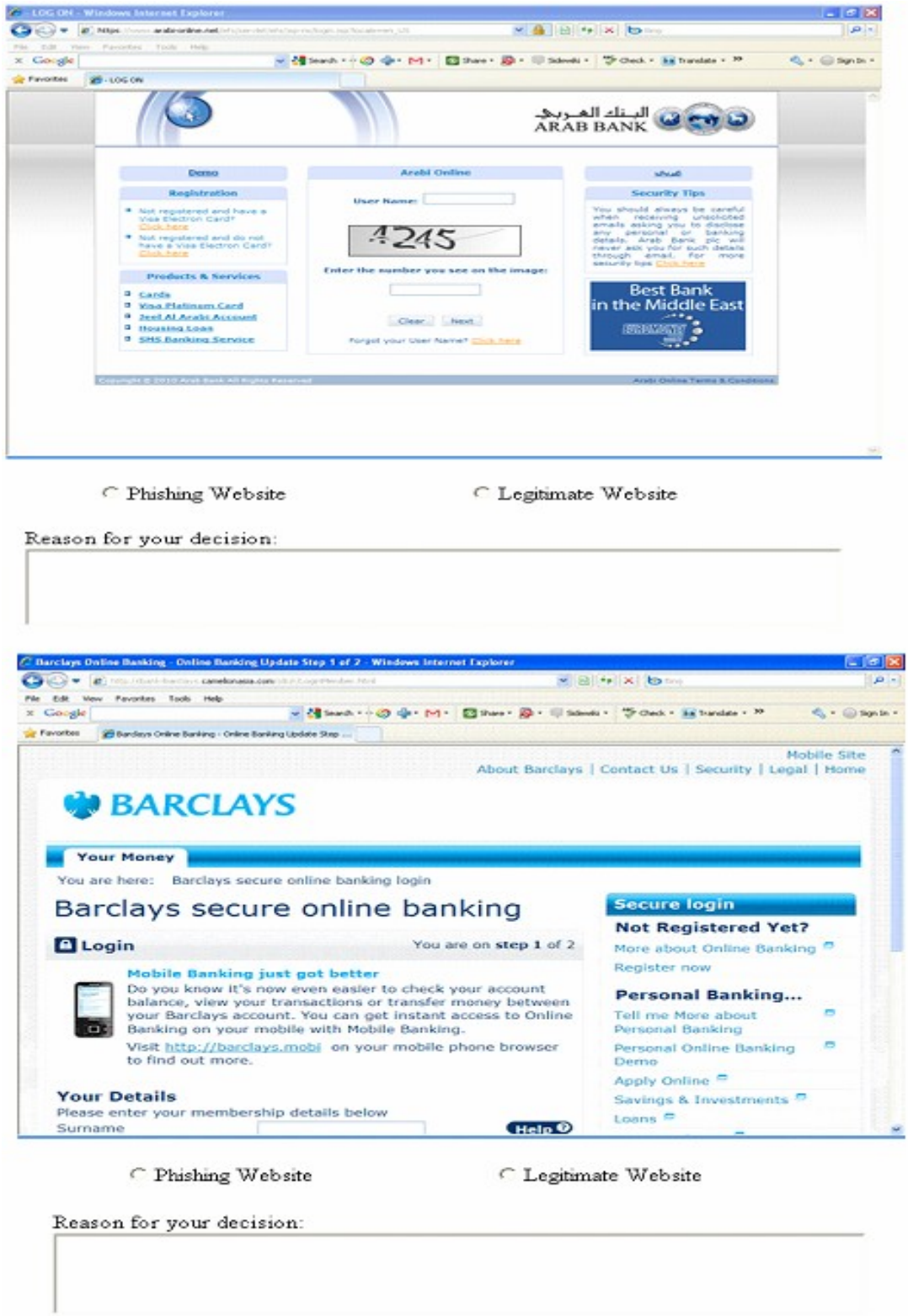


Figure 3.3: An example of phishing website scenario survey

We showed the two participating groups (trained and untrained) the 50 different website scenarios that appear to belong to decent financial institutions and reputable banks, as shown in Figure 3.3, and asked them to determine which ones were fraudulent and which ones were legitimate and to give the reason for their decision and evaluation.

We showed the participants that the purpose of this experiment was to help them discover their knowledge and awareness of the new rising phenomenon of social engineering phishing website attack, and their capability to identify and to distinguish the legitimate genuine website from the phishing spoofed website.

For our part, the purposes of our experiment are to find the most common phishing clues and indicators that appear in the scenarios, to determine what aspects of a website effectively convey authenticity to our employees, and to try to identify which malicious strategies and attack techniques are successful at deceiving general users, and why (Alnajim and Munro, 2008).

From this experiment, we also tried to determine the effectiveness and the value of implementing some security training awareness and phishing courses or classes about phishing threats and detection expertise, and how this might reflect the determination of website legitimacy by the second, trained, group.

Our 27 phishing website factors and features were all deliberately distributed randomly across the 30 phishing website scenarios. One phishing factor could appear in many phishing scenarios and one phishing scenario could have more than one factor or feature. This is illustrated in Table 3.2.

Table 3.2: Phishing factor indicators

Phishing Factor Indicator	No. of Appearance	Appearance Percentage %
Using the IP Address	**14**	**46.66**
Abnormal Request URL	**30**	**100**
Abnormal URL of Anchor	**7**	**23.33**
Abnormal DNS Record	**2**	**06.66**
Abnormal URL	**5**	**16.66**
Using SSL Certificate	**17**	**56.66**
Certification Authority	**4**	**13.33**
Abnormal Cookie	**2**	**06.66**
Distinguished Names Certificate(DN)	**4**	**13.33**
Redirect Pages	**3**	**10.00**
Straddling Attack	**2**	**06.66**
Pharming Attack	**4**	**13.33**
Using onMouseOver to Hide the Link	**6**	**20.00**
Server Form Handler (SFH)	**2**	**06.66**
Spelling Errors	**24**	**80.00**
Copying Website	**5**	**16.66**
Using Forms with "*Submit*" Button	**6**	**20.00**
Using Pop-Ups Windows	**8**	**26.66**
Disabling Right-Click	**2**	**06.66**
Long URL Address	**22**	**73.33**
Replacing Similar Characters for URL	**16**	**53.33**
Adding Prefix or Suffix	**9**	**30.00**
Using the @ Symbol to Confuse	**6**	**20.00**
Using Hexadecimal Character Codes	**8**	**26.66**
Much Emphasis on Security and Response	**5**	**16.66**
Public Generic Salutation	**12**	**40.00**
Buying Time to Access Accounts	**3**	**10.00**

As Table 3.2 presents, the phishing factor indicator ARUL "Abnormal Request URL" appeared in all 30 of the phishing scenarios. Furthermore, the phishing factor indicator, "Spelling Error", appeared in 80% of the phishing scenarios (24 appearances). In contrast, phishing factors such as "Abnormal DNS Record" and "Disabling Right Click"

have the fewest appearances (6.66 %, representing 2 appearances). We made sure that each phishing factor indicator had appeared at least once in the phishing website scenarios.

The result from this experiment was very interesting. As shown in Table 3.3, in the first, untrained, group we found 72% of their decisions were wrong regarding the legitimacy of the websites presented to them in the experiment. These results were represented by either False Positive Case (FP, 38%), which happens when a legitimate website is considered as phishing by the participant, or by False Negative (FN, 34%), which happens when a phishing website is considered legitimate by the participant. Just 28% of their decisions were right regarding the legitimacy of the website, represented by either True Positive Case (TP, 11%) , which happens when a legitimate website is considered legitimate by the participant, or by True Negative (TN, 17%), which happens when a phishing website is considered as phishing by the participant. Figure 3.4 represents the column chart for website legitimacy decisions for the first, untrained, group.

We found that most of these wrong decisions made by first, untrained, group arose from their lack of knowledge and awareness of the most common phishing website tricks and deceptions. Most of them did not pay attention at all to some very obvious phishing clues or indications like address bar contents, URL, domain name, page style, page contents and security indicators like SSL certificate or logos, leading to this high incorrect decision percentage. Most of their decisions and judgements concentrated on the look of the website and its fancy colours, pictures and animation style, thus supporting the arguments mentioned by Dhamija, et al., (2006).

Table 3.3: The results of website legitimacy decisions for the first group (Untrained group)

Decision / Website Legitimacy	True	False
Positive	**TP (11%) 275 Decision**	**FP (38%) 950 Decision**
Negative	**TN (17%) 425 Decision**	**FN (34%) 850 Decision**

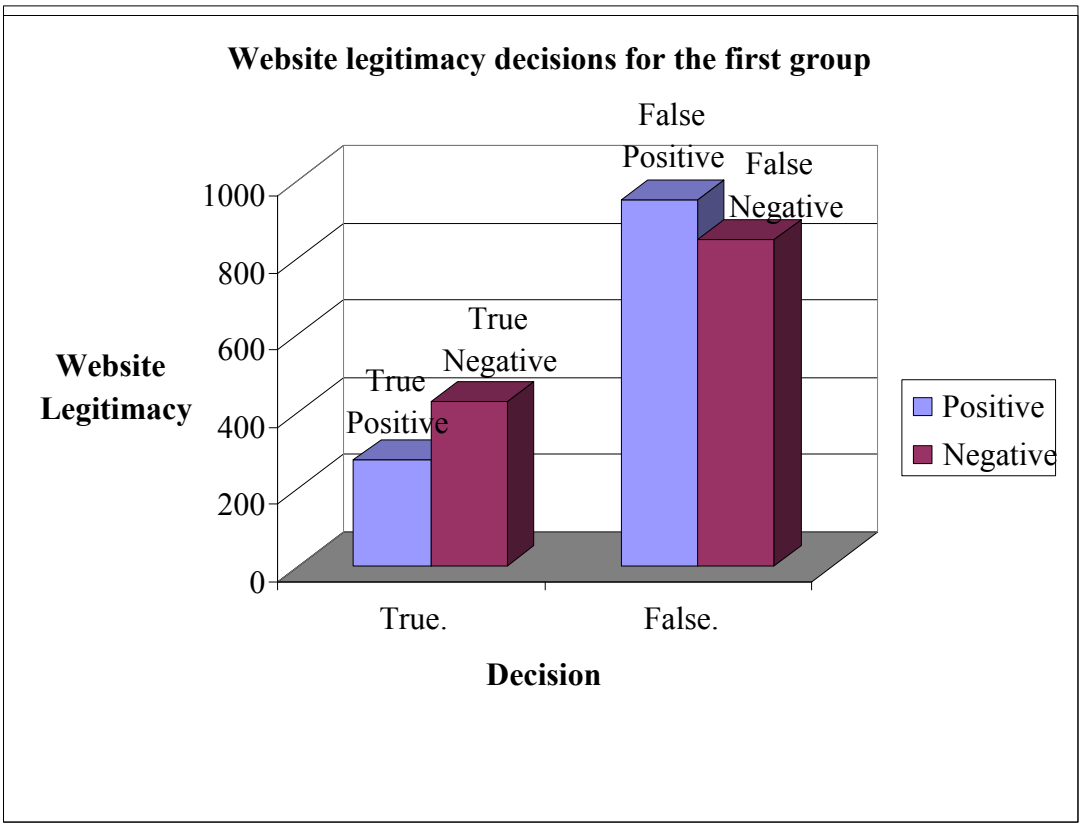


Figure 3.4: Website legitimacy decisions chart for first group

Regarding the second, trained, group, the results were totally different. Their previous experience of the phishing website experiment and the skills they gained from that were very obvious, and played a big role in the total outcomes.

As shown in Table 3.4, from the second, trained, group we found 72% of their decisions were right regarding the legitimacy of the website, represented by either True Positive Case (TP, 39%) or by True Negative (TN, 33%). Just 28% of their decisions were wrong regarding the legitimacy of the websites presented to them in the experiment. These results were represented by either False Positive Case (FP, 12%) or by False Negative (FN, 16%). Figure 3.5 represents the column chart for website legitimacy decisions by the second, trained, group.

We found that most of these correct decisions made by the second, trained, group resulted from their good experience, knowledge and awareness of the most common phishing website tricks and deception attacks that they had faced before. Most of them depended on their judgment and assessment of the website address bar, URL domain name and the different security indicators. They were not fooled by the design, style or fancy look of the website structure or animation, and their main concentration was focused on detecting all phishing website factor indicators, which led to this acceptable correct decision percentage. This of course suggests the importance of conducting phishing training awareness for all users.

Table 3.4: The results of website legitimacy decisions for the second group (Trained group)

Decision **Website Legitimacy**	**True**	**False**
Positive	**TP (39%)** **975 Decision**	**FP (12%)** **300 Decision**
Negative	**TN (33%)** **825 Decision**	**FN (16%)** **400 Decision**

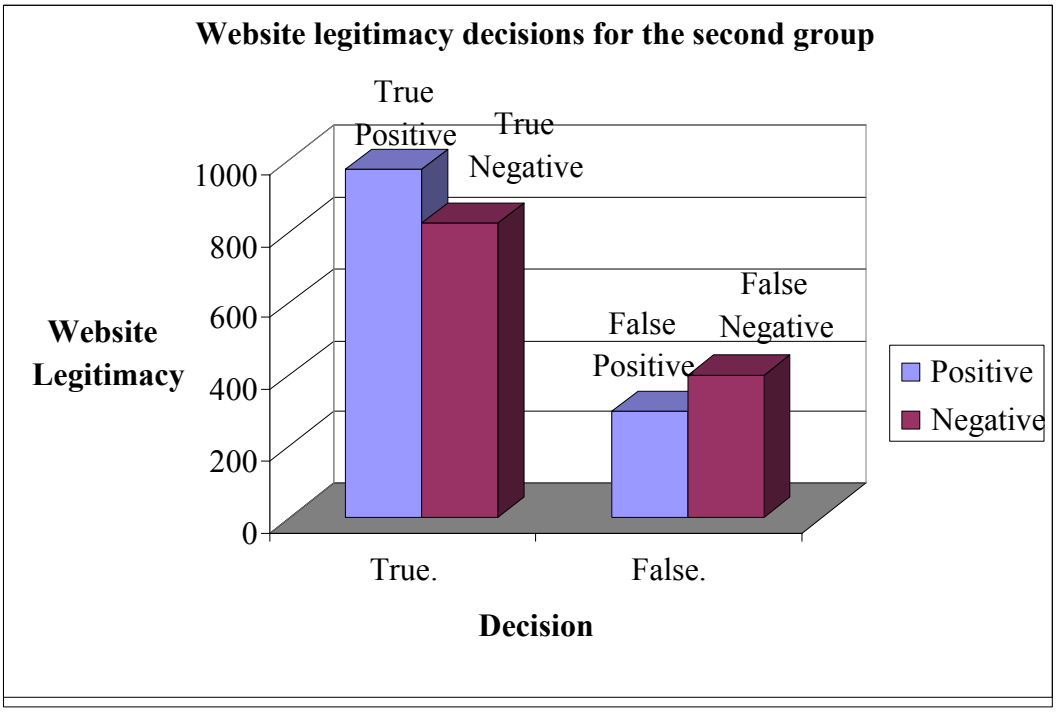


Figure 3.5: Website legitimacy decisions chart for second group

Nevertheless, still some expert employees of the second trained group did not took the right decision for some of phishing or legitimate websites, and they were fooled for some visual deception phishing attacks. These results illustrate that traditional standard security phishing factor indicators are not effective enough for detecting phishing website, and suggest that alternative intelligent approaches are needed.

3.6 Reactions Analysis to Website Phishing Experiment

43% of the employees saw the learning value of the experience, and appreciated the insights they had gained as a result of being part of the study. The rest of the employees felt that the study had no value, and felt violated at not having been asked permission before the experiment was performed. They called the experiment unethical, inappropriate, illegal and unprofessional. These reactions highlight that phishing has a significant psychological cost for victims. 75% of the employees stated that they did not

and would never fall for such an attack. This natural denial reaction suggests that we may find it hard to admit to our own vulnerability. As a consequence, many successful phishing attacks may go unreported, meaning that phishing success rates in surveys may be severely underestimated. Phishers know that most users don't know how to check the security and often assume that sites requesting sensitive information are secured. When users don't know how secure they are, they assume that they *are* secured, and it's not easy for them to see the difference between authentic security and mimicked security features. We found that security is often a secondary goal for most of our employees. They did not look at the address bar, status bar, or certificate authority. They often focus on their major tasks, and neglect all other security pointers or warning messages. We found that 48% of the employees were fooled by the presence of an SSL closed padlock icon appearing within the body of a web page instead of looking for it in the right place. Many employees always looked for a certain type of content like the closed padlock icon when making their judgment and never mentioned the other security features like the characters and numbers shown in the address bar, the certificate authority or any other factors whatsoever. 37% of the employees did not look for any SSL signs that can distinguish the secured encrypted website from the non-secured one, such as observing the "HTTPS" in the address bar. 27% of the employees had some reservations when they saw an IP address instead of a domain name and they were able to distinguish between them. On the other hand, 66% of the employees did not know what an IP address is!

42% of our employees did not check the certificate that was presented to their browser in our study since they do not know what it means; those that *do* know occasionally check them out. 30% of the employees pointed out that the content details of the website and its fancy design and style were one of the main reasons for their opinion

about the legitimacy of the website. They assumed that the site would be legitimate if it contained high quality images and lots of animations. 45% of the employees who clicked on the forged VeriSign logo that we created did not compare the URL displayed in the faked pop-up window, which shows the SSL certificate status of www.ahlionline.com.jo hosted at VeriSign, to the URL in the address bar to detect whether they are referring to the same website. Unfortunately, any site can provide a link to this pop-up page in order to gain credibility (Jagatic, et al., 2007). We found 67% of our employees do not know how to check or locate the self-signed certificate, and they have never checked a certificate before. We also found that visual deception attacks can fool even the most sophisticated users.

As a conclusion, our employees made incorrect decisions about the legitimacy of the e-banking website because of their lack of knowledge and understanding of the phishing techniques and its malicious methods and indicators.

3.7 Approaches to Quantify Website Phishing Problems

In our research, we used three primary approaches to quantify the phishing website problems:

1- Using questionnaire, inspection, examination, investigation and survey in order to quantify the problem and all its characteristics and factors.

2- Performing lab experiments. This approach also covers common trials, which allows the evaluation of the new and expected attacks and suitable countermeasures.

3- Performing experiments that mimic real website phishing attacks; this creates a tricky ethical issue for the researcher, since measuring the actual success rates can only be done by making sure that the study cannot be distinguished from reality.

Chapter 4

Fuzzy Logic Model for Phishing Website Detection

4.1 Introduction

Fuzzy logic is a powerful tool for defining systems through a natural use of language. It connects knowledge discovery mechanisms that automatically isolate and generate rules covering both relationships in the data as well as the processes that connect these relationships; in a way, it can be used to optimize our e-banking phishing detection system. In the next section we will propose a novel framework for using fuzzy logic for modelling phishing website detection system.

4.2 Proposed Model for Phishing Website Detection Using Fuzzy Logic

Results of phishing website detection risk rate are usually qualified with a statement of uncertainties. This work presents a novel approach to overcome the 'fuzziness' in phishing website detection by using fuzzy logic. Fuzzy logic is used to characterize the phishing website factors and indicators as fuzzy variable, which determines the

likelihood of phishing. After that, fuzzy set operations are used to combine the severity of these indicators and likelihood of occurrence to calculate and detect phishing website risk probability. Phishing website risk rate detection is an "assessment" of something hypothetical, defined as "phishing website risk", which must then be interpreted as "phishy", "suspicious" or "legitimate".

Fuzzy logic variables and fuzzy set operations enable characterization of vaguely defined (or fuzzy) sets of likelihood and consequence, impact the mathematics, to combine them using expert knowledge, to detect phishing websites. The fuzzy phishing website risk rate approach presented in this chapter is the first of its kind.

4.3 Collected Phishing Websites' Features and Patterns

From our background phishing knowledge experience and the vast knowledge we gained from conducting a series of phishing experiments with case-studies and surveys for analyzing anti-phishing techniques and solutions, we managed to collect 27 main phishing website features and characteristics that can help us to differentiate the phishing website from the legitimate one.

The list below demonstrates all our 27 collected phishing website features, which will be used later in the methodology study analysis for our fuzzy-based phishing detection model.

1. **Spelling errors**: Most phishing websites have errors in spelling and grammar since they are created on a temporary basis and the phishers are always in a hurry. Increased number of spelling errors could be a sign of phishing website.
2. **Long URL address**: A website with a short URL address is more reliable and trustworthy than a website with suspiciously long URL address. For example,

this website with a short URL address : http://www.ahli.com is more reliable than this suspicious URL address:

http://www.boj.View?DocId=Index&siteId=AC&langID=EN

3. **Emphasis on security and quick response:** Some phishers can defraud and lure many visitors into using their forged websites by emphasizing on the security issue to gain their trust and by always asking for their prompt action to protect their personal information from being hacked.
4. **Personalization vs. public generic salutation:** Personalization increases the trustworthiness of the website, and the more personal information present on the website the more likely it is to be legitimate; vice versa, the more generic and public the information on the website the more suspicious the website is. (Example of generic salutation: Dear customer or Dear member).
5. **Using SSL certificate and padlock icons:** The website with the secured encryption transaction SSL certificate (https ://) is more trustworthy and reliable than the unsecured website (http ://) since most of the forged phishing websites don't use this feature for many reasons. Using SSL can be distinguished by looking for the padlock at the bottom of a browser frame.
6. **Certification authority**: Mouse-over reveals a made-up certification authority and digital signature.
7. **Replacing similar characters for URL and registered domains**: Transforming the real URLs by replacing characters such as an uppercase "I" with a lowercase "L" or the number "1" — transforming WWW.CITI BANK.COM to WWW.C1T1BANK.COM , for instance; and also registering a domain name very similar to the original, owned by a reputable company.

8. **Adding a prefix or suffix**: Add a prefix or suffix to the real domain name, as with www.online-citibank.com or www.citibank-card.com. Here for example, we can see the prefix word "online" before the legitimate Citi Bank domain name, www.citibank.com, in order to confuse the user.
9. **Redirect pages**: Utilise programming bugs in real websites to redirect to other pages — for example, the Citibank site used to include a script that could redirect users to any site specified in place of PHISHING LINK in the URL http://citibank.com/ws/citibankISAPI.dll?MfcISAPICommand=RedirectToDomain&DomainUrl=PHISHINGLINK.
10. **Straddling attack**. The phishermen insert spiteful data in the HTML code of the long-range web page. When the web page is downloaded, the script inside will be executed.
11. **Pharming Attack**. The fishermen amend users' HOST files to shine upon the domain name which is often visited and spurious IP address by cockhorse procedure, spy software, browser hijacking and so on. So the users will joint trap website though having imported correct domain name.
12. **Copying website**: Copying the content of an official web page and imitating its whole style and contents.
13. **Using forms with "*Submit*" button**: Generally, the "Submit" button at the bottom of the form causes the information to be sent to the fraudster's specified location. <FORM action=http://www.citibank-offer.com/sendmail.php method=get target=_blank>
14. **Using onMouseOver to hide the Link:** Using JavaScript event handler "onMouseOver" to show a false URL in the status bar.

15. **Using the IP Address:** Fraudsters attempt to conceal the destination website by obscuring the URL. One method of concealing the destination is to use the IP address of the website, rather than the hostname. Here is an example of an IP address used in a fraudulent website: http://210.14.228.66/sr/.

16. **Using the @ Symbol to Confuse:** When the 'at' symbol (@) is used in an "http://" or "https://" URL, all text before the @ symbol is ignored and the browser references only the information following the @ symbol. In other words, if the format <*userinfo*>@<*host*> is used, the browser is directed to the <*host*> site and the <*userinfo*> is ignored. To further conceal the URL, the @ symbol can be represented by its hexadecimal character code "%40."

17. **Using Hexadecimal Character Codes:** Fraudsters can also hide URLs by using hexadecimal character codes to represent the numbers in the IP address. Each hexadecimal character code begins with "%." For example: http://www.visa.com%00@%32%32%30%2E%36%38%2E%32%31%34%2E%32%31%33. The URL is put in <*userinfo*><*null*>@<*host*> format.

18. **Using Pop-Ups windows**: Many fraudulent web pages are opened as pop-ups which redirect the main browser window to the real company site. This transaction appears to the user as a pop-up over the real company site. Fraudsters use this technique to make their information-gathering appear more credible. Some fraudsters use JavaScript to reopen the fraudulent pop-ups, if closed, until the user fills out the requested information.

19. **Disabling Right-Click:** Using JavaScript to disable the right-click function, which prevents the user from viewing and saving the source code. Sometimes the right-click function is also disabled on fraudulent web pages that are opened in the menu browser window. The following is JavaScript taken from a

fraudulent PayPal website. Function click() { if (event.button==2) { alert('WARNING ! © Copyright 1999-2004 PayPal. All Rights Reserved.')}}.

20. **Buying Time to Access Accounts:** Fraudsters try to buy some time before their victims check on their accounts to give the fraudsters an opportunity to use the personal information they have acquired. The scammers indicate in the web pages that it will take a certain amount of time for the account to be updated. They hope that this will prevent their victims from checking their accounts during this time period.

21. **Abnormal Request URL (RURL):** External objects (such as images, css, and external scripts) in a web page are loaded from another URL. For a normal corporate website, a large percentage of those URLs are in its own domain.

22. **Abnormal URL of Anchor (AURL):** A web page is suspicious when the domains of most of the AURLs are different from the page's domain, or anchors do not link to any page. A high proportion of anchors in a legitimate website point to the same domain as the page itself. (for example: <a href=http://www.citibank.com/>).

23. **Abnormal DNS record**: A full DNS record usually has identified relevant information. For phishing sites, either the record of the host name is not found in the WHOIS database, or the claimed identity is not contained in the record.

24. **Abnormal URL:** The host name in URL does not match its claimed identity (a URL is unique in cyberspace. For a regular website, its identity is usually part of its URL).

25. **Server Form Handler (SFH):** Most e-banking websites usually contain a server form handler. For phishing sites, the SFHs are usually a void ("about: blank" or " ") or refer to a different domain.

26. **Abnormal Cookie:** In a phishing site, its cookies either point to its own domain, which is inconsistent with the claimed identity, or point to the real site, which is inconsistent with its own domain.

27. **Distinguished Names (DN) Certificate:** In many phishing attacks, the Distinguished Names (DN) in their certificates is inconsistent with the claimed identities.

4.4 The Phishing Website Detection Design Methodology

The technique of the model involves the fuzzification of input variables that is based on the 27 phishing website characteristics and factors (previously extracted from our implemented phishing website case-studies experiments, anti-phishing tools and surveys which are mentioned and analyzed in Chapter 3) , rule evaluation, aggregation of the rule outputs, and defuzzification technique as shown in Figure 4.1.

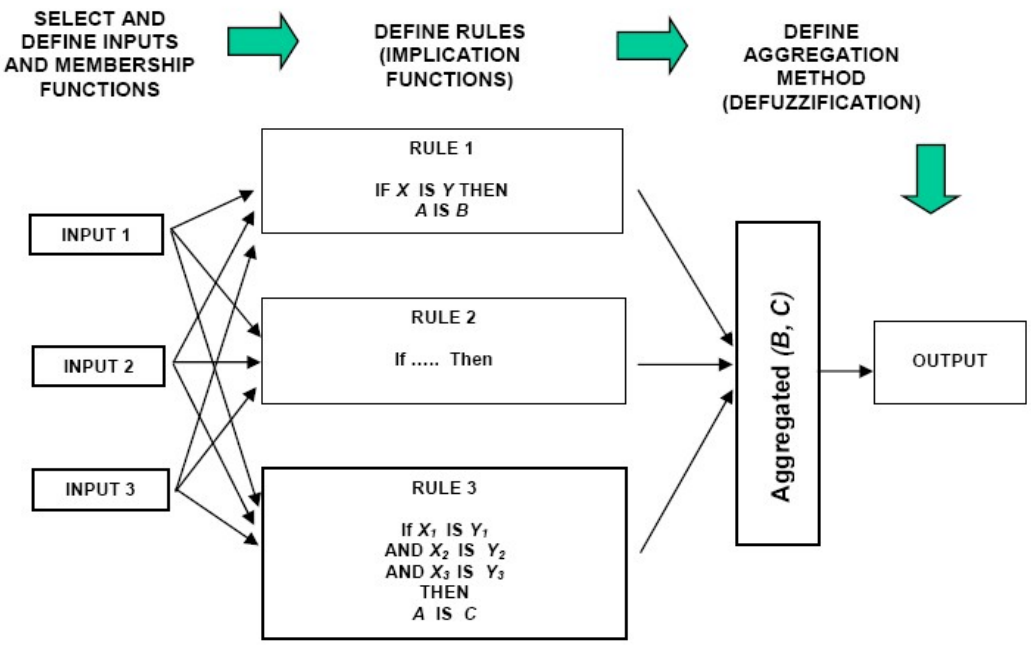


Figure 4.1: The four steps of inference fuzzy system

The model consists of four phases. We will explain each phase in more detail to fully understand its function and output, and how it is connected with the other phases, to produce the final desired output.

4.4.1 Fuzzification

This is the process of generating membership values for a fuzzy variable using membership functions. The first step is to take the crisp inputs from the 27 characteristics and factors which stamp the forged phishing website and determine the degree to which these inputs belong to each appropriate fuzzy set. This crisp input is always a numeric value limited to the universe of discourse. Once the crisp inputs are obtained, they are fuzzified against the appropriate linguistic fuzzy sets. The fuzzy detection model provides more thorough definitions for each factor and its interactions with other factors. This approach will provide a decision tool for identifying phishing websites.

The essential advantage offered by fuzzy logic techniques is the use of linguistic variables to represent key phishing characteristic indicators and the relation of phishing website probability. In this step, linguistic descriptors such as High, Low, and Medium are assigned to a range of values for each key phishing characteristic indicator. Since these descriptors will form the basis for capturing expert inputs based on the impact of Key Phishing Characteristic Indicators on the Phishing Website, it is important to calibrate them to how they are commonly interpreted by the experts providing input.

The valid ranges of inputs are considered and divided into classes, or fuzzy sets. For example, length of URL address can range from ‘low’ to ‘high’ with other values in

between. 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]. Linguistic values are assigned to each phishing indicator as *Low, Moderate and High* and for phishing website risk rate as *Legitimate, Suspicious and Phishy* (triangular and trapezoidal membership function). For each input the values range from 0 to 10 while, for output, they range from 0 to 100.

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 4.2 below. The x-axis in each plot represents the range of possible values for the corresponding key phishing characteristic indicators (*Low, Moderate and High*). The y-axis represents the degree to which a value for the key phishing characteristic indicators is represented by the linguistic descriptor.

For example, and as we can see in the plot of the membership function for URL Address Length, 4.5 cm is considered 'Low' with a membership of 30% and is also considered 'Moderate' with a membership of 65%. The fact that 4.5 cm URL Address Length is considered both Low and Moderate to varying degrees is a distinguishing feature of fuzzy logic, as opposed to binary logic which artificially imposes black-and-white constraints. The fuzzy representation more closely matches human cognition, thereby facilitating expert input and more reliably representing experts' understanding of underlying dynamics (Bridges and Vaughn, 2001).

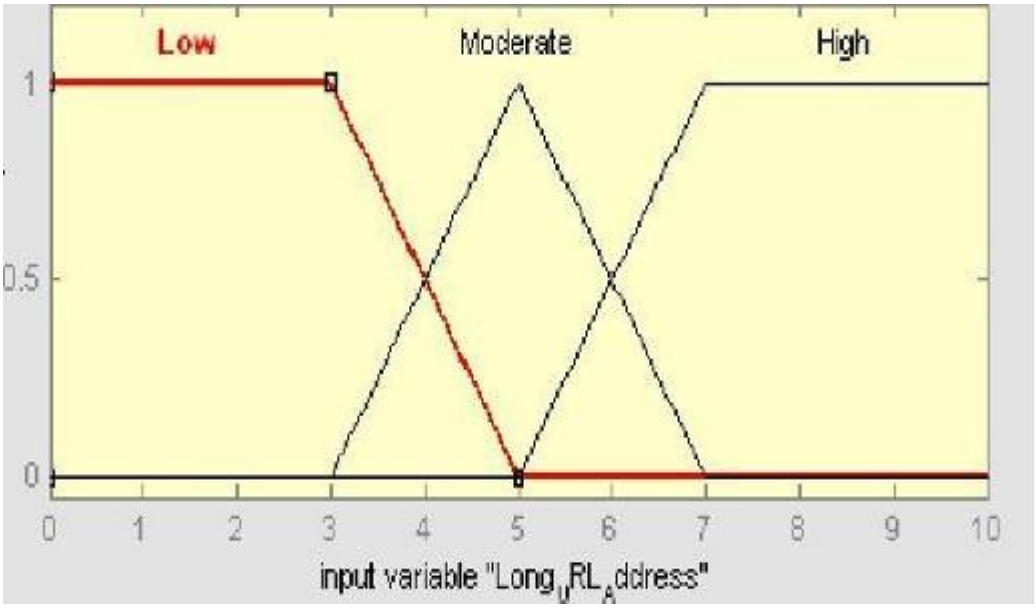


Figure 4.2: Input variable for URL Address Length component

URL Address Length – Low, Moderate, High.
***Linguistic Variable*: URL Address Length**

Linguistic value	Numerical Range
Low	[0, 0, 3, 5]
Moderate	[3, 5, 7]
High	[5, 7, 10, 10]

Another example of the linguistic descriptors used to represent key phishing characteristic indicators is the Pop-Up Windows feature. If the website has two hyperlinks with pop-up windows asking for user credentials, then it is considered 'Low' with a membership of 50 % and is also considered 'Moderate' with a membership of 50%.

Pop-Up Windows – Low, Moderate, High.
***Linguistic Variable*: Pop-Up Windows**

Linguistic value	Numerical Range
Low	[0, 0, 1, 3]
Moderate	[1, 3, 5]
High	[3, 5, 10, 10]

The ranges for this fuzzy variable were specified depending on the high risks that accompany this particular phishing feature. We cannot allow for too many pop-up windows asking for vital information that can be used for phishing purposes. That's why we decide to put a very small fuzzy set range for fuzzy values "Low" and "Moderate" to mitigate these kinds of phishing risks.

The same approach is used to calibrate all the other key phishing website characteristic indicators. The ranges of their fuzzy variables are derived and tuned from a series of phishing experiments with case-studies, surveys and expert knowledge.

4.4.2 Fuzzy Rule Evaluation

This is the second step where the fuzzified inputs are applied to the antecedents of the fuzzy rules. Since the fuzzy rule has multiple antecedents, the fuzzy operator (AND or OR) is used to obtain a single number that represents the result of the antecedent evaluation. We apply the AND fuzzy operation (intersection) to evaluate the conjunction of the rule antecedents.

Having specified the risk associated with the phishing website and its key phishing characteristic indicators, the next logical step is to specify how the phishing website probability varies as a function of the Key Phishing Characteristic Indicators. Experts provide fuzzy rules in the form of ***if…then*** statements that relate phishing website probability to various levels of key phishing characteristic indicators based on their knowledge and experience.

Phishing website experiments, anti-phishing tool analysis, web surveys, and detailed phishing questionnaires were used to find and evaluate all factors and features of

phishing websites, with all their relationships and associations with one another. This helped us greatly as experts in creating the phishing website fuzzy rules.

4.4.3 Aggregation of the Rule Outputs

This is the process of unification of the outputs of all the rules. In other words, we are combining the membership functions of all the rules' consequents previously scaled into single fuzzy sets (output). Thus, input of the aggregation process is the list of scaled consequent membership functions, and the output is one fuzzy set for each output variable.

4.4.4 Defuzzification

This is the last step in the fuzzy inference process, where a fuzzy output of a fuzzy inference system is transformed into a crisp output. Fuzziness helps to evaluate the rules, but the final output of this system has to be a crisp number. The input for the defuzzification process is the aggregate output fuzzy set and the output is a number. This step was done using the Centroid technique because it is the most commonly-used method of defuzzification (Cox, 2001).

The output is the phishing website risk rate and is defined in fuzzy sets like '**phishy**' to '**legitimate'**. The fuzzy output set is then defuzzified to arrive at a scalar value as shown in Figure 4.3.

***Linguistic Variable*: Phishing Website Risk Rate**

Linguistic value	Numerical Range
Legitimate	[0, 0, 30, 50]
Suspicious	[30, 50, 70]
Phishy	[50, 70, 100]

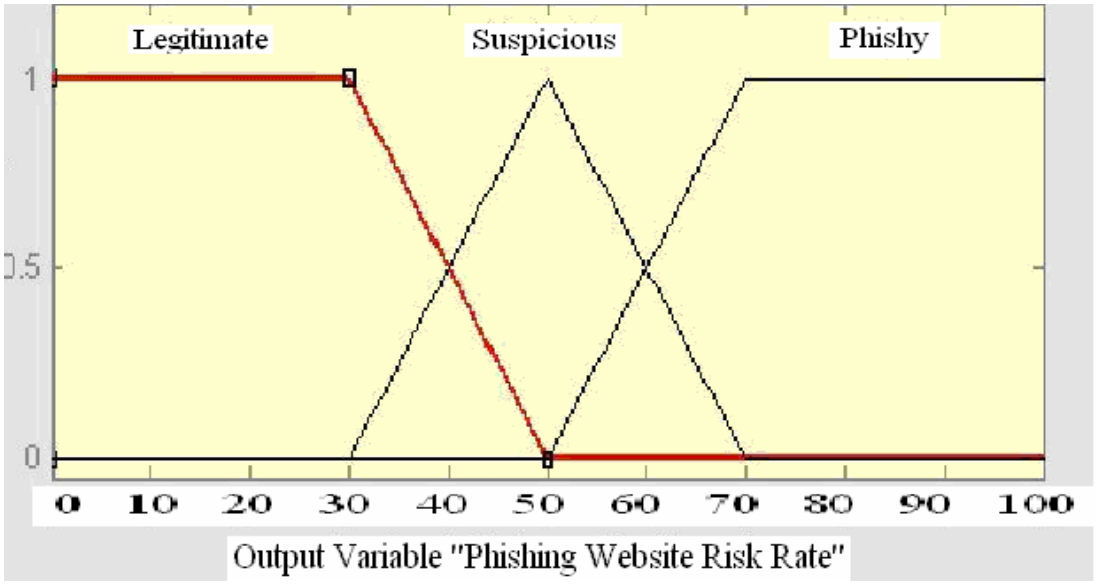


Figure 4.3: Output variable for phishing website rate

***Phishy*:** High guarantee that the website is a forged phishing website, which will hack all user personal information and passwords, with dangerous or catastrophic consequences.

***Suspicious*:** There is reasonable doubt about the legitimacy of the website and there should be some kind of caution in dealing with this website, because it could have risky consequences.

***Legitimate*:** High guarantee that the website is a legal, genuine website, and there is no reason to say otherwise. It can be used safely.

4.5 Fuzzy Logic Phishing Detection Model

In this phishing fuzzy model, we categorize the 27 phishing website characteristics and factors into six different criteria based on their attack type and source. After that, we ranked and weighted the characteristic features in each criteria based on their importance, influence, effectiveness and complexity before considering those in the fuzzy learning process. We have undertaken the grouping process to simplify the fuzzy model since dealing with the 27 website phishing features as a whole can make the fuzzy rule evaluation very complicated and time-consuming.

We grouped and categorized these 27 phishing website features and factors into six criteria (URL & Domain Identity, Security & Encryption, Source Code & Java script, Page Style & Contents, Web Address Bar and Social Human Factor). Each criterion has its own fitted phishing feature criteria. A layering process was also implemented in these phishing website features to enhance and improve the final phishing website risk rate fuzzy output. Table 4.1 represents detailed information on grouping the phishing website features into specific criteria and their association-related layers based on the types of phishing source and nature. The weights assigned to those are according to their effectiveness and influence.

The architecture of the fuzzy logic inference-based phishing website risk rate detection model is shown in Figure 4.4. As can be shown from the structure figure, the final output website phishing result for this fuzzy model depends on evaluating the fuzzy outputs of the three layers and then combining those for the final result.

Table 4.1: Components and layers of phishing website criteria

Criteria	No.	Component	Layer No.
URL & Domain Identity (Weight = 0.3)	1	Using the IP Address	Layer One Sub weight = 0.3
	2	Abnormal Request URL	
	3	Abnormal URL of Anchor	
	4	Abnormal DNS Record	
	5	Abnormal URL	
Security & Encryption (Weight = 0.2)	1	Using SSL Certificate	Layer Two Sub weight = 0.4
	2	Certification Authority	
	3	Abnormal Cookie	
	4	Distinguished Names Certificate(DN)	
Source Code & Java script (Weight = 0.2)	1	Redirect Pages	
	2	Straddling Attack	
	3	Pharming Attack	
	4	Using onMouseOver to Hide the Link	
	5	Server Form Handler (SFH)	
Page Style & Contents (Weight =0.1)	1	Spelling Errors	Layer Three Sub weight = 0.3
	2	Copying Website	
	3	Using Forms with "*Submit*" Button	
	4	Using Pop-Ups Windows	
	5	Disabling Right-Click	
Web Address Bar (Weight = 0.1)	1	Long URL Address	
	2	Replacing Similar Characters for URL	
	3	Adding Prefix or Suffix	
	4	Using the @ Symbol to Confuse	
	5	Using Hexadecimal Character Codes	
Social Human Factor (Weight = 0.1)	1	Much Emphasis on Security and Response	
	2	Public Generic Salutation	
	3	Buying Time to Access Accounts	
Total Weight			1

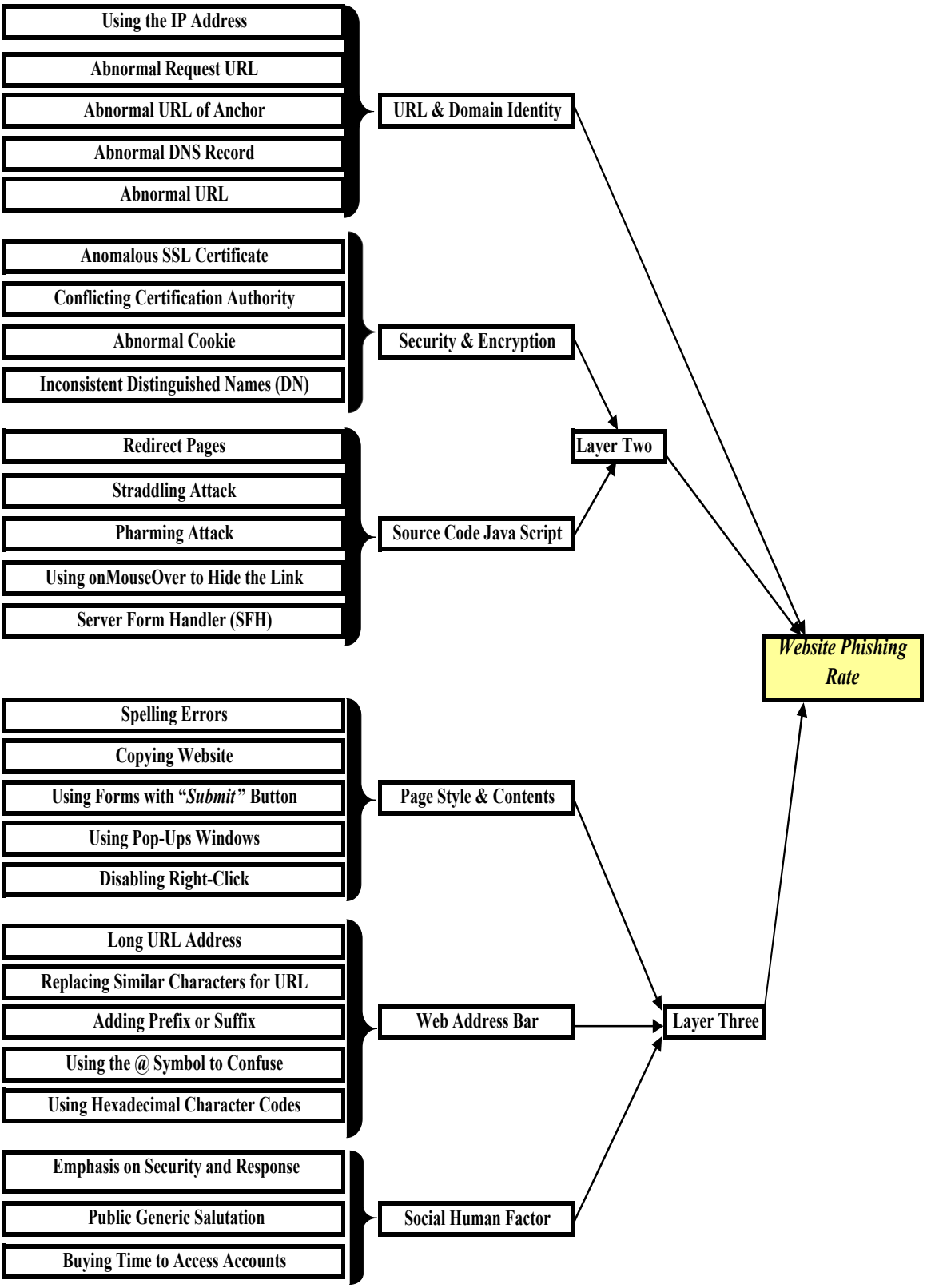


Figure 4.4: Architecture of the phishing detection fuzzy modelling system

4.6 System Design

Detecting phishing website risk rate is performed based on six criteria: URL & Domain Identity, Security & Encryption, Source Code & Java script, Page Style & Contents, Web Address Bar and Social Human Factor. There are also a different number of components for each criterion, such as five components for URL & Domain Identity, Source Code & Java script, Page Style & Contents and Web Address Bar, four components for Security & Encryption, and three components for Social Human Factor. Therefore, there are twenty-seven components in total.

There are three layers in this phishing website fuzzy model. The first layer contains only URL & Domain Identity criteria with a weight equal to 0.3 according to its importance; the second layer contains Security & Encryption criteria and Source Code & Java script criteria with a weight equal to 0.2 each; the third layer contains Page Style & Contents criteria, Web Address Bar criteria And Social Human Factor criteria with a weight equal to 0.1 each. Depending on this fuzzy logic layered architecture model we can calculate the final phishing fuzzy output result as:

Phishing Website Risk Rating = 0.3 * URL & Domain Identity crisp [First layer] + ((0.2 * Security & Encryption crisp)+(0.2 * Source Code & Java script crisp)) [Second layer] + ((0.1 * Page Style & Contents crisp) +(0.1 * Web Address Bar crisp) + (0.1 * Social Human Factor crisp)) [Third layer]

4.7 Fuzzy Rule Base

All fuzzy rules implemented in our proposed detection model were derived based on our own phishing background experience and expert knowledge supported by a series of experimental phishing scenarios with case-studies. Next we will show all fuzzy rules for all phishing website criteria and layers.

4.7.1 The Rule Base1 for Layer 1

The rule base has five input parameters and one output. The rule contains all the "IF-THEN" rules of the system. For each entry of the rule base, each component is assumed to be one of the three values and each criterion has five components. Therefore, the rule base 1-1 contains $(3^5) = 243$ entries. The output of rule base 1-1 is one of the phishing website risk rate fuzzy sets (Genuine, Doubtful or Fraud) representing URL & Domain Identity criteria phishing risk rate. A sample of the structure and the entries of the rule base 1-1 for layer 1 are shown in Table 4.2. The system structure for URL & Domain Identity criteria is the joining of its five components (Using the IP Address, Abnormal Request URL, Abnormal URL of Anchor, Abnormal DNS record and Abnormal URL), which produces the URL & Domain Identity criteria (Layer one) as shown in Figure 4.5. Further, the three-dimensional plots of this system structure are shown in Figure 4.6 using MATLAB.

Table 4.2: Sample of rule base1-1 entries for URL & Domain Identity criteria

Rule #	(comp. 1) Using the IP Address	(comp. 2) Abnormal Request URL	(comp. 3) Abnormal URL Anchor	(comp. 4) Abnormal DNS record	(comp. 5) Abnormal URL	URL & Domain Identity Criteria Phishing Risk *(Layer one)*
1	Low	Low	Low	Low	Low	Genuine
2	Low	Low	Low	Low	Moderate	Genuine
3	Low	Low	Low	Moderate	Moderate	Doubtful
4	Low	Low	Low	Moderate	High	Doubtful
5	Low	Low	Moderate	Moderate	High	Fraud
6	Low	Moderate	Moderate	Low	High	Fraud
7	Moderate	Low	High	Moderate	High	Fraud
8	High	Moderate	Low	Low	Low	Doubtful
9	Low	High	Low	Low	Moderate	Doubtful
10	High	Moderate	High	High	Low	Fraud

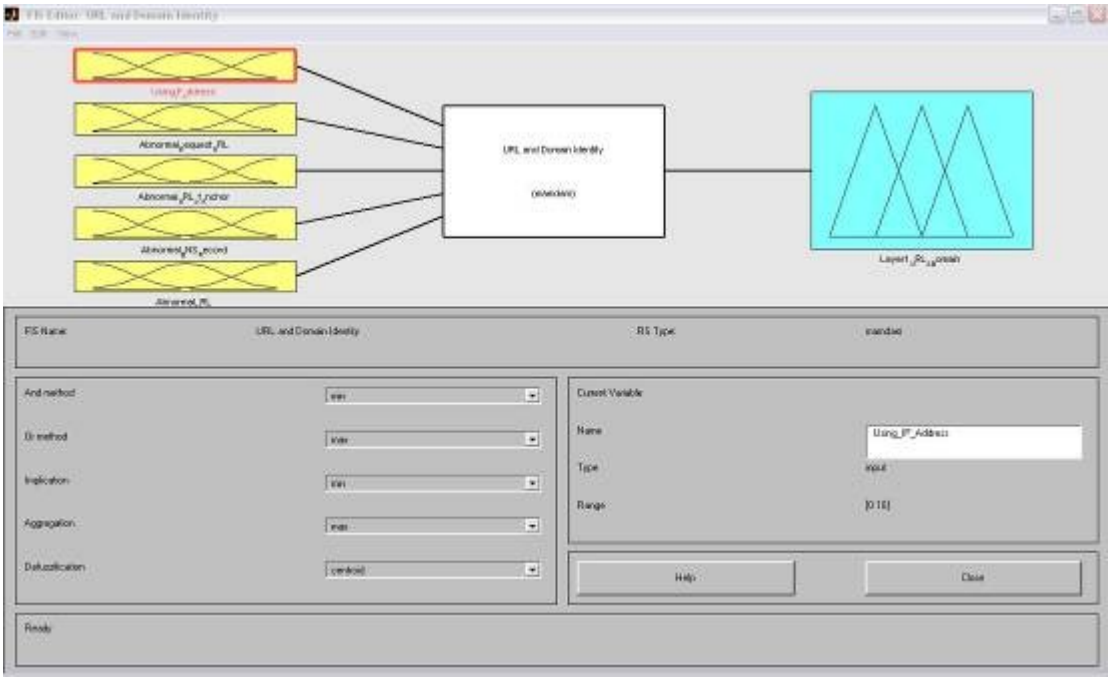


Figure 4.5: System structure for URL & Domain Identity criteria

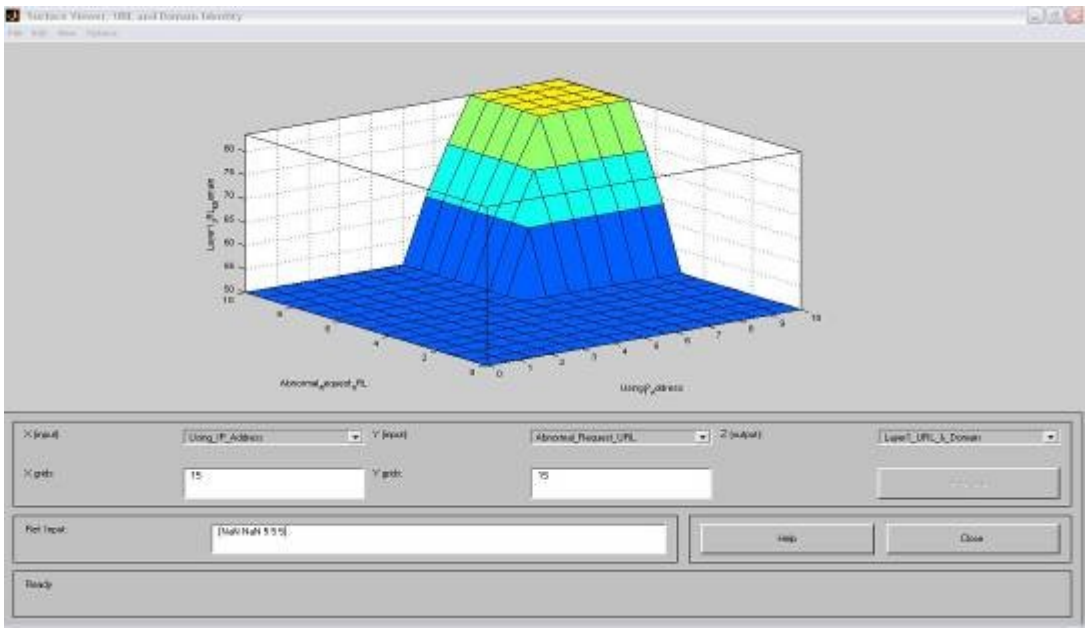


Figure 4.6: Three-dimensional plot for URL & Domain Identity criteria

4.7.2 The Rule Base for Layer 2

In Layer 2, there are two inputs, Security & Encryption and Source Code & Java script, and one output. The system structure for Security & Encryption criteria is the joining of

its four components (Using SSL certificate, Certification authority, Abnormal Cookie and Distinguished Names Certificate(DN)) using rule base 2-1, which produces Security & Encryption criteria. The system structure for Source Code & Java script criteria is the joining of its five components (Redirect pages, Straddling attack, Pharming Attack, Using onMouseOver to hide the Link and Server Form Handler (SFH)) using Rule base 2-2, which produces Source Code & Java script criteria.

Table 4.3 shows a sample of the rule base 2-1 for Security & Encryption criteria using its four components.

Table 4.3: Sample of rule base 2-1 entries for Security & Encryption criteria

Rule #	(comp. 1) Using SSL Certificate	(comp. 2) Certification Authority	(comp. 3) Abnormal Cookie	(comp. 4) Distinguished Names Certificate	Security & Encryption Criteria Phishing Risk
1	Low	Low	Low	Low	Genuine
2	Low	Moderate	Low	Low	Genuine
3	Moderate	Low	Low	Moderate	Doubtful
4	Low	Moderate	Low	Moderate	Doubtful
5	Low	Low	Moderate	Moderate	Fraud
6	Low	Moderate	Moderate	Low	Doubtful
7	Moderate	Low	High	Moderate	Fraud
8	High	Moderate	Low	Low	Doubtful
9	Low	High	Low	Low	Fraud
10	High	Moderate	High	High	Fraud

The structure and the entries of the rule base for layer 2 are illustrated in Table 4.4. The system structure for layer 2 is the combination of two phishing website criteria (Security & Encryption and Source Code & Java script), which produces rule base 2 as shown in Figure 4.7, and its three-dimensional plots are shown in Figure 4.8 using MATLAB. The rule base contains $(3^2) = 9$ entries and the output of rule base 2 is one of the phishing website risk rate fuzzy sets (Legal, Uncertain or Fake) representing layer two criteria phishing website risk rate.

Table 4.4: Rule base 2 structure and entries for layer two

Rule #	Security & Encryption	Source Code & Java script	Phishing Risk (Layer Two)
1	Genuine	Genuine	Legal
2	Genuine	Doubtful	Legal
3	Genuine	Fraud	Uncertain
4	Doubtful	Genuine	Legal
5	Doubtful	Doubtful	Uncertain
6	Doubtful	Fraud	Uncertain
7	Fraud	Genuine	Uncertain
8	Fraud	Doubtful	Uncertain
9	Fraud	Fraud	Fake

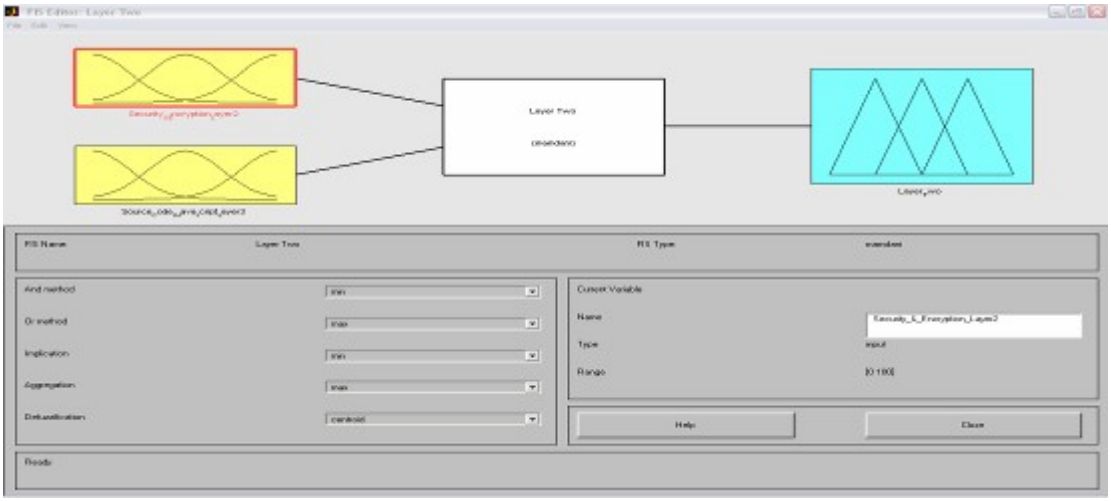


Figure 4.7: System structure for layer two

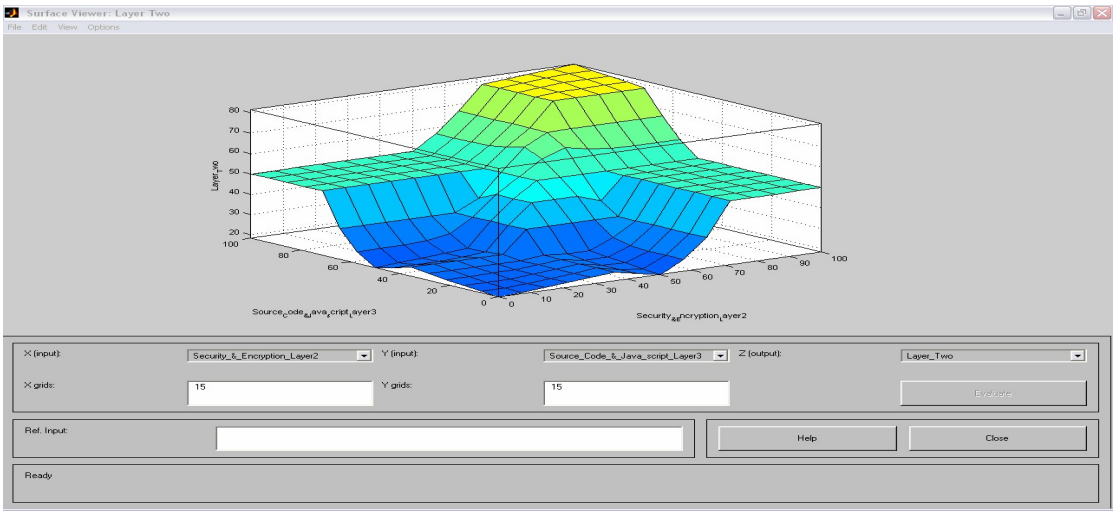


Figure 4.8: Three-dimensional plot for layer two

4.7.3 The Rule Base for Layer 3

In layer 3, there are three inputs, Page Style & Contents, Web Address Bar and Social Human Factor, and one output. The system structure for Page Style & Contents criteria is the joining of its five components (Spelling errors, Copying website, Using forms with "*Submit*" button, Using Pop-Up windows and Disabling Right-Click) using Rule base 3-1, which produces Page Style & Contents criteria. The system structure for Web Address Bar criteria is the joining of its five components (Long URL address, Replacing similar characters for URL, Adding a prefix or suffix, Using the @ Symbol to Confuse and Using Hexadecimal Character Codes) using Rule base 3-2, which produces Web Address Bar criteria. The system structure for Social Human Factor criteria is the joining of its three components (Much emphasis on security and response, Public generic salutation and Buying Time to Access Accounts) using Rule base 3-3, which produces Social Human Factor criteria.

Table 4.5 shows a sample of the rule base 3-3 for Social Human Factor criteria using its three components.

Table 4.5: Sample of rule base 3-3 entries for Social Human Factor criteria

Rule #	(comp. 1) Much Emphasis on Security and Response	(comp. 2) Public Generic Salutation	(comp. 3) Buying Time to Access Accounts	Social Human Factor Criteria Phishing Risk
1	Low	Low	Low	Genuine
2	Low	Moderate	Low	Genuine
3	Moderate	Low	Moderate	Doubtful
4	Low	Moderate	Low	Doubtful
5	Moderate	Low	Moderate	Doubtful
6	Moderate	Moderate	Moderate	Fraud
7	Moderate	Low	Low	Doubtful
8	High	Moderate	High	Fraud
9	Low	High	Low	Fraud
10	High	Moderate	High	Fraud

A sample of the structure and the entries of the rule base for layer 3 are shown in Table 4.6. The system structure for layer 3 is the combination of Page Style & Contents, Web Address Bar and Social Human Factor, which produces rule base 3 as shown in Figure 4.9. The three-dimensional plots of this structure are shown in Figure 4.10 using MATLAB. The rule base contains $(3^3) = 27$ entries and the output of rule base 3 is one of the phishing website risk rate fuzzy sets (Legal, Uncertain or Fake) representing Layer Three criteria phishing risk rate.

Table 4.6: Rule base3 structure and entries for layer three

Rule #	Page Style & Contents	Web Address Bar	Social Human Factor	**Phishing Risk (Layer Three)**
1	Genuine	Genuine	Genuine	Legal
2	Genuine	Doubtful	Fraud	Uncertain
3	Genuine	Fraud	Fraud	Fake
4	Doubtful	Genuine	Genuine	Legal
5	Doubtful	Doubtful	Doubtful	Uncertain
6	Doubtful	Fraud	Doubtful	Uncertain
7	Fraud	Genuine	Genuine	Legal
8	Fraud	Doubtful	Doubtful	Uncertain
9	Fraud	Fraud	Fraud	Fake

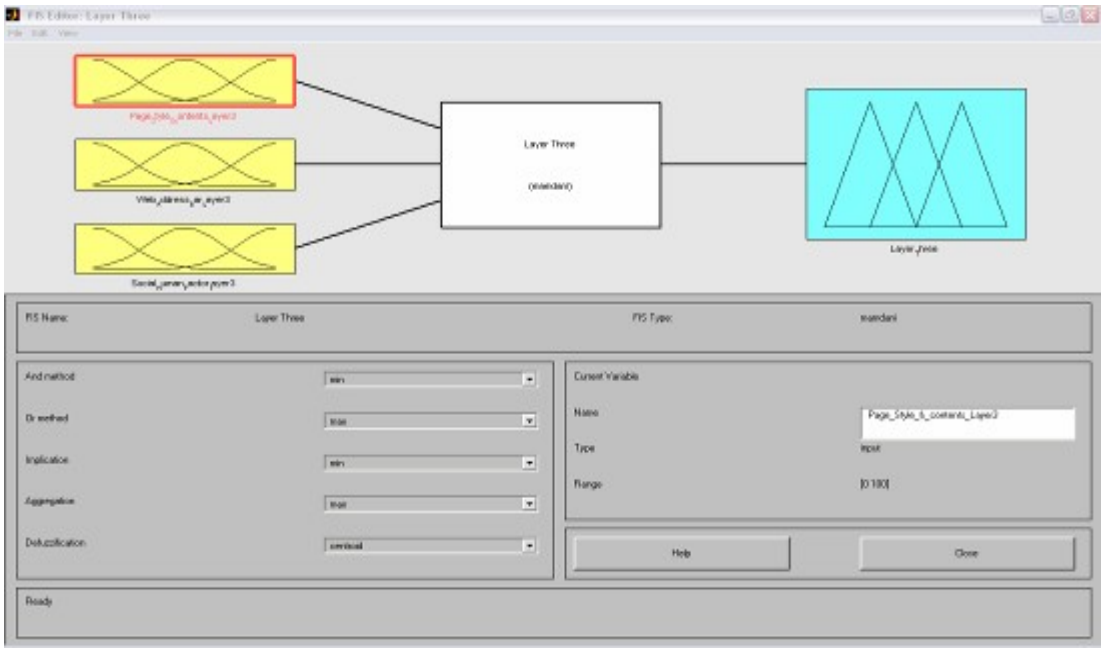


Figure 4.9: System Structure for Layer Three

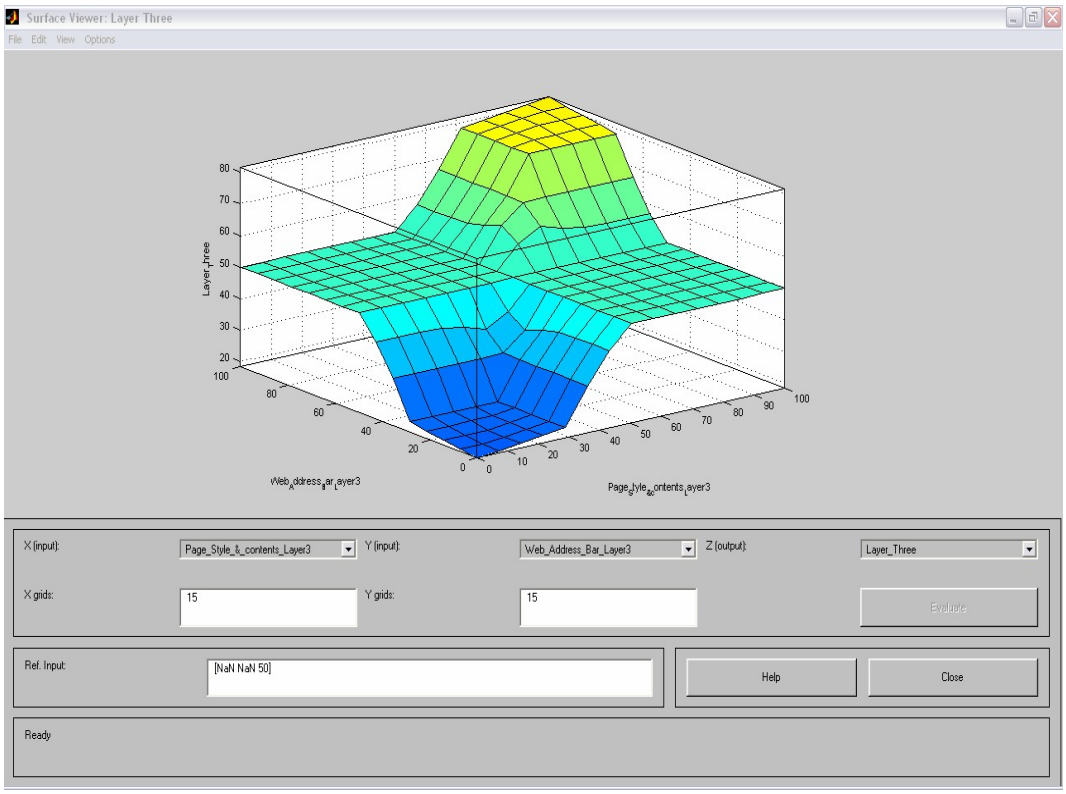


Figure 4.10: Three-dimensional plots for layer three

4.7.4 The Rule Base for Final Phishing Website Risk Rate

In the phishing website rule base last phase, there are three inputs, layer one, layer two and layer three, and one output identifying the risk rate of the phishing website. The structure and the entries of the rule base for phishing website risk rate are shown in Table 4.7. The system structure for the fuzzy detection model is the combination of layer one, layer two and layers three, which produces the final phishing website rule base as shown in Figure 4.11. The three-dimensional plots of this structure are shown in Figure 4.12 using MATLAB. The rule base contains $(3^3) = 27$ entries and the output of final phishing website rule base is one of the final output fuzzy sets (Legitimate, Suspicious, Phishy) representing the final phishing website risk rate.

Table 4.7: Rule base structure and entries for the final phishing website risk rate

Rule	URL & Domain Identity (Layer one)	Layer Two	Layer Three	Final Phishing Website Risk Rate
1	Genuine	Legal	Legal	Legitimate
2	Genuine	Legal	Uncertain	Legitimate
3	Genuine	Legal	Fake	Suspicious
4	Genuine	Uncertain	Legal	Suspicious
5	Genuine	Uncertain	Uncertain	Suspicious
6	Genuine	Uncertain	Fake	Phishy
7	Genuine	Fake	Legal	Suspicious
8	Genuine	Fake	Uncertain	Phishy
9	Genuine	Fake	Fake	Phishy
10	Doubtful	Legal	Legal	Legitimate
11	Doubtful	Legal	Uncertain	Suspicious
12	Doubtful	Legal	Fake	Phishy
13	Doubtful	Uncertain	Legal	Suspicious
14	Doubtful	Uncertain	Uncertain	Suspicious
15	Doubtful	Uncertain	Fake	Phishy
16	Doubtful	Fake	Legal	Phishy
17	Doubtful	Fake	Uncertain	Phishy
18	Doubtful	Fake	Fake	Phishy
19	Fraud	Legal	Legal	Suspicious
20	Fraud	Legal	Uncertain	Suspicious
21	Fraud	Legal	Fake	Phishy
22	Fraud	Uncertain	Legal	Suspicious
23	Fraud	Uncertain	Uncertain	Suspicious
24	Fraud	Uncertain	Fake	Phishy
25	Fraud	Fake	Legal	Phishy
26	Fraud	Fake	Uncertain	Phishy
27	Fraud	Fake	Fake	Phishy

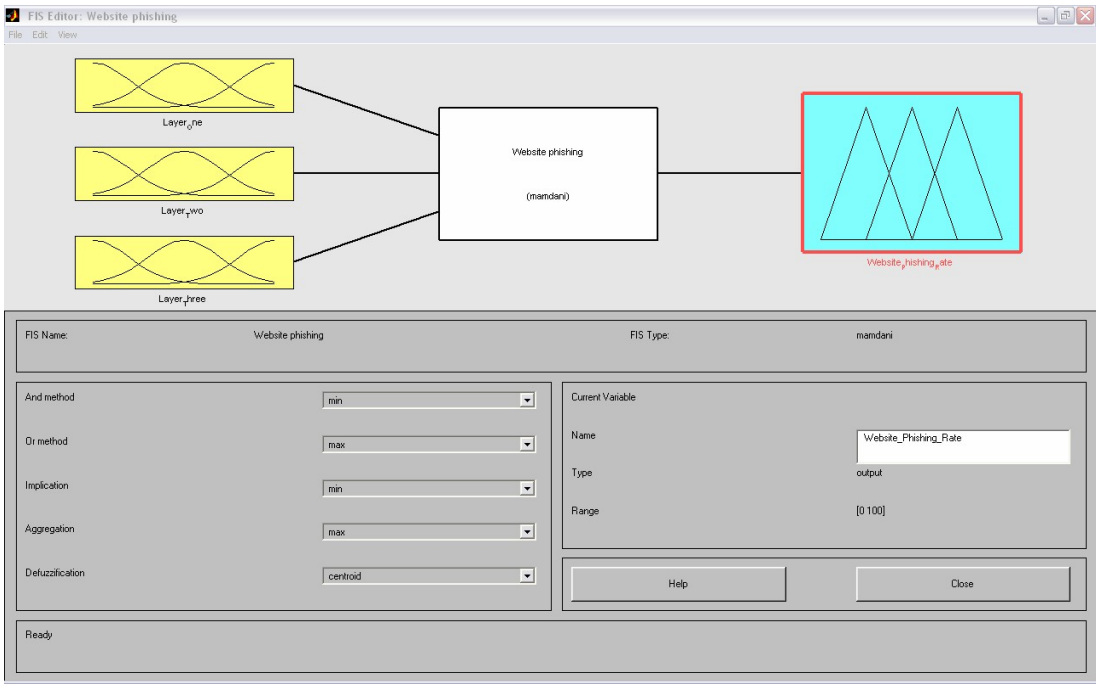


Figure 4.11: System structure for final phishing website risk rate

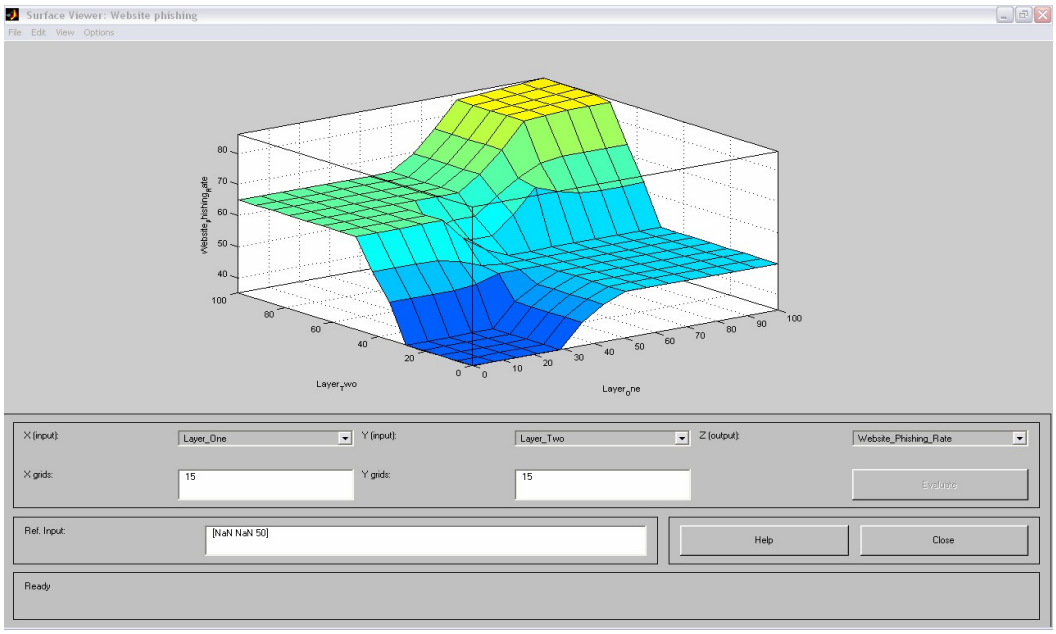


Figure 4.12: Three-dimensional plots for final phishing website risk rate

4.8 Experiments and Evaluation Results

The aggregated surface of the rule evaluation is defuzzified using the Mamdani method (Liu, et al., 2005) to find the Centre Of Gravity (COG). Centroid defuzzification technique shown in Equation (1) can be expressed as:

$$X^* = \frac{\int \mu_i(x)\, x\, dx}{\int \mu_i(x)\, dx}$$

Equation (1)

Where x^* is the defuzzified output, $\mu_i(x)$ is the aggregated membership function and x is the output variable.

The proposed Phishing website detection system has been implemented in MATLAB 6.5. The results of some input combinations are listed in Tables 4.8, 4.9, 4.10, 4.11, 4.12, 4.13.

When all phishing website risk criteria represented by the three layers have zero inputs, which points to a *Low* phishing indicator as represented by the linguistic value, the final phishing website risk rate will be very low (13.8%), representing [***legitimate website***] as shown in Table 4.8 and Figure 4.13, respectively.

Further, when all phishing website risk criteria represented by the three layers have 10 input values, which points to a *High* phishing indicator as represented by the linguistic value, the final phishing website risk rating will be very high (86.2 %), representing [***phishy website***] as shown in Table 4.9, which means that the website is undoubtedly a forged phishing website which is used for phishing users and clients to obtain their bank accounts, passwords, credit card numbers, or other important information leading to catastrophic consequences.

Meanwhile, a final phishing website risk rating will be balanced (50%), representing [***suspicious website***], when Layer one (URL & Domain Identity) of the phishing website risk criteria has 10 input values, which points to a *High* phishing indicator as represented by the linguistic value, and all other layers have the value of zero inputs as shown in Table 4.10 and Figure 4.14, respectively. The same result can be achieved and shown in Table 4.11 when all phishing website risk criteria represented by the three layers have middle (5) input values, which points to a *Moderate* phishing indicator as represented by the linguistic value. These results shows the significance and importance of the phishing website criteria (URL & Domain Identity) represented by layer one, especially when compared to the other criteria and layers.

Table 4.12 shows that, when Layer one and Layer two of the phishing website risk criteria have middle (5) input values, pointing to a *Moderate* phishing indicator as represented by the linguistic value, and the third Layer has the value of 10 input values, pointing to a *High* phishing indicator as represented by the linguistic value, the final phishing website risk rating will be reasonably high (65%), representing [***phishy website***]. This result clearly shows that, even if some of the phishing website characteristics are not very clear or are not definite, the website can still be phishy, especially when other phishing characteristics can be clearly identified.

Table 4.13 shows that, when Layer one of the phishing website risk criteria (URL & Domain Identity) has middle (5) input values, pointing to a *Moderate* phishing indicator as represented by the linguistic value, and all the other Layers have the value of zero input values, pointing to a *Low* phishing indicator as represented by the linguistic value, the final phishing website risk rating will be reasonably low (35%), representing a [***legitimate website***]. This result clearly shows that, even if we were able to identify some of phishing website characteristics, the website can still be safe and legitimate, especially when other phishing characteristics cannot be clearly recognized.

The results also indicate that the worst phishing website rate equals 86.2% and the best phishing website rate is 13.8%, rather than a full range, i.e. 0 to 100, because of the fuzzification process.

Table 4.8: All lowest (0) inputs for layer one, layer two, and layer three

Comp	Layer One URL & Domain Identity	Layer Two		Layer Three			% Phishing Website Risk Rating
		Security & Encryption	Source Code & Java script	Page Style & Contents	Web Address Bar	Social Human Factor	
1	0	0	0	0	0	0	13.8% Legitimate Website
2	0	0	0	0	0	0	
3	0	0	0	0	0	0	
4	0	0	0	0	0		
5	0		0	0	0		

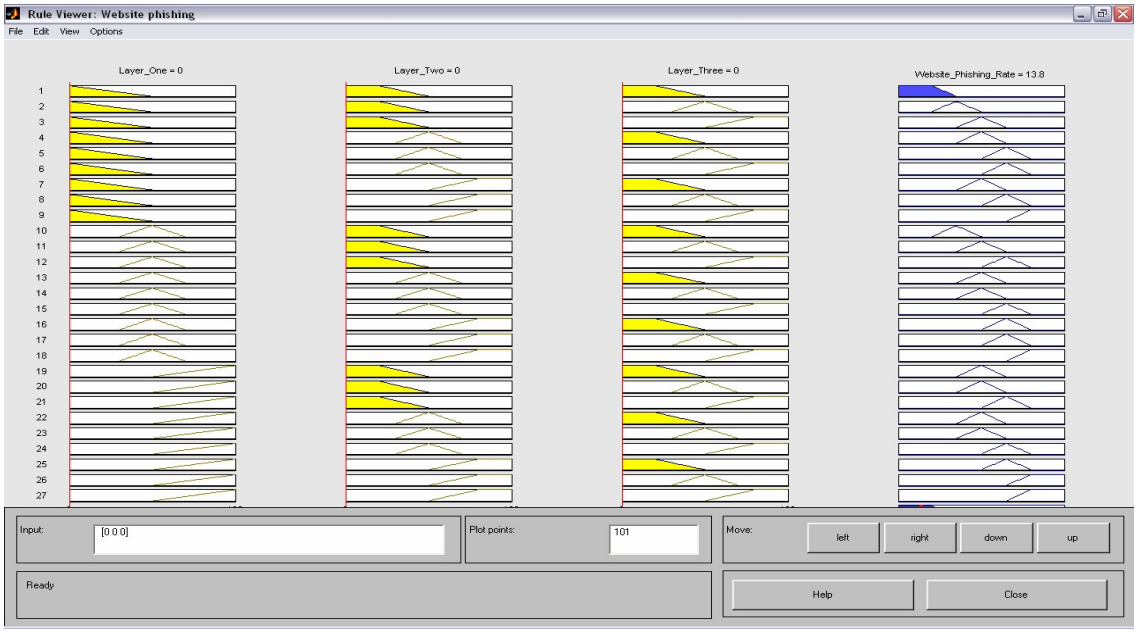


Figure 4.13: Rule viewer for final phishing website risk rate. Lowest (0) inputs for all layers criteria

Table 4.9: All highest (10) inputs for all three layers

Comp	Layer One URL & Domain Identity	Layer Two		Layer Three			% Phishing Website Risk Rating
		Security & Encryption	Source Code & Java script	Page Style & Contents	Web Address Bar	Social Human Factor	
1	10	10	10	10	10	10	86.2 Phishy Website
2	10	10	10	10	10	10	
3	10	10	10	10	10	10	
4	10	10	10	10	10		
5	10		10	10	10		

Table 4.10: Five highest (10) for layer one (URL & Domain Identity) and all others lowest (0)

Comp	Layer One URL & Domain Identity	Layer Two		Layer Three			% Phishing Website Risk Rating
		Security & Encryption	Source Code & Java script	Page Style & Contents	Web Address Bar	Social Human Factor	
1	10	0	0	0	0	0	50% Suspicious Website
2	10	0	0	0	0	0	
3	10	0	0	0	0	0	
4	10	0	0	0	0		
5	10		0	0	0		

Figure 4.14: Rule viewer for final phishing website risk rate. Five highest (10) inputs for criteria (URL & Domain Identity) and all others lowest (0) inputs

Table 4.11: Middle (5) inputs for all three layers

Comp	Layer One URL & Domain Identity	Layer Two		Layer Three			% Phishing Website Risk Rating
		Security & Encryption	Source Code & Java script	Page Style & Contents	Web Address Bar	Social Human Factor	
1	5	5	5	5	5	5	50% Suspicious Website
2	5	5	5	5	5	5	
3	5	5	5	5	5	5	
4	5	5	5	5	5		
5	5		5	5	5		

Table 4.12: Five middle (5) inputs for layer one and layer two and highest (10) inputs for layer three

Comp	Layer One URL & Domain Identity	Layer Two		Layer Three			% Phishing Website Risk Rating
		Security & Encryption	Source Code & Java script	Page Style & Contents	Web Address Bar	Social Human Factor	
1	5	5	5	10	10	10	65% Phishy Website
2	5	5	5	10	10	10	
3	5	5	5	10	10	10	
4	5	5	5	10	10		
5	5		5	10	10		

Table 4.13: Five middle (5) inputs for layer one (URL & Domain Identity) and all others lowest (0) inputs

Comp	Layer One URL & Domain Identity	Layer Two		Layer Three			% Phishing Website Risk Rating
		Security & Encryption	Source Code & Java script	Page Style & Contents	Web Address Bar	Social Human Factor	
1	5	0	0	0	0	0	35% Legitimate Website
2	5	0	0	0	0	0	
3	5	0	0	0	0	0	
4	5	0	0	0	0		
5	5		0	0	0		

To validate these results, we implemented these fuzzy rules used for the fuzzy inference engine on a sample of 120 different e-banking websites drawn from the public benchmark Phishtank archive data (Phishtank, 2008), distributed as 60 phishing websites, 35 suspicious websites and 25 legitimate websites. The results show that there are good numbers of false positive and false negative signs for all legitimate, suspicious and phishing websites, indicating a rather high website misclassification rate.

Table 4.14: Results of website legitimacy decision using fuzzy-based detection model

Decision Website Legitimacy	Legitimate	Suspicious	Phishy
Legitimate Website	18	4	3
Suspicious Website	6	24	5
Phishing Website	9	8	43

For example, and as shown in Table 4.14, there were 7 legitimate websites misclassified as suspicious or phishy websites, and 17 phishing websites misclassified as legitimate or suspicious websites.

These results were expected, since all fuzzy phishing rules were extracted only from human expert knowledge and background experience, which cannot guarantee a high precision in the rules' validity. This emphasises the need to look for more efficient methods to extract from and mine these rules in an intelligent way to achieve a more precise and accurate results output. This will be presented in the following chapter of this research.

4.9 Improving Our Fuzzy-Based Phishing Detection Model

The fuzzy phishing website detection system requires human experts to determine the fuzzy sets and set of fuzzy rules which allow us to easily construct if-then rules that reflect common ways of describing and detecting phishing websites' characteristics and attacks. These tasks are time-consuming. However, if the fuzzy rules are automatically generated, less time would be consumed in building a good phishing website characteristic classifier and the development time for building or updating phishing website classifiers would be shortened by reducing the human intervention.

In the following chapter we will propose and develop a prototype intelligent fuzzy phishing detection system to demonstrate the effectiveness of data mining techniques that utilise fuzzy logic. This system should combine two distinct website phishing detection approaches:

1) Website phishing detection using fuzzy data mining techniques, and
2) Website phishing detection using traditional rule-based expert system techniques.

The first approach components look for deviations from stored patterns of normal phishing behaviour. The second approach looks for previously described patterns of behaviour that are likely to indicate phishing. Both websites' contents and system audit data are used as inputs.

Our aim next is to demonstrate that the fuzzy data mining technique provides an effective means to learn and become alert, based on patterns extracted from large

amounts of data, and to demonstrate that the integration of fuzzy logic with the data mining techniques enables an improved performance over similar techniques.

Chapter 5

Fuzzy-Based Classification Mining Intelligent Model for Phishing Website Detection

5.1 Introduction

In the first half of this chapter, we will introduce and investigate well-known traditional classification approaches and Associative Classification machine-learning techniques. We will also review well-known traditional classification approaches to introduce association rule discovery and classification tasks in data mining.

In the second part of this chapter, we will propose a dynamic, resilient Intelligent System model, based on a specific AI approach to phishing website detection. The technique that is being investigated includes fuzzy logic with a supervised machine learning approach, which uses simple data mining associative classification techniques to process the phishing data features and patterns. The proposed system automates the fuzzy rules production by using the extracted classification rules, which are produced by using associative classification algorithms and techniques. These fuzzy rules allow us to construct 'if-then' rules that reflect the relationships between the different phishing characteristics and features and their association with one another for the final phishing

website detection rate. In other words, we are interested in designing, developing and evaluating a resilient and effective Intelligent Phishing Detection System with a technique that is capable of identifying indications for phishing websites tracks, and classifying individual web pages to determine whether they are legitimate, suspicious or phishing websites. The model should be precise and adaptable; it should also have a real-time application, and reduce error rates and false alarms. We believe that this machine learning model will expose most of deceptions, tricks and schemes used by phishers today.

5.2 Classification in Data Mining

Data mining and knowledge discovery techniques have been applied to several areas including market analysis, industrial retail, decision support and financial analysis. Knowledge Discovery from Databases (KDD) (Fayyad, et al., 1998) involves data mining as one of its main phases to discover useful patterns.

The data mining task is to generate all association rules in the database, which have a support greater than *min sup*, i.e., the rules are frequent, and which also have confidence greater than *min conf*, i.e., the rules are strong. Here we are interested in rules with a specific item, called the *class*, as a consequent, i.e., we mine rules of the form $A \rightarrow c_i$ where c_i is a class attribute $(1 \leq i \leq k)$ (Qaddoum, 2009). This technique is called classification./

The goal of classification is to build a model (a set of rules) from a labelled training data set, in order to classify new data objects, known as test data objects, as accurately as possible. Classification in data mining is a two-step process, where in the first step; a

classification algorithm is used to learn the rules from a training data set. The second step involves using the rules extracted in the first step to predict classes of test objects.

There are many classification approaches for extracting knowledge from data such as divide-and-conquer (Quinlan, 1987a), separate-and-conquer (Furnkranz, 1999) (also known as rule induction), covering (Cendrowska, 1987) and statistical approaches (Duda and Hart, 1973; Meretakis and Wüthrich, 1999). The divide-and-conquer approach starts by selecting an attribute as a root node using information gain (Quinlan, 1979), and then it makes a branch for each possible level of that attribute. This will split the training instances into subsets, one for each possible value of the attribute. The same process is repeated until all instances that fall in one branch have the same classification or the remaining instances cannot split any further. The separate-and-conquer approach on the other hand, starts by building up the rules in a greedy fashion, one by one. After a rule is found, all instances covered by the rule are removed and the same process is repeated until the best rule found that has a large error rate. Statistical approaches such as Naïve Bayes (Duda and Hart, 1973) computes probabilities of classes in the training data set using the frequency of attribute values associated with them in order to classify test instances. Other approaches such as covering algorithms select each of the available classes in the training data in turn, and look for a way of covering the most of training instances to that class in order to come up with high accuracy rules.

Numerous algorithms are based on these approaches such as decision trees (Quinlan, 1986; Quinlan, 1993; Quinlan, 1998), PART (Frank and Witten, 1998), RIPPER (Cohen, 1995), Prism (Cendrowska, 1987) and others.

5.3 Common Classification Techniques

5.3.1 Decision Trees (C4.5 Algorithm)

A popular approach for classification and prediction is that of decision trees (Quinlan, 1979; Quinlan, 1986; Quinlan, 1998). A common example of a decision tree is the twenty questions game, where one thinks of a common thing that is known by all the participants in the game. Participants start asking questions, usually up to twenty, in order to guess the identity of that thing. Most often, a good player seldom needs to ask all the questions. In that game, the decision tree represents the series of the questions in which the answer of the first question determines the next question to be asked and so on. In constructing a decision tree, a candidate record will enter the root node, and a branch for each possible value for the candidate is built. The same process is applied recursively until all the records in a node end up with the same class or the tree cannot be split any further (Quinlan, 1979). The selection of the candidate attribute to split the data on is a crucial task, since it effects the distribution of classes in each branch. This process can be implemented in various ways based on the algorithm in use. After the tree has been constructed, each path from the root node to each of the leaf nodes represents a rule. The antecedent of the rule is given by the path from the root node to the leaf node, and the consequent is the majority class that is assigned by the leaf node.

Several pruning methods are used to simplify the rules and to discard unnecessary ones. Pruning the tree will involve either replacing some sub-trees with leaf nodes (subtree replacement) or raising some nodes to replace the nodes higher in the tree (sub-tree rising) (Quinlan, 1993). Both of these operations are examples of post-pruning techniques (Witten and Frank, 2000).

5.3.2 Rule Induction and Covering Approach (RIPPER)

Repeated Incremental Pruning to Produce Error Reduction algorithm (RIPPER) is a rule induction algorithm that has been developed by Cohen (Cohen, 1995). RIPPER builds the rules set as follows: The training data set is divided into two sets, a pruning set and a growing set. RIPPER constructs the classifier using these two sets by repeatedly inserting rules starting from an empty rule set. The rule-growing algorithm starts with an empty rule, and heuristically adds one condition at a time until the rule has no error rate on the growing set. In fact, RIPPER is a refined version of the IREP algorithm that adds some modifications. First, a new stopping condition for generating rules has been introduced. IREP utilises a heuristic that stops adding rules when a rule learned has an error rate greater than 50% on the pruning data. This heuristic may stop too early especially for application domains that hold large number of low coverage rules. RIPPER stops adding a rule using the minimum description length principle (MDL) (Rissanen, 1985) where after a rule is inserted, the total description length of the rules set and the training data is estimated. If this description length is larger than the smallest MDL obtained so far, RIPPER stops adding rules. The MDL assumes that the best model (set of rules) of data is the one that minimises the size of the model plus the amount of information required to identify the exceptions relative to the model (Witten and Frank, 2000).

5.3.3 PRISM

Prism was developed by Cendrowska in (Cendrowska, 1987) and can be categorised as a covering algorithm for constructing classification rules. The covering approach starts by taking one class among the available ones in the training data set, and then it seeks a

way of covering all instances to that class, at the same time it excludes instances not belonging to that class. This approach usually tries to create rules with maximum accuracy by adding one condition to the current rule antecedent. At each stage, Prism chooses the condition that maximises the probability of the desired classification. The process of constructing a rule terminates as soon as a stopping condition is met. Once a rule is derived, Prism continues building rules for the current class until all instances associated with the class are covered. Once this happens, another class is selected, and so forth. Prism normally generates perfect rules (those with 0% error rate) and measures the accuracy of its rules using the accuracy formula:

$$(P/T) \tag{5.1}$$

Where P represents the number of positive examples and T represents the number of negative examples covered by a rule. Prism has an advantage over decision trees in that a rule can be added to the created rule set without having any impact on any existing rules. On the other hand, adding a path to the tree structure may require reshaping the whole tree (Witten and Frank, 2000). Though, unlike decision trees which classify an instance using rules produced by reading them directly from the tree, independence of the rules in Prism may suffer from problems, such as an instance may be associated with more than one rule with different classes.

5.3.4 Hybrid Approach (PART)

Unlike the C4.5 and RIPPER techniques that operate in two phases, the PART algorithm generates rules one at a time by avoiding extensive pruning (Frank and Witten, 1998). The C4.5 algorithm employs a divide-and-conquer approach, and the RIPPER algorithm uses a separate-and-conquer approach to derive the rules. PART

combines both approaches to find and generate rules. It adopts separate-and-conquer to generate a set of rules and uses divide-and-conquer to build partial decision trees. The way PART builds and prunes a partial decision tree is similar to that of C4.5, but PART avoids constructing a complete decision tree and builds partial decision trees. PART differs from RIPPER in the way rules are created, where in PART, each rule corresponds to the leaf with the largest coverage in the partial decision tree. On the other hand, RIPPER builds the rule in a greedy fashion, starting from an empty rule, it adds conditions, until the rule has no error rate and the process is repeated. Missing values and pruning techniques are treated in the same way as C4.5.

Experimental tests using PART, RIPPER and C4.5 on different data sets from (Merz and Murphy, 1996) have been reported in (Frank and Witten, 1998). The results revealed that despite the simplicity of PART, it generates sets of rules, which are as accurate as C4.5 and more accurate (though larger) than those of RIPPER.

5.4 Classification Based on Association (CBA)

Classification and association-rule discovery are two of the most important tasks addressed in the data mining literature. Association mining aims to discover descriptive knowledge from databases, while classification focuses on building a classification model for categorizing new data. Both association pattern discovery and classification rule mining are essential to practical data mining applications. If these two relevant jobs can be somehow integrated, great savings and conveniences to the user can be resulted. Hence, considerable efforts have been made to integrate these two techniques into one system. In recent years, extensive research has been carried out to integrate both

approaches. By focusing on a limited subset of association rules, i.e. those rules where the consequent of the rule is restricted to the class variables, it is possible to build more accurate classifiers (Lan, et al., 2005).

Association rule mining and classification are analogous tasks, with the exception that classification's main aim is the prediction of class labels, while association rule mining describes associations between attribute values in a database. In the last few years, association rule mining has been successfully used to build accurate classifiers, which resulted in a new approach, known as Associative Classification (AC) (Ali, et al., 1997; Liu, et al., 1998). Associative Classification (AC) is a branch in data mining that combine's classification and association rule mining.

In other words, it utilises association rule discovery methods in classification data sets. Several studies (Liu, et al., 1998; Li, et al., 2001; Yin and Han, 2003) provide evidence that AC approaches are able to extract more accurate classifiers than traditional classification techniques, such as decision trees (Quinlan, 1993, Quinlan, 1998), rule induction (Quinlan and Cameron-Jones, 1993; Cohen, 1995) and probabilistic (Duda and Hart, 1973) approaches.

CBA is the first algorithm using association rules for classification (Liu et al., 1998). This algorithm generates a special subset of association rules called Class Association Rules (CARs). The difference between association rules and CARs is the consequences of the rules. The consequence of CARs is only limited to class label value. Thus the form of CAR called ruleitem is $X \rightarrow C$ where C is a set of all class labels.

The CBA algorithm consists of two parts: a rule generator called CBA-RG and a classifier builder called CBA-CB. For CBA-RG part, all frequent ruleitems are generated by using the algorithm like association rule mining process. For CBA-CB

part, all frequent ruleitems from the CBA-RG are ranked in decreasing order according to the following criteria.

Given two rules, r_i and r_j. r_i is ranked higher than r_j if

1. $conf(r_i) > conf(r_j)$, or

2. $conf(r_i) = conf(r_j)$, but $sup(r_i) > sup(r_j)$, or

3. $conf(r_i) = conf(r_j)$ and $sup(r_i) = sup(r_j)$, but r_i is generated before rj.

A training dataset called database consists of transactions. In process of selecting the rules into the classifier, the algorithm iterates through each rule starting from the first order rule to find all transactions containing all items in the antecedence of the current rule (covered by the rule). If at least one transaction covered by the rule is classified correctly by this rule, the rule is selected into the classifier and all of these transactions covered the rule are removed from the database, otherwise the rule is pruned. This process terminates when either all of rules are considered or no transactions are left in the database. In addition, a default class is selected by the majority class in the remaining transactions.

In classifying an unseen case, that case is predicted as a class by the consequence of the first rule covering the case. The default class is used to classify when no covering rules in the classifier can be used (Srisawat and Kijsirikul, 2008).

5.5 Heuristics Web Page Analysis

Heuristic phishing solutions look for specific techniques and patterns used by phishers. The techniques for analyzing website pages involve examining the properties of the web page and all its features and patterns to distinguish between phishing, suspicious and

legitimate websites. Page properties are typically derived and extracted from the website page's contents as HTML tags, URL address and Java Script source code. Examples of properties are the existence of password fields, SSL certificate, the number of links, and the DNS domain name**.**

For our study, we aimed to determine these properties and features, and their applicability as reasonable candidates for use with the associative classification fuzzy machine learning technique, for constructing and building a phishing websites detection model, to distinguish between phishing, suspicious and legitimate websites.

5.6 Mining Phishing Detection Data

Data mining is the automated extraction of previously unrealized information from large data sources for the purpose of supporting actions. The rapid development in data mining has made available a wide variety of algorithms, drawn from the field of statistics, pattern recognition, machine learning and databases ((Idris and Shanmugam, 2006). Association and classification rule algorithms find correlations between features or attributes used to describe a data set. Having specified the risk associated with an e-banking phishing website and its key phishing website characteristics and factor indicators, the next step is to specify how the different features of the e-banking phishing website are related and associated with one another. Experts provide fuzzy rules in the form of ***if…then*** statements that relate e-banking phishing website probability to various levels of key phishing characteristic indicators based on their knowledge and experience. On that matter, and instead of just employing an expert system, we utilised data mining classification and association rule approaches in our new e-banking phishing website detection model to automatically find significant

classified rules that find and control all associations and correlations between different patterns of phishing characteristics in the e-banking phishing website archive data. Particularly, we used a number of different existing data mining association and classification techniques implemented within WEKA (WEKA, 2006) and CBA packages (Liu, et al., 1998). JRip (Witten and Frank, 2005) WEKA's implementation of RIPPER, PART (Witten and Frank, 2005), PRISM (Cendrowska, 1987) and C4.5 (Quinlan, 1996) classification algorithms are selected to discover the relationships between the selected different phishing features and their correlation with one another. They were conducted using the WEKA software system, which is an open Java source code for the data mining community that includes implementations of different methods for several different data mining tasks such as classification, association, and regression. Meanwhile, for the association classification algorithm, CBA were conducted using an implementation version provided in the work by Liu, et al., (1998). We have chosen these classification algorithms based on the different strategies they use to generate the rules and because their learned classifiers are easily understood by humans (Ciesielski and Lalani, 2003).

5.7 Experimental Setup

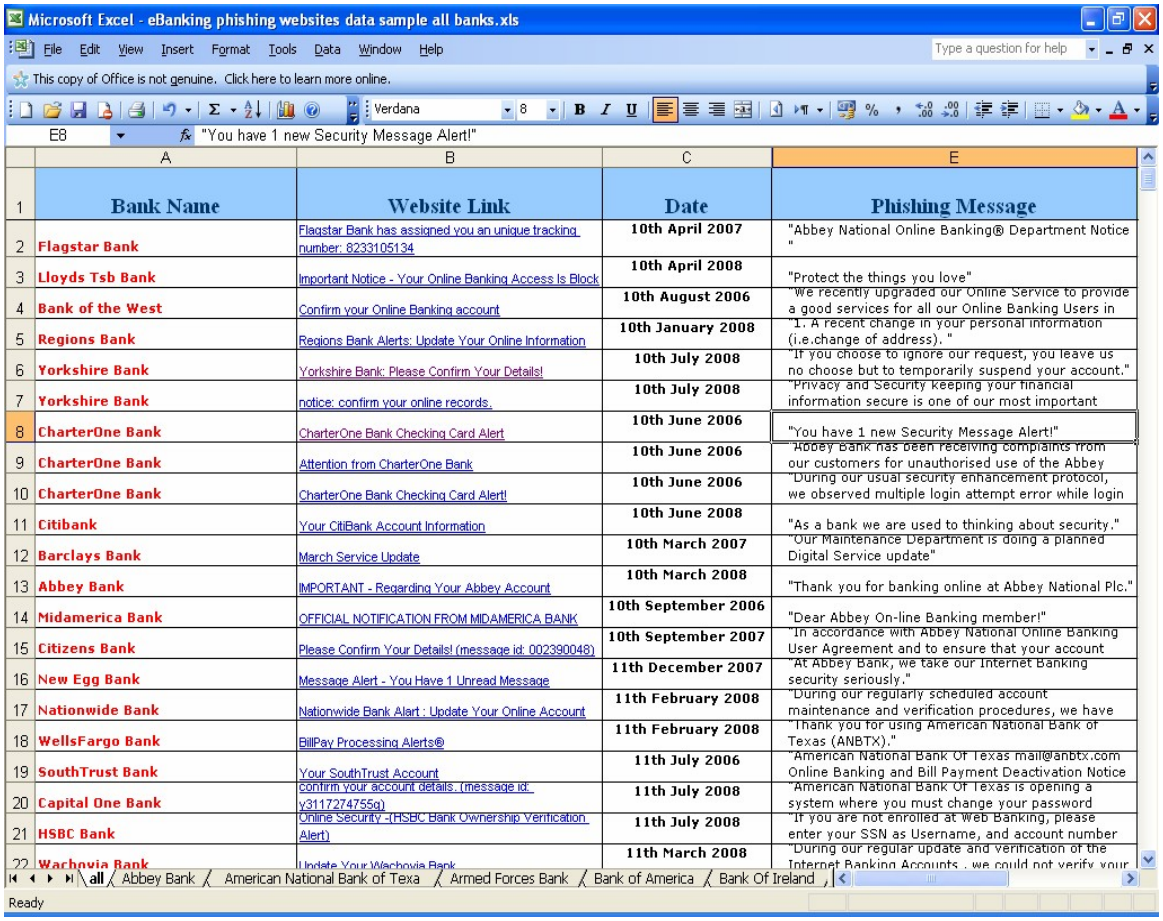
5.7.1 Phishing Dataset

Two publicly available phishing datasets were used to test our implementation: The "PhishTank" from the phishtank.com (Phishtank, 2008) and the Anti-Phishing Working Group(APWG) which maintains a "Phishing Archive" describing phishing attacks (APWG, 2008). We choose them as sources of phishing and suspicious websites since

the information from these sites is freely available and the amount of reported phishing sites is very large. The PhishTank database is considered one of the primary phishing-report collators. It records the URL for the suspect website that has been reported, the time of that report and, sometimes, further details such as screenshots of the website, as they are publicly available.

For our study, it was very necessary to collect a large number of phishing, suspicious and legitimate pages. That's why we managed to construct a dataset of 2178 phishing, suspicious and legitimate websites collected between January 10, 2006 and September 15, 2008, to be used in our research study, of which 731 are phishing and 711 are suspicious. This set of phishing and suspicious websites covers many of the newer trends and styles in designing and developing phishing websites. It was very important to make local copies of each phishing website using an application called Website eXtractor (InternetSoft, 2008), since most of them are only online for a short period of time, and extracting all phishing characters, factors and patterns from the website page's contents, HTML tags, Java Script code, and URL address requires some time.

We performed a cognitive walk-through on these datasets within this archive, and used a series of short Java scripts to programmatically extract the phishing features, storing these in an excel sheet for quick reference as shown in Figure 5.1, Figure 5.2, Figure 5.3 and Figure 5.4 as examples.



Microsoft Excel - eBanking phishing websites data sample all banks.xls

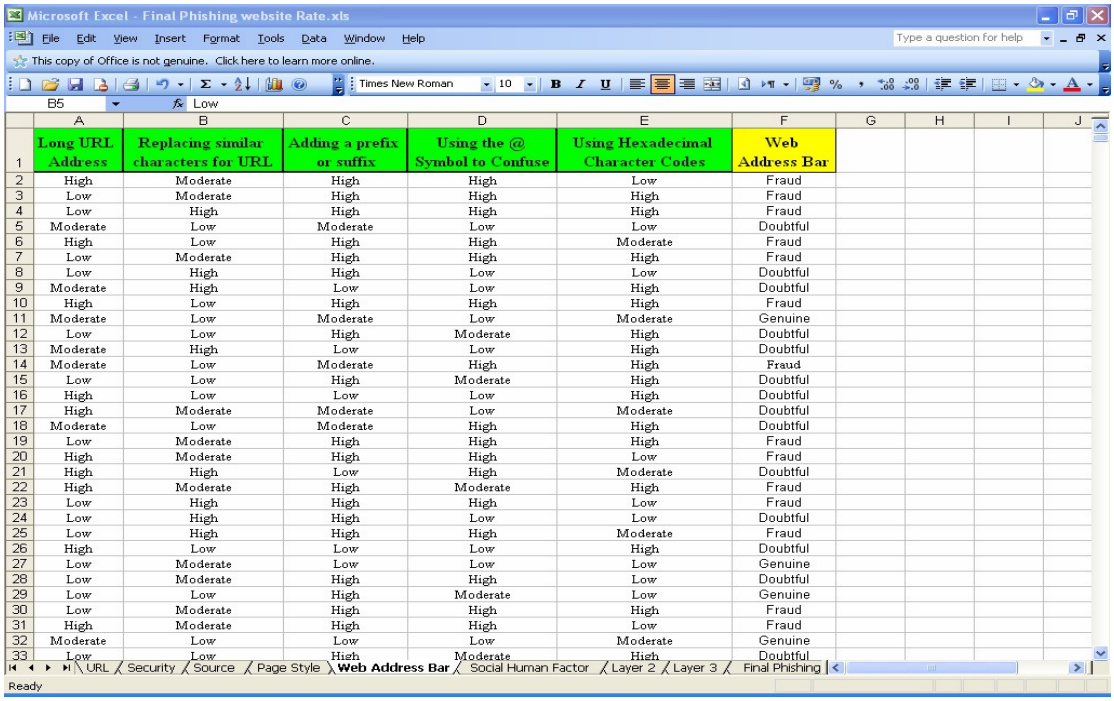
E8: "You have 1 new Security Message Alert!"

	Bank Name	Website Link	Date	Phishing Message
2	Flagstar Bank	Flagstar Bank has assigned you an unique tracking number: 8233105134	10th April 2007	"Abbey National Online Banking® Department Notice "
3	Lloyds Tsb Bank	Important Notice - Your Online Banking Access Is Block	10th April 2008	"Protect the things you love"
4	Bank of the West	Confirm your Online Banking account	10th August 2006	"We recently upgraded our Online Service to provide a good services for all our Online Banking Users in
5	Regions Bank	Regions Bank Alerts: Update Your Online Information	10th January 2008	"1. A recent change in your personal information (i.e.change of address). "
6	Yorkshire Bank	Yorkshire Bank: Please Confirm Your Details!	10th July 2008	"If you choose to ignore our request, you leave us no choose but to temporarily suspend your account."
7	Yorkshire Bank	notice: confirm your online records.	10th July 2008	"Privacy and Security keeping your financial information secure is one of our most important
8	CharterOne Bank	CharterOne Bank Checking Card Alert	10th June 2006	"You have 1 new Security Message Alert!"
9	CharterOne Bank	Attention from CharterOne Bank	10th June 2006	"Abbey Bank has been receiving complaints from our customers for unauthorised use of the Abbey
10	CharterOne Bank	CharterOne Bank Checking Card Alert!	10th June 2006	"During our usual security enhancement protocol, we observed multiple login attempt error while login
11	Citibank	Your CitiBank Account Information	10th June 2008	"As a bank we are used to thinking about security."
12	Barclays Bank	March Service Update	10th March 2007	"Our Maintenance Department is doing a planned Digital Service update"
13	Abbey Bank	IMPORTANT - Regarding Your Abbey Account	10th March 2008	"Thank you for banking online at Abbey National Plc."
14	Midamerica Bank	OFFICIAL NOTIFICATION FROM MIDAMERICA BANK	10th September 2006	"Dear Abbey On-line Banking member!"
15	Citizens Bank	Please Confirm Your Details! (message id: 002390048)	10th September 2007	"In accordance with Abbey National Online Banking User Agreement and to ensure that your account
16	New Egg Bank	Message Alert - You Have 1 Unread Message	11th December 2007	"At Abbey Bank, we take our Internet Banking security seriously."
17	Nationwide Bank	Nationwide Bank Alart : Update Your Online Account	11th February 2008	"During our regularly scheduled account maintenance and verification procedures, we have
18	WellsFargo Bank	BillPay Processing Alerts®	11th February 2008	"Thank you for using American National Bank of Texas (ANBTX)."
19	SouthTrust Bank	Your SouthTrust Account	11th July 2006	"American National Bank Of Texas mail@anbtx.com Online Banking and Bill Payment Deactivation Notice
20	Capital One Bank	confirm your account details. (message id: y3117274755g)	11th July 2008	"American National Bank Of Texas is opening a system where you must change your password
21	HSBC Bank	Online Security -(HSBC Bank Ownership Verification Alert)	11th July 2008	"If you are not enrolled at Web Banking, please enter your SSN as Username, and account number
22	Wachovia Bank	Update Your Wachovia Bank	11th March 2008	"During our regular update and verification of the Internet Banking Accounts , we could not verify your

Sheet tabs: all | Abbey Bank | American National Bank of Texa | Armed Forces Bank | Bank of America | Bank Of Ireland

Ready

Figure 5.1: Sample of extracted e-banking phishing websites with its links and details



Microsoft Excel - Final Phishing website Rate.xls

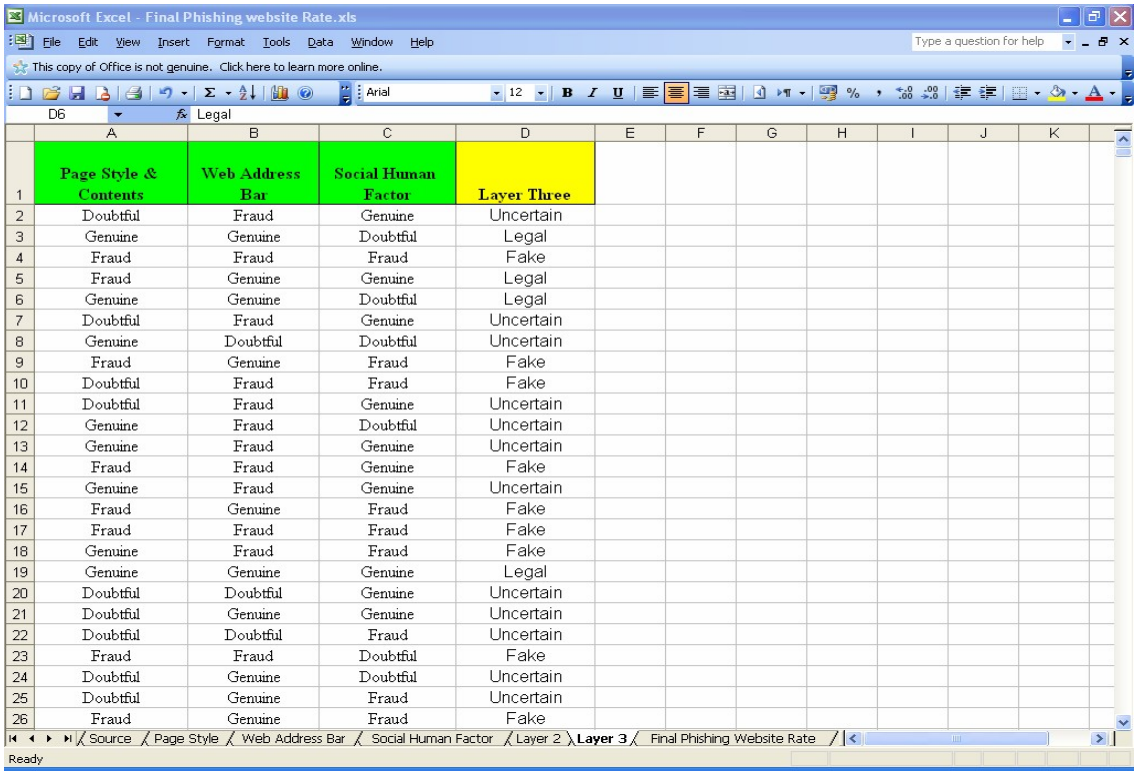
B5: Low

	Long URL Address	Replacing similar characters for URL	Adding a prefix or suffix	Using the @ Symbol to Confuse	Using Hexadecimal Character Codes	Web Address Bar
2	High	Moderate	High	High	Low	Fraud
3	Low	Moderate	High	High	High	Fraud
4	Low	High	High	High	High	Fraud
5	Moderate	Low	Moderate	Low	Low	Doubtful
6	High	Low	High	High	Moderate	Fraud
7	Low	Moderate	High	High	High	Fraud
8	Low	High	High	Low	Low	Doubtful
9	Moderate	High	Low	Low	High	Doubtful
10	High	Low	High	High	High	Fraud
11	Moderate	Low	Moderate	Low	Moderate	Genuine
12	Low	Low	High	Moderate	High	Doubtful
13	Moderate	High	Low	Low	High	Doubtful
14	Moderate	Low	Moderate	High	High	Fraud
15	Low	Low	High	Moderate	High	Doubtful
16	High	Low	Low	Low	High	Doubtful
17	High	Moderate	Moderate	Low	Moderate	Doubtful
18	Moderate	Low	Moderate	High	High	Doubtful
19	Low	Moderate	High	High	High	Fraud
20	High	Moderate	High	High	Low	Fraud
21	High	High	Low	High	Moderate	Doubtful
22	High	Moderate	High	Moderate	High	Fraud
23	Low	High	High	High	Low	Fraud
24	Low	High	High	Low	Low	Doubtful
25	Low	High	High	High	Moderate	Fraud
26	High	Low	Low	Low	High	Doubtful
27	Low	Moderate	Low	Low	Low	Genuine
28	Low	Moderate	High	High	Low	Doubtful
29	Low	Low	High	Moderate	Low	Genuine
30	Low	Moderate	High	High	High	Fraud
31	High	Moderate	High	High	Low	Fraud
32	Moderate	Low	Low	Low	Moderate	Genuine
33	Low	Low	High	Moderate	High	Doubtful

Sheet tabs: URL | Security | Source | Page Style | Web Address Bar | Social Human Factor | Layer 2 | Layer 3 | Final Phishing

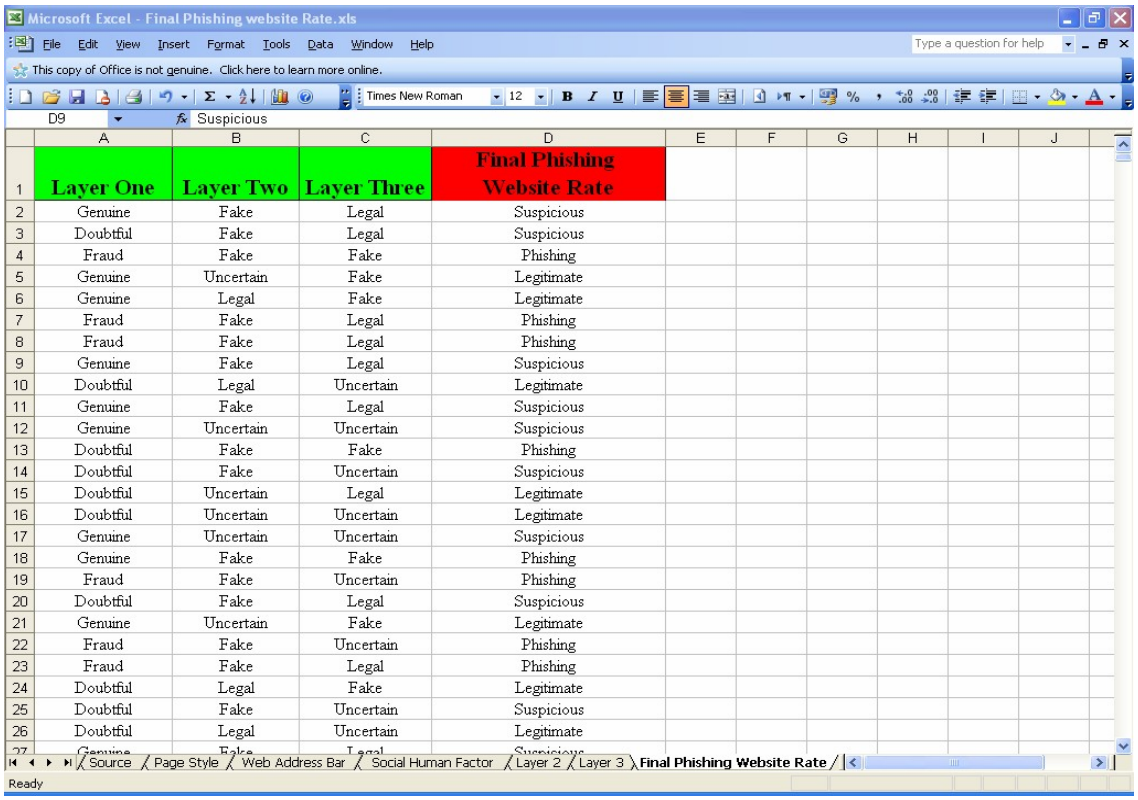
Ready

Figure 5.2: Linguistic values for phishing features related to web address bar criteria



	A	B	C	D
1	Page Style & Contents	Web Address Bar	Social Human Factor	Layer Three
2	Doubtful	Fraud	Genuine	Uncertain
3	Genuine	Genuine	Doubtful	Legal
4	Fraud	Fraud	Fraud	Fake
5	Fraud	Genuine	Genuine	Legal
6	Genuine	Genuine	Doubtful	Legal
7	Doubtful	Fraud	Genuine	Uncertain
8	Genuine	Doubtful	Doubtful	Uncertain
9	Fraud	Genuine	Fraud	Fake
10	Doubtful	Fraud	Fraud	Fake
11	Doubtful	Fraud	Genuine	Uncertain
12	Genuine	Fraud	Doubtful	Uncertain
13	Genuine	Fraud	Genuine	Uncertain
14	Fraud	Fraud	Genuine	Fake
15	Genuine	Fraud	Genuine	Uncertain
16	Fraud	Genuine	Fraud	Fake
17	Fraud	Fraud	Fraud	Fake
18	Genuine	Fraud	Fraud	Fake
19	Genuine	Genuine	Genuine	Legal
20	Doubtful	Doubtful	Genuine	Uncertain
21	Doubtful	Genuine	Genuine	Uncertain
22	Doubtful	Doubtful	Fraud	Uncertain
23	Fraud	Fraud	Doubtful	Fake
24	Doubtful	Genuine	Doubtful	Uncertain
25	Doubtful	Genuine	Fraud	Uncertain
26	Fraud	Genuine	Fraud	Fake

Figure 5.3: Linguistic values for phishing criteria related to layer three



	A	B	C	D
1	Layer One	Layer Two	Layer Three	Final Phishing Website Rate
2	Genuine	Fake	Legal	Suspicious
3	Doubtful	Fake	Legal	Suspicious
4	Fraud	Fake	Fake	Phishing
5	Genuine	Uncertain	Fake	Legitimate
6	Genuine	Legal	Fake	Legitimate
7	Fraud	Fake	Legal	Phishing
8	Fraud	Fake	Legal	Phishing
9	Genuine	Fake	Legal	Suspicious
10	Doubtful	Legal	Uncertain	Legitimate
11	Genuine	Fake	Legal	Suspicious
12	Genuine	Uncertain	Uncertain	Suspicious
13	Doubtful	Fake	Fake	Phishing
14	Doubtful	Fake	Uncertain	Suspicious
15	Doubtful	Uncertain	Legal	Legitimate
16	Doubtful	Uncertain	Uncertain	Legitimate
17	Genuine	Uncertain	Uncertain	Suspicious
18	Genuine	Fake	Fake	Phishing
19	Fraud	Fake	Uncertain	Phishing
20	Doubtful	Fake	Legal	Suspicious
21	Genuine	Uncertain	Fake	Legitimate
22	Fraud	Fake	Uncertain	Phishing
23	Fraud	Fake	Legal	Phishing
24	Doubtful	Legal	Fake	Legitimate
25	Doubtful	Fake	Uncertain	Suspicious
26	Doubtful	Legal	Uncertain	Legitimate

Figure 5.4: Linguistic values of the three layers for the final phishing website detection rate

For the legitimate portion of the dataset, we used 736 websites collected from very popular official internet banking websites and other financial institutions' websites such as Bank of America, Citi Bank, HSBC Bank, Yorkshire Bank, Barclays Bank, Bank of Jordan, Ahli bank, Gulf Bank and many others. Hence, in total our dataset contains 2178 websites, of which 34% are legitimate. Table 5.1 and Figure 5.5 show the distribution and the percentages of these complete datasets.

Table 5.1: Dataset distribution percentage

Data Set	No. of Websites	Percentage (%)
Legitimate	736	34%
Phishing	731	33%
Suspicious	711	33%
Total	2178	100%

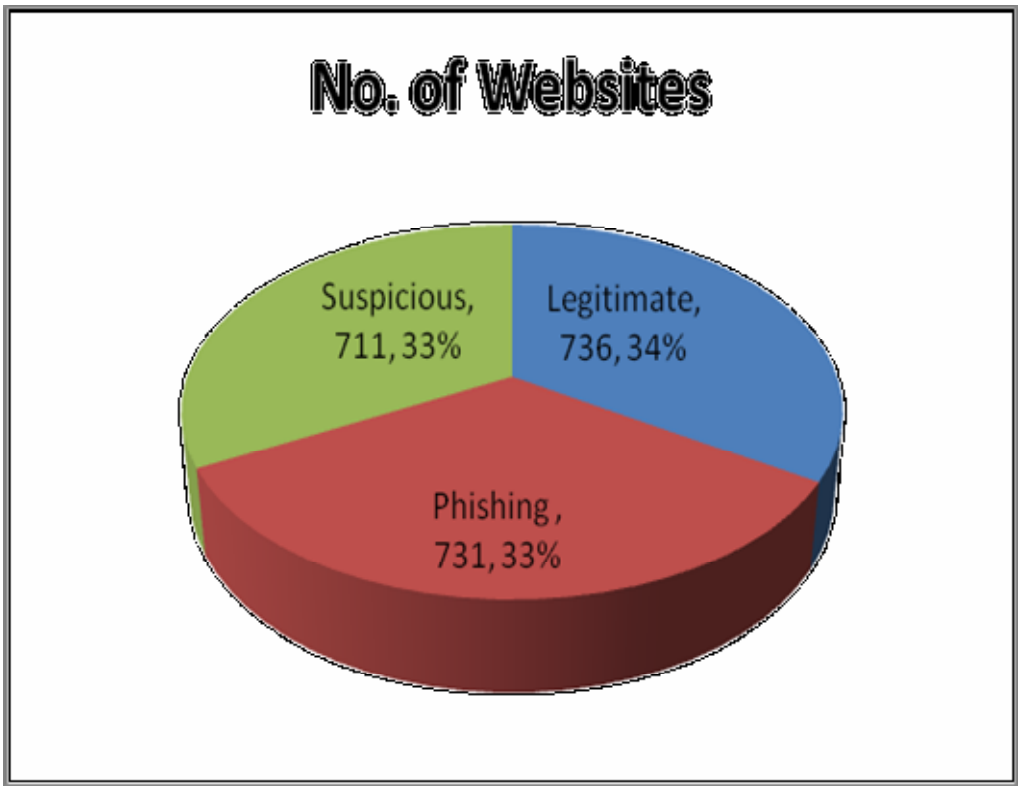


Figure 5.5: Dataset distribution percentage chart

5.7.2 Phishing Website Extracted Features and Patterns

Our overall approach centres on extracting the information, features and patterns that are most relevant for identifying phishing websites from our collected datasets. We do this by looking at features from each phishing website from the phishing dataset collection. We managed, as stated in chapter 4, to gather 27 phishing features and factors that can be used to identify phishing websites, representing different structural characteristics and several styles used by phishers to lure victims by making phishing websites look legitimate. This involves extracting data directly present in the phishing website, as well as collecting information from external sources like phishing experts and surveys. We tried to extract them from the website page's contents, HTML tags, Java Script code, and URL address. We analyzed all phishing, suspicious and legitimate web pages datasets, assigning concrete values to the properties for each page. Using the collected data as training input and testing input, we applied Associative Classification machine learning techniques to generate phishing web page classifier rules.

Our goal is to gather information about the strategies that are used by attackers and to formulate assumptions about classifying and categorizing all the different e-banking phishing attacks techniques. By thoroughly investigating these phishing attacks we've created a dataset containing information about the different techniques and methods that have been used and how the usage of these techniques has been correlated, associated and utilised for creating phishing websites.

5.7.3 Mining e-banking Phishing Considerations

There are a number of considerations posed in making a post hoc classification of e-banking phishing websites. Most of them only apply to the e-banking phishing websites

data and materialize as a form of information, which has the net effect of increasing the false negative rate. The age of the dataset is the most significant problem, which is particularly relevant with the phishing corpus. e-Banking phishing websites are short-lived, often lasting only in the order of 48 hours. Some of our features cannot therefore be extracted from older websites, making it difficult to conduct our tests. The average phishing site stays live for approximately 2.25 days (FDIC, 2004). Furthermore, the process of transforming the original e-banking phishing website archives into record feature datasets is not without error. It requires the use of heuristics at several steps. Thus, high accuracy from the data mining algorithms cannot be expected. However, the evidence supporting the 'golden nuggets' comes from a number of different algorithms and feature sets and we believe it is compelling (Fette et al., 2006).

5.8 Utilisation of Different DM Classification Algorithms

In classification problems, a classifier tries to learn several feature variables as inputs to predict an output. In the case of phishing website classification, a classifier rule tries to classify a website as phishing, suspicious or legitimate by learning certain characteristics, features and patterns in the website. In the following section, we briefly describe the classifiers used in our experiments.

The practical part of this study utilises five different common DM algorithms (C4.5, Ripper, Part, Prism and CBA). Our choice of these methods is based on the different strategies they use in learning rules from datasets (Misch, 2006). The C4.5 algorithm employs a divide-and-conquer approach, while the RIPPER algorithm uses a separate-and-conquer approach. We applied the J48 algorithm to extract a decision tree that can

classify web pages as legitimate, suspicious or phishing. J48 is an implementation of the classic C4.5 decision tree algorithm in Weka, a well-known data mining tool. We selected the C4.5 classifier since it provides an intuitive insight into which features are important in classifying a dataset, and it is known to work well for a wide range of classification problems (Ludl et al., 2007). The choice of PART algorithm is based on the fact that it combines both approaches to generate a set of rules. It adapts separate-and-conquer to generate a set of rules and uses divide-and-conquer to build partial decision trees. The way PART builds and prunes a partial decision tree is similar to the C4.5 implementation with a difference which can be explained as follows: C4.5 generates one decision tree and uses pruning techniques to simplify it; each path from the root node to one of the leaves in the tree represents a rule. On the other hand, PART avoids the simplification process by building up partial decision trees and choosing only one path in each one of them to derive a rule. Once the rule is generated, all instances are associated with it, and the partial tree is discarded. PRISM is a classification rule which can only deal with nominal attributes and which doesn't do any pruning. It implements a top-down (general to specific) sequential-covering algorithm that employs a simple accuracy-based metric to pick an appropriate rule antecedent during rule construction. Finally, CBA algorithm employs association rule mining to learn the classifier and then adds pruning and prediction steps. CBA utilises database coverage pruning to decrease the number of rules. It is worth noting that, without adding constraints to the rule discovery, the very large numbers of rules make it impossible for humans to understand the classifier. This pruning technique tests the generated rules against the training dataset, and only high quality rules that cover at least one training instance not considered by other higher ranked rules are kept for later classification.

There are two important advantages of the classifier using the CBA algorithm. First, as the classifier is a set of rules, it is easy for the user to understand and interpret the prediction of phishing susceptibility for a new case. Second, CBA has the ability to handle literal attributes to construct the classifier (Srisawat and Kijsirikul, 2004). This results in a classification approach named 'associative classification' (Thabtah, et al., 2005). Experiments were conducted using stratified tenfold cross-validation (which is set as default in Weka). In cross-validation, the training dataset is divided randomly into *10* blocks, each block is held out once, and the classifier is trained on the remaining *9* blocks; then its error rate is evaluated on the holdout block. Thus, the learning procedure is executed ten times on slightly different training datasets (Witten and Frank, 2005).

5.9 Proposed Intelligent Phishing Website Detection Model

Based on the collected website properties and patterns from the dataset defined in the previous section, we build an associative classification fuzzy model that attempts to use these properties to distinguish between phishing and legitimate websites. The phishing website detection intelligent model utilises fuzzy logic along with data mining association classification techniques and algorithms. It proposes a mechanism to automate the rule generation process and reduce the human intervention represented by an expert's past knowledge and experience of phishing. The system uses a simple data-mining association classification algorithm to identify the features of phishing websites and their relationships with one another, to produce the most proper classification rules. It will be integrated with the created fuzzy rules produced from fuzzy sets, and inserted into the fuzzy inference engine for the final phishing websites detection rate. The

inference engine works based on the Mamdani inference mechanism since it is most suited to our model architecture. The system architecture shown in Figure 5.6 builds class prediction models for identifying and detecting phishing website attacks.

From the phishing website archive data details, we undertake some data pre-processing for extracting specific patterns and attributes, establishing relevant features of the phishing patterns attack. Attributes are represented by names that will be used as linguistic variables by the Data Miner and the Fuzzy Inference Engine**.** Once attributes of relevant websites have been defined and the phishing dataset identified, training subset data and test subset data are constructed. Training subset data are then used as input data for the association and classification algorithms and techniques, producing AC phishing data miner, which allows the efficient processing of phishing features. The data miner is capable of discovering association and classification phishing rules and their relationships with one another. The rules that meet the confidence and support constraints are considered as input. They are then tested using the test subset data. Classified rules generated from classification algorithms are used as fuzzy rules, to be combined with external expert rules. Data Analyzer is employed to compute configuration parameters that regulate the operation of association classification algorithm and fuzzy inference engine for the final production of the phishing website indicator rate.

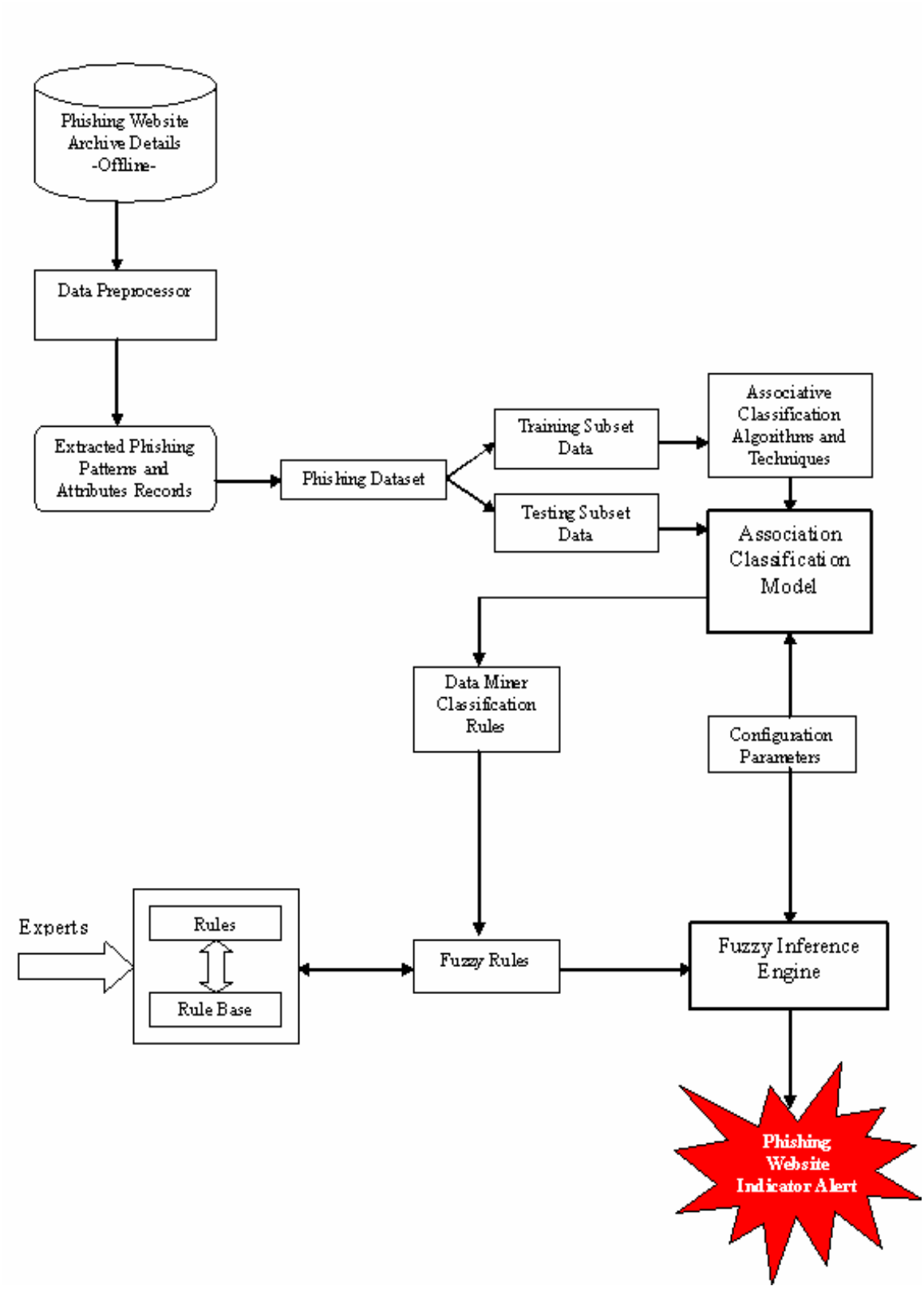


Figure 5.6: Architecture of the intelligent association classification mining fuzzy model for phishing website detection

5.10 Generated Classification Rules for Criteria and Layers

We will now show a sample of rules generated by selected associative classification and classification algorithms. We used CBA application for generating AC rules, and WEKA application for implementing the different classification algorithms (JRIP, PART, PRISM, J48) for generating all classifier rules, which will all be integrated by the fuzzy inference engine to produce more accurate results for the final phishing website detection rate.

Following this, we will demonstrate and analyze the classified rules for URL & Domain Identity and for Security & Encryption only. The other classified criteria rules are shown in Appendix A in order to avoid repetition.

5.10.1 Rules for URL & Domain Identity (Layer One)

Association Classification rules for layer one (URL & Domain Identity Criteria) consist of five components (Using the IP Address, Abnormal Request URL, Abnormal URL of Anchor, Abnormal DNS Record, Abnormal URL). Fuzzy variables are High, Moderate and Low for inputs, and Fraud, Doubtful and Genuine for the output class.

CBA Rules:

Num of Test Case: 2178; Correct Prediction: 2005; Error Rate: 7.943 %
MinSup: 10.000%, MinConf: 80.000%

Rule 1: Abnormal_DNS_record = High
Using_the_IP_Address = High -> class = Fraud
Rule 2: Abnormal_DNS_record = High
Abnormal_Request_URL = High -> class = Fraud
Rule 3: Abnormal_URL = Low
Abnormal_DNS_record = Low
Abnormal_URL_Anchor = Low -> class = Genuine
Rule 4: Abnormal_URL = Low
Abnormal_DNS_record = Low -> class = Genuine
Rule 5: Abnormal_URL = Low
Abnormal_URL_Anchor = Low -> class = Genuine
Rule 6: Abnormal_DNS_record = Low
Abnormal_Request_URL = Low -> class = Genuine
Rule 7: Abnormal_URL = Low
Abnormal_Request_URL = Low -> class = Genuine
Rule 8: Abnormal_DNS_record = High

Abnormal_URL_Anchor = High -> class = Fraud
Rule 9: Abnormal_URL = Moderate -> class = Doubtful
Rule 10: Abnormal_URL_Anchor = Moderate -> class = Doubtful
Rule 11: Abnormal_DNS_record = Moderate -> class = Doubtful
Rule 12: Abnormal_Request_URL = Moderate -> class = Doubtful

Number of Rules: 12

Sample of JRIP rules:

==========

Correctly Classified Instances	2175	99.8623 %
Incorrectly Classified Instances	3	0.1377 %
Mean absolute error	0.0018	

=== Confusion Matrix ===

a	b	c	<-- classified as
705	0	0	a = Genuine
1	903	0	b = Doubtful
0	2	567	c = Fraud

From the confusion matrix we can conclude that all 705 rules classified as "Genuine" were never misclassified or contradicted by any other rules in the dataset. But, regarding "Doubtful" classified rules, all 903 were classified correctly and as expected, and just one was misclassified as "Genuine" instead. Finally, for the "Fraud" classified rules, 567 were classified correctly and 2 were misclassified as "Doubtful".

(Abnormal_DNS_record = High) and (Using_the_IP_ddress = High) =>
URL_&_Domain_Identity_Criteria_Phishing_Risk=Fraud (312.0/0.0)
(Abnormal_DNS_record = High) and (Abnormal_Request_URL = High) =>
URL_&_Domain_Identity_Criteria_Phishing_Risk=Fraud (155.0/0.0)
(Abnormal_URL = Low) and (Abnormal_Request_URL = Low) and (Using_the_IP_ddress = Low) =>
URL_&_Domain_Identity_Criteria_Phishing_Risk=Genuine (68.0/0.0)
(Abnormal_URL_Anchor = Low) and (Abnormal_URL = Low) =>
URL_&_Domain_Identity_Criteria_Phishing_Risk=Genuine (308.0/0.0)
(Abnormal_Request_URL = Low) and (Abnormal_DNS_record = Low) and (Abnormal_URL =
Moderate) => URL_&_Domain_Identity_Criteria_Phishing_Risk=Genuine (115.0/0.0)

Number of Rules: 14

Sample of PART decision list

Abnormal_URL = Moderate AND
Abnormal_Request_URL = Low: Genuine (115.0)
Abnormal_URL = Low AND
Abnormal_URL_Anchor = Low: Genuine (308.0)
Abnormal_DNS_record = High AND
Abnormal_URL_Anchor = Low: Fraud (156.0)
Abnormal_DNS_record = High AND
Abnormal_URL_Anchor = Moderate: Doubtful (156.0/1.0)

Abnormal_DNS_record = High AND
Abnormal_Request_URL = High: Fraud (155.0)

Number of Rules: 22

Sample of Prism rules

```
If Abnormal_URL_Anchor = Low
  and Using_the_IP_ddress = Low
  and Abnormal_URL = Moderate
  and Abnormal_Request_URL = High
  and Abnormal_DNS_record = Low then Genuine
If Using_the_IP_ddress = Moderate
  and Abnormal_Request_URL = Low
  and Abnormal_URL_Anchor = Low then Genuine
If Abnormal_DNS_record = Moderate
  and Abnormal_URL = Moderate
  and Abnormal_URL_Anchor = High then Doubtful
If Abnormal_DNS_record = High
  and Abnormal_URL = High
  and Using_the_IP_ddress = Moderate
  and Abnormal_Request_URL = Low
  and Abnormal_URL_Anchor = Moderate then Fraud
If Using_the_IP_ddress = High
  and Abnormal_URL_Anchor = High
  and Abnormal_URL = Low
  and Abnormal_Request_URL = Moderate
  and Abnormal_DNS_record = Moderate then Fraud
```

Number of Rules: 39

Sample of J48 pruned tree

```
Abnormal_DNS_record = Low
|  Abnormal_URL = Low
|  |  Abnormal_URL_Anchor = Low: Genuine (274.0)
|  |  Abnormal_URL_Anchor = Moderate
|  |  |  Using_the_IP_ddress = Low: Genuine (0.0)
|  |  |  Using_the_IP_ddress = Moderate: Genuine (69.0)
|  |  |  Using_the_IP_ddress = High: Doubtful (35.0)
|  |  Abnormal_URL_Anchor = High
|  |  |  Abnormal_Request_URL = Low: Genuine (0.0)
|  |  |  Abnormal_Request_URL = Moderate: Genuine (36.0)
|  |  |  Abnormal_Request_URL = High: Doubtful (35.0)
|  Abnormal_URL = Moderate
|  |  Abnormal_Request_URL = Low: Genuine (115.0)
|  |  Abnormal_Request_URL = Moderate
|  |  |  Abnormal_URL_Anchor = Low: Genuine (22.0)
|  |  |  Abnormal_URL_Anchor = Moderate: Doubtful (23.0)
```

Number of Leaves: 43
Size of the tree: 64

We recorded the prediction accuracy and the number of rules generated by the classification algorithms for URL & Domain Identity criteria in Table 5.2.

Table 5.2: Classification prediction accuracy and rules number for URL criteria

	URL & Domain Identity Criteria (Layer One)				
Algorithms	**JRIP R.I.P.P.E.R**	**PART**	**PRISM**	**C4.5 Decision Tree(J48)**	**CBA**
Test Mode	**10 FOLD CROSS VALIDATION**				
Attributes	**Using the IP Address, Abnormal Request URL, Abnormal URL of Anchor, Abnormal DNS Record, Abnormal URL CLASS**				
Number of Rules	**14**	**22**	**39**	**43 Leaves Tree size 64**	**12**
Correctly Classified	**2175 (99.862 %)**				**2005 (92.057%)**
Incorrectly Classified	**3 (0.138 %)**				**173 (7.943%)**
Number of Instances	**2187**				

5.10.2 Rules for Security and Encryption Criteria

Association Classification rules for Security and Encryption Criteria, which consist of four components (Using SSL Certificate, Certification Authority, Abnormal Cookie, and Distinguished Names Certificate (DN)). Fuzzy variables are High, Moderate and Low for inputs and Fraud, Doubtful and Genuine for the output class.

CBA Rules:

Num of Test Case: 2178; Correct Prediction: 2091; Error Rate: 3.994%
MinSup: 10.000%, MinConf: 80.000%

Rule 1: Distinguished_Names_Certificate[DN] = Low
Abnormal_Cookie = Low
Certification_authority = High -> class = Doubtful
Rule 2: Distinguished_Names_Certificate[DN] = High
Certification_authority = High -> class = Fraud
Rule 3: Distinguished_Names_Certificate[DN] = Low
Using_SSL_certificate = High -> class = Doubtful
Rule 4: Abnormal_Cookie = High
Certification_authority = High -> class = Fraud
Rule 5: Distinguished_Names_Certificate[DN] = Low
Certification_authority = Low
Using_SSL_certificate = Moderate -> class = Genuine
Rule 6: Distinguished_Names_Certificate[DN] = High
Using_SSL_certificate = High -> class = Fraud
Rule 7: Distinguished_Names_Certificate[DN] = High

Abnormal_Cookie = High
Using_SSL_certificate = Low -> class = Fraud
Rule 8: Certification_authority = High
Using_SSL_certificate = High -> class = Fraud
Rule 9: Abnormal_Cookie = High
Using_SSL_certificate = High -> class = Fraud
Rule 10: Using_SSL_certificate = Moderate -> class = Doubtful
Rule 11: Certification_authority = Low -> class = Doubtful
Rule 12: Abnormal_Cookie = Moderate -> class = Doubtful
Rule 13: Distinguished_Names_Certificate[DN] = Low -> class = Genuine
Rule 14: Abnormal_Cookie = Low -> class = Genuine

Num of Rules: 14

Sample of JRIP rules:

==========

Correctly Classified Instances	2169	99.5868 %
Incorrectly Classified Instances	9	0.4132 %
Mean absolute error	0.0055	

=== Confusion Matrix ===

a	b	c	<-- classified as
384	2	0	a = Genuine
3	930	2	b = Doubtful
0	2	855	c = Fraud

(Using_SSL_certificate = Moderate) and (Certification_authority = Low) and (Names_Certificate(DN) = Low) => Security_&_Encryption_Criteria_Phishing_Risk=Genuine (232.0/2.0)
(Certification_authority = Moderate) and (Abnormal_Cookie = Low) =>
Security_&_Encryption_Criteria_Phishing_Risk=Genuine (155.0/1.0)
(Names_Certificate(DN) = High) and (Abnormal_Cookie = High) =>
Security_&_Encryption_Criteria_Phishing_Risk=Fraud (467.0/0.0)
(Using_SSL_certificate = High) and (Certification_authority = High) =>
Security_&_Encryption_Criteria_Phishing_Risk=Fraud (234.0/0.0)

Number of Rules: 7

Sample of PART decision list

Names_Certificate(DN) = High AND
Abnormal_Cookie = High: Fraud (467.0)
Using_SSL_certificate = High AND
Names_Certificate(DN) = Low: Doubtful (233.0)
Using_SSL_certificate = High: Fraud (312.0)
Certification_authority = High AND
Abnormal_Cookie = Low: Doubtful (311.0/1.0)
Certification_authority = Low AND
Names_Certificate(DN) = Low: Genuine (156.0/1.0)

Number of Rules: 9

Sample of Prism rules

```
If Using_SSL_certificate = Moderate
   and Certification_authority = Low
   and Names_Certificate(DN) = Low
   and Abnormal_Cookie = Moderate then Genuine
If Abnormal_Cookie = Low
   and Certification_authority = Moderate
   and Names_Certificate(DN) = Low then Genuine
If Using_SSL_certificate = Moderate
   and Certification_authority = High
   and Abnormal_Cookie = Low
   and Names_Certificate(DN) = Low then Doubtful
If Abnormal_Cookie = High
   and Certification_authority = High
   and Using_SSL_certificate = Low
   and Names_Certificate(DN) = Low then Fraud
If Names_Certificate(DN) = High
   and Using_SSL_certificate = Moderate
   and Certification_authority = Low
   and Abnormal_Cookie = Moderate then Fraud
```

Number of Rules: 24

Sample of J48 pruned tree

```
Names_Certificate(DN) = Low
|   Abnormal_Cookie = Low
|   |   Certification_authority = Low
|   |   |   Using_SSL_certificate = Low: Doubtful (0.0)
|   |   |   Using_SSL_certificate = Moderate: Genuine (76.0/1.0)
|   |   |   Using_SSL_certificate = High: Doubtful (155.0)
|   |   Certification_authority = Moderate: Genuine (77.0)
|   |   Certification_authority = High: Doubtful (311.0/1.0)
|   Abnormal_Cookie = Moderate
|   |   Using_SSL_certificate = Low: Genuine (0.0)
|   |   Using_SSL_certificate = Moderate: Genuine (156.0/1.0)
|   |   Using_SSL_certificate = High: Doubtful (78.0)
|   Abnormal_Cookie = High
|   |   Certification_authority = Low: Doubtful (0.0)
|   |   Certification_authority = Moderate: Doubtful (78.0/1.0)
|   |   Certification_authority = High: Fraud (78.0/2.0)
Names_Certificate(DN) = Moderate
```

Number of Leaves: 19
Size of the tree: 28

We recorded the prediction accuracy and the number of rules generated by the classification algorithms for Security & Encryption Criteria in Table 5.3.

Table 5.3: Classification prediction accuracy and rules number for security criteria

	Security & Encryption Criteria				
Algorithms	**JRIP R.I.P.P.E.R**	**PART**	**PRISM**	**C4.5 Decision Tree(J48)**	**CBA**
Test Mode	**10 FOLD CROSS VALIDATION**				
Attributes	**Using SSL Certificate, Certification Authority, Abnormal Cookie, Distinguished Names Certificate (DN) CLASS**				
Number of Rules	**7**	**9**	**24**	**19 Leaves Tree size 28**	**14**
Correctly Classified	**2169 (99.587 %)**				**2091 (96.005%)**
Incorrectly Classified	**9 (0.413 %)**				**87 (3.994%)**
Number of Instances	**2187**				

5.10.3 Rules for Layer Two

Association Classification rules for Layer Two which consist of Two Criteria (Security Code & Encryption Criteria, Source Code & Java Script Criteria). Fuzzy variables are Fraud, Doubtful and Genuine for inputs and Fake, Uncertain and Legal for the output class.

CBA Rules:

Num of Test Case : 2178; Correct Prediction : 2170; Error Rate : 0.367%
MinSup: 10.000%, MinConf: 80.000%

Rule 1: Source_Code_&_Java_script = Genuine
Security_&_Encryption = Genuine -> class = Legal
Rule 2: Source_Code_&_Java_script = Fraud
Security_&_Encryption = Fraud -> class = Fake
Rule 3: Security_&_Encryption = Fraud -> class = Fake
Rule 4: Security_&_Encryption = Doubtful -> class = Fake
Rule 5: Source_Code_&_Java_script = Fraud
Security_&_Encryption = Genuine -> class = Uncertain
Rule 6: Security_&_Encryption = Genuine -> class = Legal

Num of Rules: 6

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

Correctly Classified Instances 2170 99.6327 %
Incorrectly Classified Instances 8 0.3673 %

Mean absolute error 0.0049
=== Confusion Matrix ===

a	b	c	<-- classified as
547	3	1	a = Legal
0	296	4	b = Uncertain
0	0	1327	c = Fake

(Source_Code_&_Java_script = Fraud) and (Security_&_Encryption = Genuine) =>
Layer_Two_Phishing_Risk=Uncertain (299.0/3.0)
(Security_&_Encryption = Genuine) => Layer_Two_Phishing_Risk=Legal (547.0/0.0)
=> Layer_Two_Phishing_Risk=Fake (1332.0/5.0)

Number of Rules : 3

PART decision list

Security_&_Encryption = Fraud: Fake (735.0/2.0)
Security_&_Encryption = Doubtful: Fake (597.0/3.0)
Source_Code_&_Java_script = Genuine: Legal (390.0)
Source_Code_&_Java_script = Fraud: Uncertain (299.0/3.0)
: Legal (157.0)

Number of Rules : 5

Sample of Prism rules

If Security_&_Encryption = Genuine
and Source_Code_&_Java_script = Genuine then Legal
If Source_Code_&_Java_script = Doubtful
and Security_&_Encryption = Genuine then Legal
If Security_&_Encryption = Genuine
and Source_Code_&_Java_script = Fraud then Legal
If Security_&_Encryption = Doubtful
If Security_&_Encryption = Fraud
and Source_Code_&_Java_script = Genuine then Uncertain
If Security_&_Encryption = Fraud
and Source_Code_&_Java_script = Fraud then Fake
If Security_&_Encryption = Fraud
and Source_Code_&_Java_script = Genuine then Fake

Number of Rules : 15

J48 pruned tree

Security_&_Encryption = Genuine
| Source_Code_&_Java_script = Genuine: Legal (390.0)
| Source_Code_&_Java_script = Doubtful: Legal (157.0)
| Source_Code_&_Java_script = Fraud: Uncertain (299.0/3.0)
Security_&_Encryption = Doubtful: Fake (597.0/3.0)
Security_&_Encryption = Fraud: Fake (735.0/2.0)

Number of Leaves : 5
Size of the tree : 7

We recorded the prediction accuracy and the number of rules generated by the classification algorithms for Layer Two in Table 5.4 and the chart of decision J48 tree

for layer two in Figure 5.7 which demonstrate the importance of security criteria inside layer two.

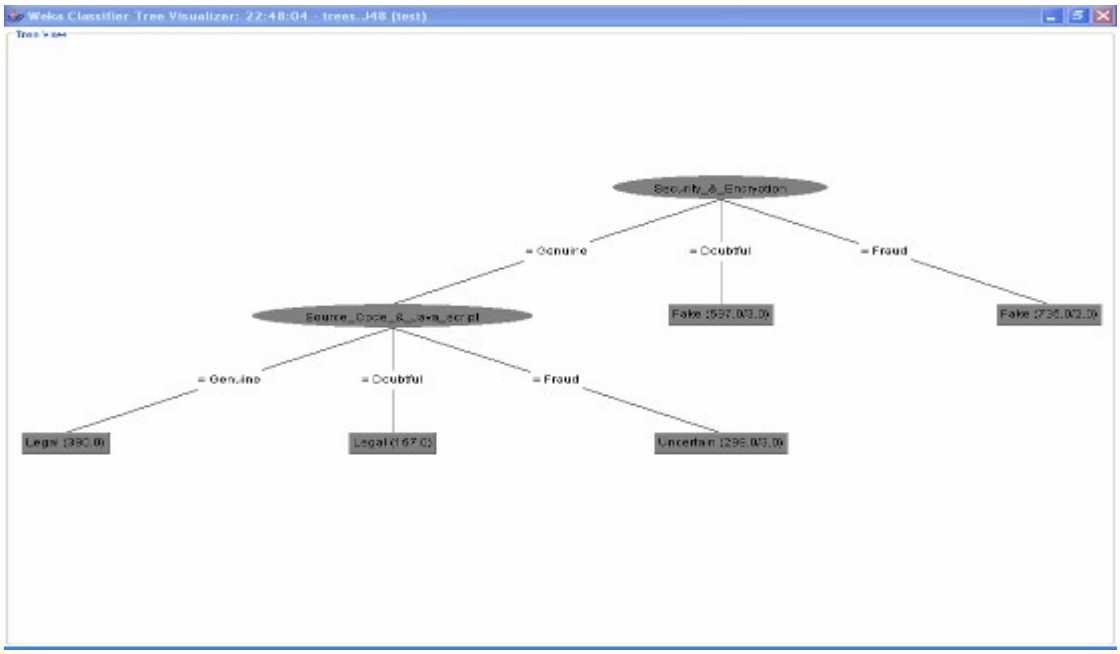


Figure 5.7: Chart of decision J48 tree for layer two

Table 5.4: Classification prediction accuracy and rules number for layer two

	Layer Two				
Algorithms	**JRIP R.I.P.P.E.R**	**PART**	**PRISM**	**C4.5 Decision Tree(J48)**	**CBA**
Test Mode	**10 FOLD CROSS VALIDATION**				
Attributes	**Security Code & Encryption, Source Code & Java Script CLASS**				
Number of Rules	**3**	**5**	**15**	**5 Leaves Tree size 7**	**6**
Correctly Classified	**2170 (96.633 %)**				**2170 (99.633 %)**
Incorrectly Classified	**8 (0.367%)**				**8 (0.367%)**
Number of Instances	**2187**				

5.10.4 Rules for Layer Three

Classification Association Classification rules for Layer Three which consist of Three Criteria (Page Style & Contents, Web Address Bar and Social Human Factor). Fuzzy

variables are Fraud, Doubtful and Genuine for inputs and Fake, Uncertain and Legal for the output class.

CBA Rules:

Num of Test Case : 2178; Correct Prediction : 2138; Error Rate : 1.837%
MinSup: 10.000%, MinConf: 80.000%

Rule 1:	Web_Address_Bar = Genuine Page_Style_&_Contents = Genuine	->	class = Legal
Rule 2:	Web_Address_Bar = Genuine Page_Style_&_Contents = Doubtful	->	class = Uncertain
Rule 3:	Social_Human_Factor = Fraud Web_Address_Bar = Fraud	->	class = Fake
Rule 4:	Web_Address_Bar = Fraud Page_Style_&_Contents = Fraud	->	class = Fake
Rule 5:	Social_Human_Factor = Fraud Page_Style_&_Contents = Fraud	->	class = Fake
Rule 6:	Social_Human_Factor = Genuine Page_Style_&_Contents = Doubtful	->	class = Uncertain
Rule 7:	Web_Address_Bar = Doubtful	->	class = Uncertain
Rule 8:	Social_Human_Factor = Genuine Web_Address_Bar = Genuine	->	class = Legal
Rule 9:	Social_Human_Factor = Genuine Web_Address_Bar = Fraud	->	class = Uncertain
Rule 10:	Web_Address_Bar = Genuine	->	class = Legal

Num of Rules: 10

Sample of JRIP rules:
==========

Correctly Classified Instances	2173	99.7704 %
Incorrectly Classified Instances	5	0.2296 %
Mean absolute error	0.003	

=== Confusion Matrix ===

a	b	c	<-- classified as
598	2	0	a = Legal
0	929	1	b = Uncertain
0	2	646	c = Fake

(Page_Style_&_Contents = Fraud) and (Social_Human_Factor = Fraud) => Layer_Three_Phishing_Risk=Fake (265.0/0.0)
(Web_Address_Bar = Fraud) and (Social_Human_Factor = Fraud) => Layer_Three_Phishing_Risk=Fake (173.0/0.0)
(Page_Style_&_Contents = Fraud) and (Web_Address_Bar = Fraud) => Layer_Three_Phishing_Risk=Fake (172.0/0.0)

(Web_Address_Bar = Genuine) and (Page_Style_&_Contents = Genuine) =>
Layer_Three_Phishing_Risk=Legal (390.0/0.0)
(Web_Address_Bar = Genuine) and (Page_Style_&_Contents = Fraud) and (Social_Human_Factor = Genuine) => Layer_Three_Phishing_Risk=Legal

Number of Rules : 8

Sample of PART decision list

Web_Address_Bar = Doubtful AND
Social_Human_Factor = Genuine: Uncertain (182.0/1.0)
Web_Address_Bar = Doubtful AND
Page_Style_&_Contents = Genuine: Uncertain (87.0/1.0)
Web_Address_Bar = Genuine AND
Page_Style_&_Contents = Genuine: Legal (218.0)
Page_Style_&_Contents = Fraud AND
Social_Human_Factor = Fraud: Fake (265.0)
Web_Address_Bar = Fraud AND
Social_Human_Factor = Fraud: Fake (173.0)

Number of Rules : 14

Sample of Prism rules

If Web_Address_Bar = Genuine
  and Page_Style_&_Contents = Genuine then Legal
If Web_Address_Bar = Genuine
  and Page_Style_&_Contents = Fraud
  and Social_Human_Factor = Genuine then Legal
If Page_Style_&_Contents = Doubtful
  and Social_Human_Factor = Doubtful
  and Web_Address_Bar = Fraud then Uncertain
If Web_Address_Bar = Fraud
  and Social_Human_Factor = Fraud then Fake
If Page_Style_&_Contents = Fraud
  and Web_Address_Bar = Fraud then Fake

Number of Rules : 22

Sample of J48 pruned tree

Web_Address_Bar = Genuine
| Page_Style_&_Contents = Genuine: Legal (390.0)
| Page_Style_&_Contents = Doubtful: Uncertain (284.0)
| Page_Style_&_Contents = Fraud
| | Social_Human_Factor = Genuine: Legal (152.0)
| | Social_Human_Factor = Doubtful: Legal (56.0)
| | Social_Human_Factor = Fraud: Fake (129.0)
Web_Address_Bar = Doubtful
| Social_Human_Factor = Genuine: Uncertain (182.0/1.0)
| Social_Human_Factor = Doubtful: Uncertain (79.0)
| Social_Human_Factor = Fraud
| | Page_Style_&_Contents = Genuine: Uncertain (57.0/1.0)
| | Page_Style_&_Contents = Doubtful: Uncertain (30.0/1.0)

| | Page_Style_&_Contents = Fraud: Fake (43.0)
Web_Address_Bar = Fraud

Number of Leaves : 17
Size of the tree : 25

We recorded the prediction accuracy and the number of rules generated by the classification algorithms for Layer Three in Table 5.5.

Table 5.5: Classification prediction accuracy and rules number for layer three

	Layer Three				
Algorithms	**JRIP R.I.P.P.E.R**	**PART**	**PRISM**	**C4.5 Decision Tree(J48)**	**CBA**
Test Mode	**10 FOLD CROSS VALIDATION**				
Attributes	**Page Style & Contents, Web Address Bar, Social Human Factor CLASS**				
Number of Rules	**8**	**14**	**22**	**17 Leaves Tree size 25**	**10**
Correctly Classified	**2173 (99.770%)**				**2138 (98.163%)**
Incorrectly Classified	**5 (0.229 %)**				**40 (1.837%)**
Number of Instances	**2187**				

5.10.5 Rules for Final Phishing Website Detection Rate

Association Classification rules for the Final Phishing Website Detection Rate which consists of Three Layers (Layer One, Layer Two and Layer Three). Fuzzy variables are Fraud, Doubtful and Genuine for Layer one input. For layer two and three, fuzzy variables are Fake, Uncertain and Legal and Phishing for inputs, Suspicious and Legitimate for the final output class.

CBA Rules:

Num of Test Case : 2178; Correct Prediction : 1972; Error Rate : 9.458%
MinSup: 10.000%, MinConf: 80.000%

Rule 1: Layer_Two = Fake
Layer_Three = Fake -> class = Phishing

Rule 2: Layer_Two = Fake
Layer_Three = Uncertain
Layer_One = Doubtful -> class = Suspicious
Rule 3: Layer_Two = Legal
Layer_One = Doubtful -> class = Legitimate
Rule 4: Layer_Two = Fake
Layer_One = Fraud -> class = Phishing
Rule 5: Layer_One = Fraud -> class = Phishing
Rule 6: Layer_Three = Uncertain
Layer_One = Genuine -> class = Legitimate
Rule 7: Layer_Two = Legal -> class = Legitimate
Rule 8: Layer_Two = Fake
Layer_Three = Legal -> class = Suspicious
Rule 9: Layer_One = Genuine -> class = Legitimate
Rule 10: Layer_One = Doubtful -> class = Legitimate

Num of Rules: 10

JRIP rules:

==========

Correctly Classified Instances	2046	93.9394 %
Incorrectly Classified Instances	132	6.0606 %
Mean absolute error	0.0693	

=== Confusion Matrix ===

a	b	c	<-- classified as
736	0	0	a = Legitimate
92	590	29	b = Suspicious
0	11	720	c = Phishing

(Layer_Two = Fake) and (Layer_One = Doubtful) and (Layer_Three = Uncertain) =>
Final_Phishing_Website_Rate=Suspicious (246.0/0.0)
(Layer_Three = Legal) and (Layer_Two = Fake) and (Layer_One = Doubtful) =>
Final_Phishing_Website_Rate=Suspicious (147.0/0.0)
(Layer_Three = Legal) and (Layer_One = Genuine) and (Layer_Two = Fake) =>
Final_Phishing_Website_Rate=Suspicious (112.0/0.0)
(Layer_One = Fraud) and (Layer_Two = Legal) => Final_Phishing_Website_Rate=Suspicious
(142.0/29.0)
(Layer_One = Fraud) => Final_Phishing_Website_Rate=Phishing (427.0/21.0)
(Layer_Two = Fake) and (Layer_Three = Fake) => Final_Phishing_Website_Rate=Phishing (296.0/0.0)
=> Final_Phishing_Website_Rate=Legitimate (808.0/72.0)

Number of Rules : 7

PART decision list

Layer_Two = Legal AND Layer_One = Doubtful: Legitimate (234.0)
Layer_One = Fraud AND Layer_Two = Fake: Phishing (344.0/21.0)
Layer_Three = Fake AND Layer_Two = Fake: Phishing (296.0)
Layer_One = Doubtful AND Layer_Two = Fake: Suspicious (393.0)
Layer_One = Genuine AND Layer_Two = Fake AND
Layer_Three = Uncertain: Legitimate (187.0/58.0)
Layer_One = Genuine AND Layer_Two = Legal: Legitimate (171.0)
Layer_Two = Fake: Suspicious (112.0)
Layer_Two = Legal AND Layer_Three = Uncertain: Suspicious (62.0)
Layer_One = Doubtful: Legitimate (117.0)

Layer_One = Genuine: Legitimate (99.0/14.0)
Layer_Two = Uncertain: Phishing (83.0)
Layer_Three = Fake: Phishing (42.0/13.0)
: Suspicious (38.0)

Number of Rules : 13

Prism rules

If Layer_Two = Legal and Layer_One = Doubtful then Legitimate
If Layer_Two = Uncertain and Layer_One = Doubtful then Legitimate
If Layer_Two = Legal and Layer_One = Genuine then Legitimate
If Layer_Two = Uncertain and Layer_One = Genuine and Layer_Three = Fake then Legitimate
If Layer_One = Genuine and Layer_Two = Uncertain and Layer_Three = Legal then Legitimate
If Layer_One = Genuine and Layer_Three = Uncertain and Layer_Two = Fake then Legitimate
If Layer_Two = Uncertain and Layer_One = Genuine and Layer_Three = Uncertain then Legitimate
If Layer_Three = Legal and Layer_Two = Fake and Layer_One = Doubtful then Suspicious
If Layer_Three = Uncertain and Layer_One = Doubtful and Layer_Two = Fake then Suspicious
If Layer_Three = Legal and Layer_One = Genuine and Layer_Two = Fake then Suspicious
If Layer_One = Fraud and Layer_Two = Legal and Layer_Three = Uncertain then Suspicious
If Layer_Three = Uncertain and Layer_One = Genuine and Layer_Two = Uncertain then Suspicious
If Layer_One = Fraud and Layer_Two = Legal and Layer_Three = Legal then Suspicious
If Layer_Three = Uncertain and Layer_Two = Fake and Layer_One = Genuine then Suspicious
If Layer_One = Fraud and Layer_Two = Legal and Layer_Three = Fake then Suspicious
If Layer_Three = Uncertain and Layer_Two = Fake and Layer_One = Fraud then Suspicious
If Layer_One = Fraud and Layer_Two = Uncertain then Phishing
If Layer_One = Fraud and Layer_Two = Fake and Layer_Three = Fake then Phishing
If Layer_One = Fraud and Layer_Two = Fake and Layer_Three = Legal then Phishing
If Layer_Three = Fake and Layer_Two = Fake then Phishing
If Layer_One = Fraud and Layer_Two = Fake and Layer_Three = Uncertain then Phishing
If Layer_One = Fraud and Layer_Three = Fake and Layer_Two = Legal then Phishing

Number of Rules : 22

J48 pruned tree

Layer_Two = Legal
| Layer_One = Genuine: Legitimate (171.0)
| Layer_One = Doubtful: Legitimate (234.0)
| Layer_One = Fraud
| | Layer_Three = Legal: Suspicious (38.0)
| | Layer_Three = Uncertain: Suspicious (62.0)
| | Layer_Three = Fake: Phishing (42.0/13.0)
Layer_Two = Uncertain
| Layer_One = Genuine: Legitimate (99.0/14.0)
| Layer_One = Doubtful: Legitimate (117.0)
| Layer_One = Fraud: Phishing (83.0)
Layer_Two = Fake
| Layer_Three = Legal
| | Layer_One = Genuine: Suspicious (112.0)
| | Layer_One = Doubtful: Suspicious (147.0)
| | Layer_One = Fraud: Phishing (97.0)
| Layer_Three = Uncertain
| | Layer_One = Genuine: Legitimate (187.0/58.0)
| | Layer_One = Doubtful: Suspicious (246.0)
| | Layer_One = Fraud: Phishing (149.0/21.0)
| Layer_Three = Fake: Phishing (394.0)

Number of Leaves : 15
Size of the tree : 22

We recorded the prediction accuracy and the number of rules generated by the classification algorithms for final phishing website detection rate in Table 5.6, and the chart of decision J48 tree in Figure 5.8, which demonstrate the importance of layer two compared to the other layers.

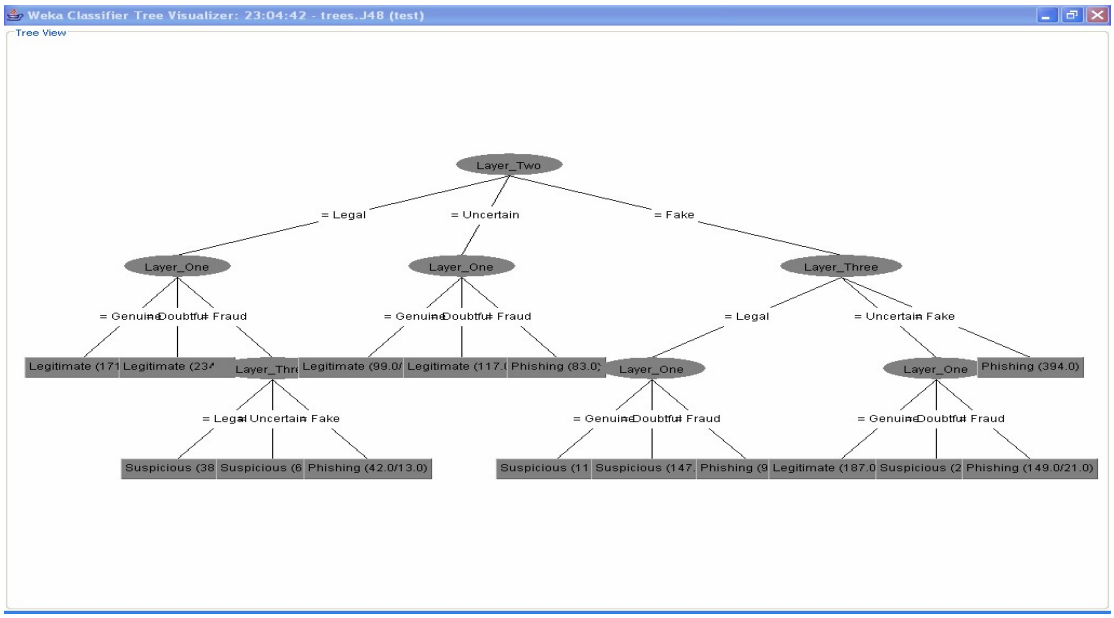


Figure 5.8: Chart of decision J48 tree for final phishing website detection rate

Table 5.6: Classification prediction accuracy and rules number for final phishing website detection rate

	Final Phishing Website Detection Rate				
Algorithms	**JRIP R.I.P.P.E.R**	**PART**	**PRISM**	**C4.5 Decision Tree(J48)**	**CBA**
Test Mode	**10 FOLD CROSS VALIDATION**				
Attributes	**Layer One, Layer Two, Layer Three CLASS**				
Number of Rules	**7**	**13**	**22**	**15 Leaves Tree size 22**	**10**
Correctly Classified	**2046 (93.939%)**				**1972 (90.542 %)**
Incorrectly Classified	**132 (6.061%)**				**206 (9.458%)**
Number of Instances	**2187**				

5.11 Experimental Results and Discussion

The experiments show that web page properties can be used to successfully distinguish between phishing and legitimate websites. As a result, page analysis techniques can, in principle, be effective in building a phishing website intelligent detection model. The experiments also demonstrate the feasibility of using Associative Classification techniques in real applications involving large databases.

Association Classification data miner experiments show lots of important results and conclusions related to classified phishing rules in all of our model layers and criteria. The rules generated from the associative classification model show the correlation and relationships between all phishing features and patterns at every phase. These mined classified rules helped us a lot in producing a more accurate phishing website detection system, as it integrated into the fuzzy logic inference engine.

From all of the above classification rules which cover all our intelligent detection model layers and criteria, using the five mining algorithms, we managed to conclude some very important classification rules. These generated rules helped us greatly in modelling our intelligent phishing website detection. For example, if only one of the phishing fuzzy input variables located in any criteria is "High", then all the criteria will have "Fraud" or "Fake" fuzzy value, whatever the other variables are. This shows of course the big influence of the "High" phishing fuzzy variable and its effect on the entire model. We also concluded that, if there are at least two "Moderate" fuzzy input variables in any criteria without any "High" fuzzy input, then the final result for those criteria will have "Doubtful" or "Suspicious" fuzzy value.

Table 5.7 also shows the most imperative essential phishing features and the most trivial influence phishing features of all criteria and layers found in our association classification detection fuzzy model that reflect the final phishing website detection rate. The table contents greatly reflected our classified phishing rules implemented on the practical implementation of our plug-in phishing toolbar, giving more accurate and precise results in detecting phishing websites, with very little false positive and false negative alarms.

Table 5.7: Influences of different features and criteria in phishing

	Most Significant Influence	Most Trivial Influence
URL & Domain Identity Criteria	**Abnormal Request URL, Abnormal DNS Record**	**Using the IP Address**
Security & Encryption Criteria	**Using SSL Certificate, Certification Authority**	**Abnormal Cookies**
Source Code & Java Script Criteria	**Pharming Attack, Redirect Pages**	**Using onMouseOver to Hide the Link**
Web Address Bar Criteria	**Long URL address, Replacing similar characters for URL**	**Using Hexadecimal Character Codes**
Page Style & Contents Criteria	**Using Forms with "*Submit*" Button, Using Pop-Ups Windows**	**Disabling Right-Click**
Social Human Factor Criteria	**Much Emphasis on Security and Response**	**Public Generic Salutation**
Intelligent Phishing Website Detection Model	**URL & Domain Identity Criteria, Security & Encryption Criteria**	**Social Human Factor Criteria**

Furthermore, to test our approach's ability, our implemented plug-ins phishing website toolbar recognized and detected approximately 92% of the phishing websites selected from our test data subset, avoiding many misclassified websites and false phishing alarms.

For our implementation introduced in Chapter 6, we have imported all the output of WEKA and CBA classification rules and saved the output in a CSV file. From this file, we have created a pool of classification rules to be integrated into our intelligent phishing detection toolbar implementation represented by classification rule table. The

benefit of integrating the rules table in our application is to gain the ability for our application to be dynamic. Thus, to introduce any new phishing classification rules, all we have to do is just adding the classification rules into the rule table, avoiding the need of changing the application each time a new phishing classification rule is introduced. The defuzzification equation was implemented in our intelligent phishing detection toolbar to defuzzify the extracted fuzzy variables, acting just like a fuzzy inference engine.

Chapter 6

Implementation of the Intelligent Fuzzy-Based Classification Phishing Detection Plug-ins Toolbar

6.1 Introduction

For our implementation of the fuzzy based classification mining model for phishing website detection, we have created our own intelligent phishing website detection toolbar as a plug-ins for the Mozilla Firefox browser. Our intelligent toolbar helps the users to identify phishing websites effectively and dynamically. We used a standard version of JavaScript to extract the basic features of the website. To extract other sophisticated website features, like protocols (https), certificates (SSL) and DNS record, the desktop-based Java (J2SE 1.6) was used. For the application user interface we used standard browser based interface language XUL (XML User Interface Language).

We used the standard JavaScript to extract the website feature because we wanted to extend the application to all standard browsers in our future work. It will be easily

adaptable to be integrated to all browsers which support JavaScript as well as its platform independent (Windows, Linux, Mac OS and UNIX) usability.

6.2 Development Solution Outline

The proposed intelligent anti-phishing toolbar has the ability to extract all of our 27 phishing website features and patterns for each browsed website. It cross-check each extracted feature to validate the phishing vulnerability based on specified fuzzy sets to correspond them to related fuzzy variables (High, Moderate and Low).

The toolbar considers and fits each extracted phishing feature in its predetermined criteria and layer, based on risk significance and type. The system has defined six criteria (URL & Domain Identity, Security & Encryption, Source Code & Java Script, Page Style & Contents, Web Address Bar and Social Human Factor) and three layers (Layer One, Layer Two, Layer Three) as suggested by our intelligent phishing website detection model in chapter 5 for the final output. We utilised the classification rules which were generated automatically from the associative classification data miner model to correlate each layer with its preceding layer output.

To define all associate rules for phishing features and patterns in every specific criterion at each particular layer, we adopted some rule pruning techniques based on the significance of the criteria and layer of phishing risk ranking and weight. We used the pruning technique to optimize the processing time for a prompt accurate result. For example, in layer one if we got a high value as a fuzzy input variable for some phishing feature; we ignored checking other features on that layer. Since one of the most important conclusion that results from data miner associative classification algorithms is

that finding only one high fuzzy input feature in any criteria is enough to make the outcome fraudulent or fake for all the criteria. The same rules applied for any two moderate fuzzy input features to make the whole criteria as doubtful or uncertain.

We used fuzzy-based heuristic mining approach and pruning technique that will make the toolbar more effective and efficient to detect phishing websites compared to any other phishing detection technique, because most of existing phishing detection techniques just use a black-listing or white-listing approach. The success of black-listing or white-listing depends on an extensive database, and dealing with a massive database makes the response time much slower and impractical. Another problem is this technique needs frequently-updated data which makes it totally unreliable, and also this technique is not effective on 0 days attacks or spear attacks that are targeted to a specific organisation or group. Our techniques outperform the old existing techniques in terms of the phishing website detection rate, response time, reliability, accuracy and human intervention dependability.

With this toolbar plug-ins, we managed to prove the applicability of using fuzzy based classification mining techniques for phishing website detection. Since website phishing detection is a fuzzy problem, so we argue that our fuzzy rules based heuristic AI approach is more accurate and appropriate for phishing detection.

6.3 Screen Shots and Source Code Examples

- **Screen Shots Examples**

Figure 6.1 and Figure 6.2 shows screenshots of our intelligent plug-ins phishing website detection toolbar for testing the legitimacy of the HSBC official e-banking website (www.hsbc.co.uk). Our intelligent toolbar checked all extracted 27 phishing features

and patterns that can be found on this site. Then using the fuzzy-based classification rule mining approach adopted by our intelligent toolbar, all layered phishing features and patterns were associated and classified with each other for the final detection decision. Since the outputs of the three layers for that website were "genuine" and "legal", the final phishing detection rate was "Legitimate website" with the green colour indicator making it more observable for users. We used the green colour for legitimate websites, red for phishing websites and yellow for suspicious websites.

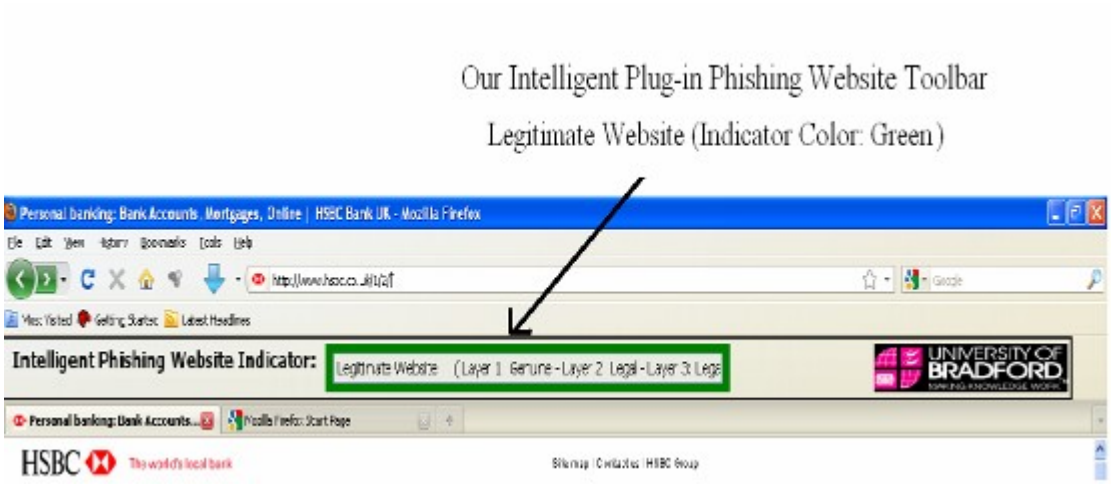


Figure 6.1: Our plug-ins phishing detection toolbar (legitimate website-green colour)

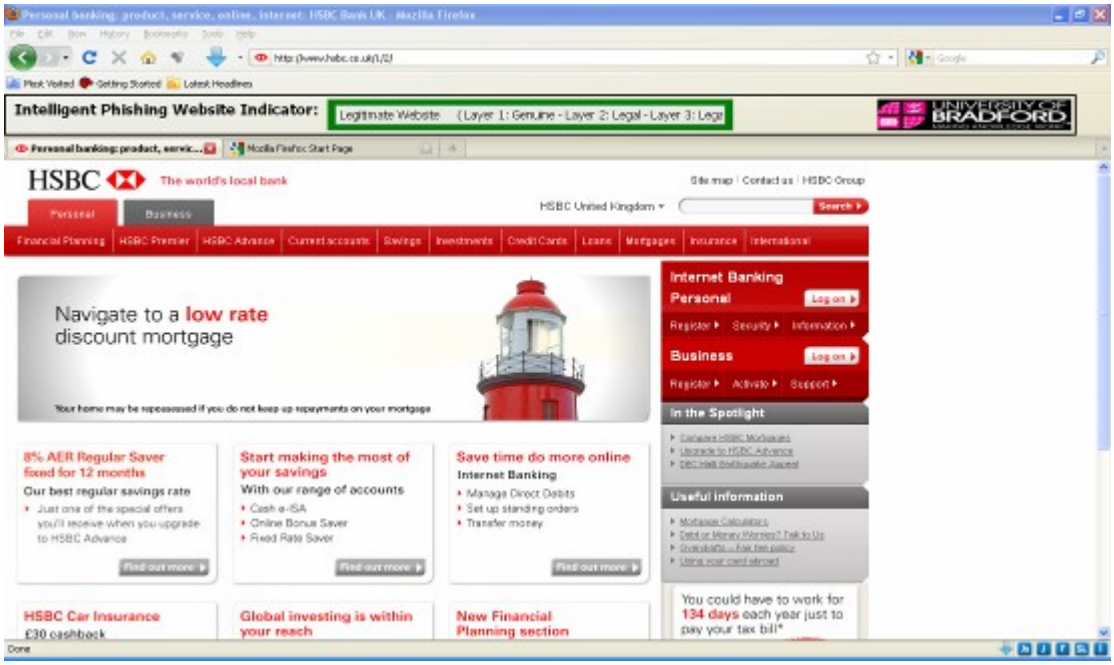


Figure 6.2: Screen shot of legitimate website (hsbc.co.uk) using our plug-ins

Figure 6.3 and Figure 6.4 shows screen shots for using our detection toolbar on a website for Citibank clients (Citybank.net). Since the outputs of the three layers for that website were mixed between "Fraud" for Layer one and "Legal" for Layer two and three, the final phishing detection rate was "Phishing website" with a red colour indicator.

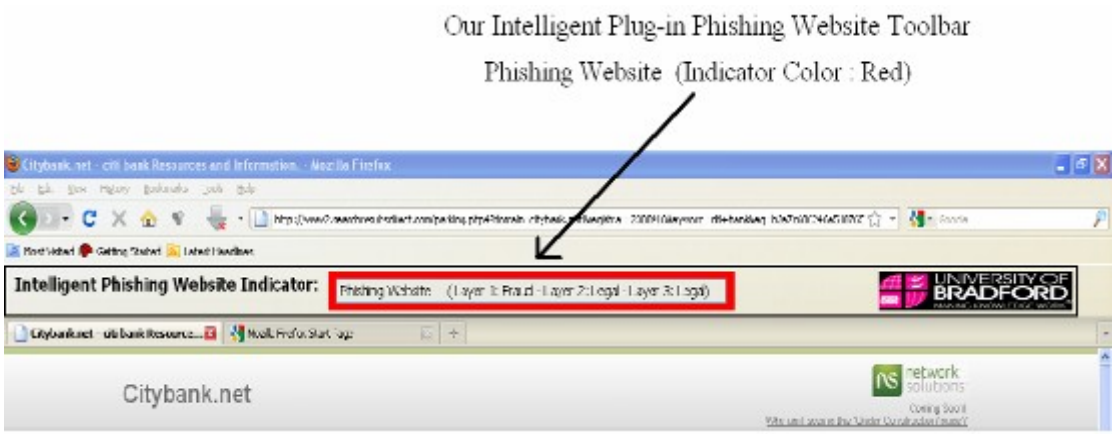


Figure 6.3: Our plug-ins phishing detection toolbar (phishing website-red colour)

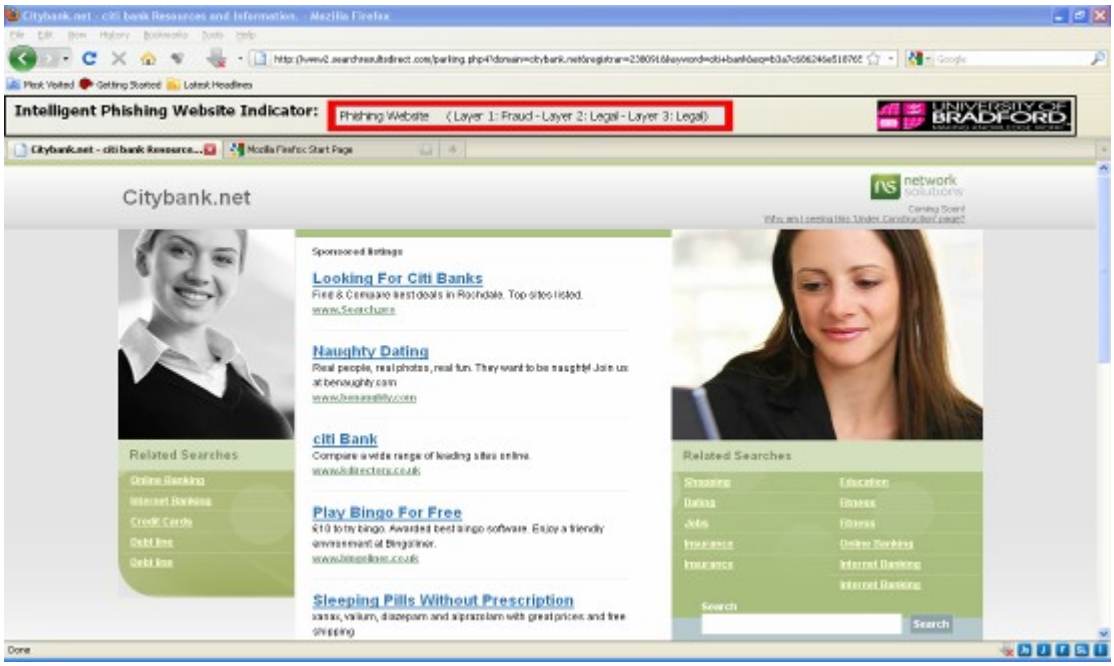


Figure 6.4: Screen shot of phishing website (Citibank.net) using our plug-ins

Figure 6.5 and Figure 6.6 shows screen shots using our detection toolbar on a website for Ahli bank clients (ahly.com). Since the outputs of the three layers for that website were mixed between "Genuine" for layer one and "Uncertain" for layer two and three, the final phishing detection rate was "Suspicious website" with yellow colour indicator.

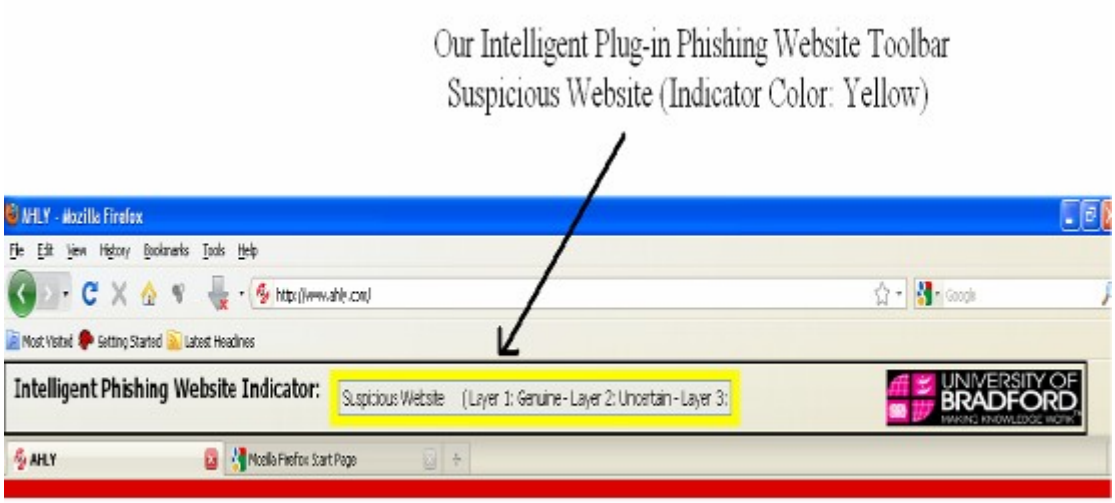


Figure 6.5: Our plug-ins phishing detection toolbar (suspicious website-yellow colour)

Figure 6.6: Screen shot of phishing website (ahly.com) using our plug-ins

Figure 6.7 and Figure 6.8 shows other screen shots of the official legitimate websites for Ahli bank clients (ahli.com) and Citibank clients (Citibank.com) as indicated by our intelligent plug-ins phishing website detection toolbar.

Figure 6.7: Screen shot of legititmate website (ahli.com) using our plug-ins

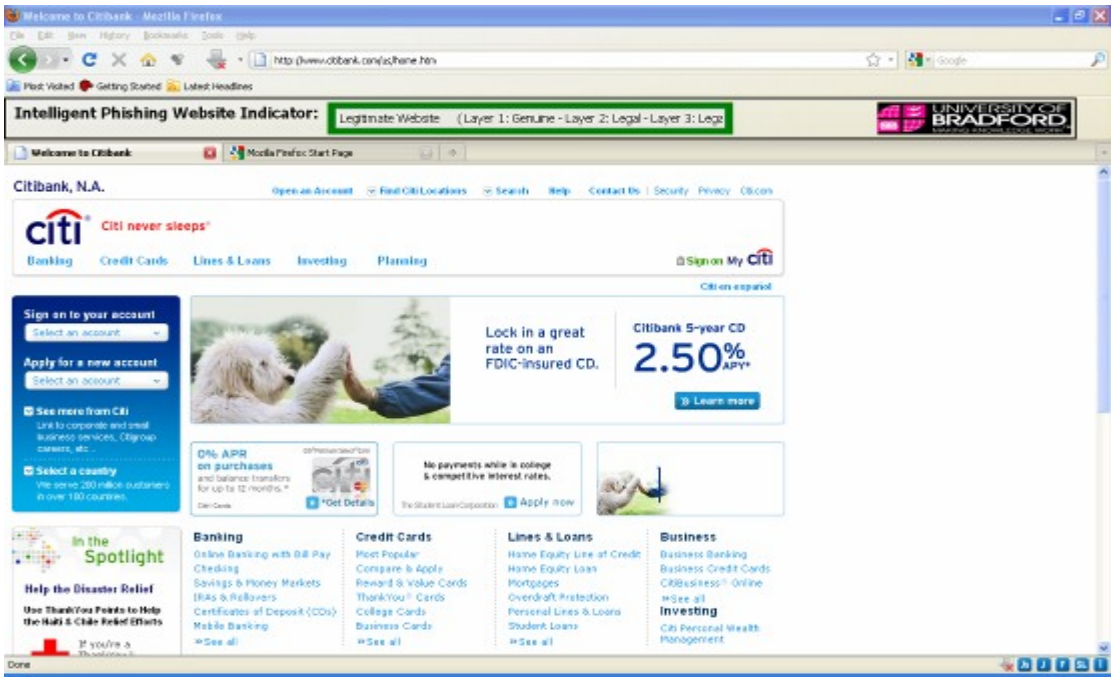


Figure 6.8: Screen shot of legititmate website (citibank.com) using our plug-ins

- **Source Code and Pseudo Code Examples**

We show now some important source code and pseudo code examples for extracting some of the phishing website features for our system implementation. This section also demonstrates how to validate the phishing features with our proposed phishing criteria and rate the fuzzy variable inputs accordingly.

- **Pop-Up Window Extracting Phishing Feature Source Code Example**

```
var popUpWindow = "Low";
        popUpCount = 0;
        var elems = window._content.document.getElementsByTagName("script");
        if(elems){
                for(i=0;i< elems.length; i++){
                        if(elems[i].innerHTML){
                                var code = elems[i].innerHTML;
                                if(code.indexOf("window.open") > -1){
                                        popUpCount++; }
                        }
                }
        }
        var toCheck = window._content.document.body.innerHTML;

        var findLock = toCheck.indexOf("window.open");

        while(findLock > -1){
                findLock = toCheck.indexOf("window.open", findLock+1);
                popUpCount++;
        }
```

```
                if(popUpCount >= 0 && popUpCount < 2){
                popUpWindow = "Low";
        }else if(popUpCount >= 2 && popUpCount <8){
                popUpWindow = "Moderate";
        }else{
                popUpWindow = "High";
        }
                if(popUpWindow == "High")
                return "Fraud";
        else if(popUpWindow == "Moderate" && formSubmit == "Moderate"){
                return "Doubtful";
        }
```

Here we count how many times the pop-up window exists on the website. If it does not exist at all in the website, or there is at most just one pop-up window, then we give the fuzzy input variable "Low" value. If there is from 2 to 7 pop-up windows then we give the fuzzy input variable "Moderate" value. Otherwise, we give it "High" fuzzy value.

- **Redirect Page Extracted Phishing Feature Source Code Example**

```
 var elems = window._content.document.getElementsByTagName("script");
        redirectCount = 0;
        usingRedirect = "Low";
        if(elems){
                for(i=0;i< elems.length; i++){
                        if(elems[i].innerHTML){
                          var code = elems[i].innerHTML;
                          var findLoc1 = code.indexOf("window.location=\"");
                          if(findLoc1 > -1){
                                var findLoc2 = code.indexOf("\"",findLoc1+1);
                                var toCheck = code.substring(findLoc1,findLoc2);
                                url =
window.top.getBrowser().selectedBrowser.contentWindow.location.href;
                                domain = url.split(/\/+/g)[1].replace('www.','');
                                pattern = "/"+domain+"/gi";
                                pattern = eval(pattern);
                                if(toCheck.match(pattern) == null)
                                        redirectCount++;  }
                                        }
        }
        var elems = window._content.document.getElementsByTagName("meta");
        if(elems){
                for(i=0; i< elems.length; i++){
                        toCheck = elems[i].content;
                        if(toCheck.indexOf("url=") > -1){
                                return redirectCount;
                                url =
window.top.getBrowser().selectedBrowser.contentWindow.location.href;
                                domain = url.split(/\/+/g)[1].replace('www.','');
                                pattern = "/"+domain+"/gi";
                                pattern = eval(pattern);
                                if(toCheck.match(pattern) == null)
                                        redirectCount++; }
                }
        }
        if(redirectCount < 2)
                usingRedirect = "Low";
```

```
else if(redirectCount >= 2 && redirectCount <= 4)
        usingRedirect = "Moderate";
else
        usingRedirect = "High";
```

There are two ways to "Redirect Pages" from one site to another. The first is a script used to redirect with a syntax "window.location" and the other one is on the page where the <meta> refresh tag is used with a URL specified to the final targeted page. In this section of the code we considered both the possibility and count number of occurrences of these two techniques on a browsed page. To rate the Redirect Page feature as "High", the fuzzy input variable we considered had more than 4 occurrences; to rate it as "Moderate" we are considering between 2-4 occurrence; and finally less than 2 occurrence were rated as "Low".

- **Abnormal URL Anchor Extracting Phishing Feature Pseudo Code Example**

```
elems :- extract all window elements by the tag name a (Anchor);
url :- get the browsing URL address from the Location bar;
Domain :- get the Domain name part from the Whole URL without the "www" part;
Pattern :- make the pattern match using the extracted domain name;
notMatchedCountAnchor :- set the counter to 0;
abnormalURLRequestAnchor :- set the Fuzzy Variable to "Low" initially
Check if there is any anchor element
        Do for every element
                Check if the link URL does not match with the pattern
                        notMatchedCountAnchor :- increment the counter;
        End
Calculate the percentage of mismatched found using the notMatchedCountAnchor counter and
the total number of Anchors in the page
notMatchedCountAnchorRatio :- (notMatchedCountAnchor/ total number of Anchor)*10
Check if notMatchedCountAnchorRatio is less then or equal to 20
        abnormalURLRequestAnchor :- "Low";
Otherwise check if notMatchedCountAnchorRatio is in the range between 21 and 50
        abnormalURLRequestAnchor :- " Moderate ";
Otherwise
        abnormalURLRequestAnchor :- " High ";
```

To validate the "Abnormal URL Anchor" feature we extracted all the anchor elements of the page. Then we counted the number of anchors that were pointed at some other website other than the browsed domain name, and we calculated the percentage of URLs that were pointed to some other website. If the percentage was less than 20% we

rated the fuzzy variable as "Low"; we rated it "Moderate" if the percentage was between 21 -50; otherwise we rated it as "High".

6.4 Implementation Constraints

We faced some implementation constraints regarding extracting and validating some of the 27 phishing website features. For example, validating the extracted spelling errors phishing feature was not 100% accurate since it included nouns which were not listed as dictionary words and would be considered spelling errors. This is likely to give a 25% error on spelling error detection.

As another example, we did not include WHOIS database query result with the validation process of phishing website features; because of the difficulties in extracting the data from WHOIS query result. That is the reason we could not validate the "Abnormal DNS Record" and "Abnormal Request URL" 100% accurately. The validation of these two features did not give the expected output for www.facebook.com, www.yahoo.com or any other website that uses a different valid and registered domain for image, script and other recourses. Nevertheless we are not facing this problem for e-Banking or e-Commerce sites, since they are very consistent in using their single domain to store every resource for security purposes.

Finally, we faced some constraints regarding 100% validation of Copying Website phishing feature. Some phishing websites copy the whole contents from some legitimate websites, and put it on their own domain. This malicious technique disguises the track of the origin of the resources, such that it appears to be owned by the phishing site, but actually it is not. This malicious technique leaves hardly 40% non matches between the

legitimate and the phishing sites. So we have to rely only on this percentage of accuracy for that feature.

6.5 Testing and Validation

While there is no mature technology that defends against phishing web sites yet, there is currently no anti-phishing benchmark set of expectation or standardized set of data for phishing detection products evaluation. Most of the claims made by vendors of available products are based on proprietary test data and testing methodology. In this research, a test framework has been constructed which can evaluate a generic anti-phishing technology against the latest existing phishing sites. This framework has been used to evaluate the effectiveness of our intelligent plug-ins phishing detection toolbar. We have selected the PhishTank data as the public benchmark for our comparing phishing detection. Details of this experimentation framework and findings are presented below.

Using testing sample of 120 different e-banking website that was used previously on our fuzzy logic phishing website detection model, we tested our intelligent web-based plug-ins toolbar to prove its validation and high phishing detection precision. The dataset sample was taken from the public benchmark Phishtank archive data (Phishtank, 2008), consisting of 60 phishing websites: 35 suspicious websites and 25 legitimate websites. Our toolbar managed to detect the phishing e-banking websites that were found in the testing sample with a very small miss-classification rate. The results indicate clearly the high precision of phishing classification with very small false positive and false negative rates, as specified in the confusion matrix shown in Table 6.1.

Table 6.1: Results of website legitimacy decision using the intelligent fuzzy-based classification detection model

Decision Website Legitimacy	Legitimate	Suspicious	Phishy
Legitimate Website	22	2	1
Suspicious Website	1	32	2
Phishing Website	1	3	56

As shown in Table 6.1, there were just 3 legitimate websites miss-classified as suspicious or phishy websites, and only 4 phishing websites were miss-classified as legitimate or suspicious website.

These results demonstrate very clearly how effective and reliable detecting phishing website can be when applying an intelligent heuristic search using association classification mining algorithms combined with a fuzzy logic model approach. The obvious enhancement that happened to the final results can be justified by using an approach not only depending on the human expert knowledge alone, but also on integrating and combining an intelligent supervised machine learning approach, using specific mining associative classification algorithms. When comparing our intelligent web browser plug-ins toolbar with other famous anti-phishing toolbars like Netcraft (Netcraft, 2006) and Spoofstick (Spoofstick, 2005) toolbars, we found that our toolbar outperformed the other detection toolbars regarding the accuracy, efficiency and the speed of classifying and detecting phishing websites. It managed to classify correctly approximately 92% of all tested websites, beating all other anti-phishing toolbars, which depend mainly on using black-list and white-list databases in classifying phishing websites. Figure 6.9 shows the comparative performance of all tested anti-phishing toolbars for the accuracy phishing classification rate.

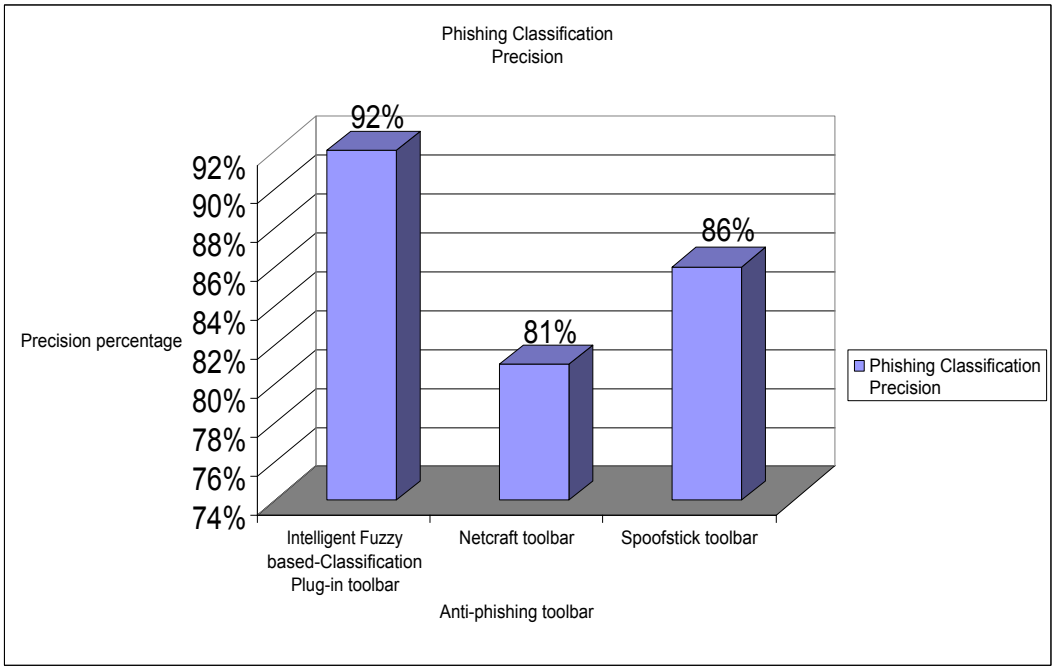


Figure 6.9: Phishing classification precision comparing chart

It is noted that the proposed tool offered best performance among the tested tools, being about 11% better compared to Netcraft and 6% better compared to Spoofstick. We argue that our solution is better since it uses a novel AI heuristic search on all phishing features that can be found on the websites, grouping them into specific criteria and layers depending on their type, and then by using specific fuzzy-based classification rules, the final phishing detection rate appears.

Chapter 7

Conclusions and Future Work

7.1 Conclusions

An AI-based hybrid system has been proposed for phishing website detection systems. Fuzzy logic has been combined with association classification data mining algorithms to provide efficient techniques for building intelligent models to detect phishing websites. Empirical phishing experimental case studies have been implemented to gather and analyze range of different phishing website features and patterns, with all its relations.

Our experimental case-studies point to the need for extensive educational campaigns about phishing and other security threats. People can become less vulnerable with a heightened awareness of the dangers of phishing. Our experimental case-studies also suggest that a new approach is needed to design a usable model for detecting e-banking phishing websites, taking into consideration the user's knowledge, understanding, awareness and consideration of the phishing pointers located outside the user’s centre of interest.

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. Fuzzy rules captured from previous human expert knowledge, processed by the fuzzy set operations into the inference engine for the final calculation of the phishing website detection rate. Results shows the significance and importance of the phishing website criteria (URL & Domain Identity) represented by layer one, especially when compared to the other criteria and layers.

Enchantment has been proposed by utilising supervised machine learning techniques to automate the fuzzy rule generation process, in order to reduce the human expert knowledge intervention and increase performance of the phishing detection system.

In this investigation, we have generated classification rules and investigated the predictive accuracy of five classifiers on a phishing data set. The classifiers included JRip (RIPPER), PART, PRISM, C4.5 Decision Tree (J48) and Classification Based on Association (CBA). By analyzing a large number of phishing pages, we built an associative classification model that attempts to use the properties of a page (e.g., URL address length, SSL certificate, Abnormal URL request, Certification Authority, etc.) to distinguish between phishing and legitimate website pages. We constructed a data set from 731 phishing websites, 711 suspicious websites and 1718 legitimate websites, where 27 phishing features were trained and tested to detect phishing websites. During training and testing we used 10-fold cross-validation to evaluate the error rate for all classifiers. Mining association classification rules were then combined with the fuzzy logic inference engine to provide efficient and competent techniques for phishing website detection rate.

We showed 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 which use an Artificial Intelligence heuristics page search. Using this approach will automate the fuzzy rule generation process and reduce the human intervention in building an effective phishing detection intelligent model.

A browser-based plug-ins phishing detection toolbar has been implemented using an intelligent heuristic approach. The toolbar has extracted all the phishing website features and patterns. Validation of the extracted features has been integrated into the solution to effectively identify phishing, legitimate and suspicious website. An intelligent pruning technique has been used to increase the performance of the phishing detection rate.

The intelligent phishing detection toolbar reduces the requirement of human knowledge intervention for detection of a phishing website. Our toolbar 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 shows that the proposed solution outperformed the existing detection toolbars regarding the accuracy, efficiency and the speed of classifying and detecting phishing websites. It managed to classify correctly approximately 92% of all tested websites.

The experimental results showed that both its false-positive rate and miss rate are reasonably low. A comparative performance of the proposed scheme was presented in order to demonstrate the merits of capabilities through a set of experiments. It is noted

that the proposed intelligent system offers better performance as compared to other existing tools and techniques.

Many contributions evolved from our investigation research which can be very useful for all researchers interested in the field of internet security and online identity theft protection using artificial intelligence (AI). Following are summary of the main contributions:

- Two phishing experiments which covered website phishing attack techniques and phishing detection survey scenario were conducted to cover all phishing approaches, motivations and deception behaviour techniques.
- 27 phishing features and patterns which characterize any phishing website were successfully extracted, divided into 6 criteria or categories distributed in three layers, depending on its attack type.
- A dynamic intelligent phishing website detection system has been proposed based on specific AI supervised machine learning approach. The technique utilises fuzzy logic combined with simple data mining associative classification techniques and algorithms to process the phishing data features and patterns, for extracting classification rules into the data miner. The proposed phishing website system combines these techniques together to automate the fuzzy rules production by using the extracted classification rules to be implemented inside the fuzzy inference engine for the final phishing website detection.
- A web-based plug-ins intelligent phishing website detection toolbar has been designed for testing and validation, using our integrated fuzzy based classification mining model to prove its feasibility, reliability and detection precision. The implementation was programmed using Java language, and it successfully

recognized and detected approximately 92% of the phishing websites selected from our test data subset, avoiding many miss-classified websites and false phishing alarms.

7.2 Future Work

A fuzzy-based classification mining technique has been introduced for building an intelligent phishing website detection system, by using a layered structure for collecting and analyzing all phishing website features and patterns. This kind of supervised machine learning technique which combined the fuzzy logic model with the associated classification technique for detecting phishing websites verified lots of potential for its validity and usability throughout our research investigation.

The results motivate future work to explore the inclusion of any additional variables to the data set, which might improve the predictive accuracy of classifiers and decrease the misclassification rate of rule classification. In addition, we will explore developing an automated mechanism to extract new potential of phishing risk features from raw phishing websites in order to keep up with new trends in phishing attacks.

As we stated before in implementation Chapter 6, we considered some of the implementation constraints we faced, regarding extracting and validating some phishing website features as our future motivation to overcome and resolve. This is important as it is a major barrier for our intelligent solution to get maximum possible performance and accuracy.

For example, the validation of the extracted spelling errors phishing features. It includes nouns that are not listed as dictionary words and will be considered as spelling errors.

We can use as a future work keyword extraction algorithm to solve the spelling error problem.

Also, some constraints regarding the validation of "Abnormal DNS Record" and "Abnormal Request URL". These two features of validation were not giving the expected output for any website that uses valid and registered but different domains for image, script and other recourses. We can use intelligent domain identifying database query to overcome this problem, because this database gives information regarding all valid and registered websites in detail.

Another example: we faced some constraints regarding 100% validation of Copying Website phishing features. Some phishing websites copy whole contents from some legitimate websites, putting all contents and resources in their own domain, leaving about 40% non matches between sites. So, for this feature we can rely only on this accuracy percentage. We can use the normalized form of the website contents (text, image, style and JavaScript) to search for similarities with other websites.

As future work also, we want to extend our work by integrating our phishing website detection toolbar to all other standard browsers (examples: exe file for internet explorer and also plug-ins such as Google chrome). Then our ultimate goal is to make the phishing detection toolbar a desktop application, so that it can run as a background process to be used as an independent phishing detection tool.

Further, to exploit this application as security awareness regarding phishing attacks and scams, we will extend it to be used as a learning tool to increase user awareness regarding phishing attacks and scams. We plan to demonstrate our decision justification by breaking down our validation of extracted phishing features and their significance

influence as summarized report. We also want to integrate phishing detection assessment user interface (example: short questionnaires, tests cases) to measure the effectiveness of our e-learning tools. To make the learning mechanism more effective and interactive we are considering integrating the concept of phishing games into the e-learning process. This ensures our package will be dynamic and user friendly.

Finally, we believe that our model can integrate other supervised machine learning techniques like Neural Network (NN). We can use our 27 phishing features as inputs to Neural Network for the first input layer, and we can use the outputs of the first layer as the input to the second hidden layer of our Neural Network. Same logic will be applied for the third hidden layer of the Neural Network. The output of the final layer will give us the phishing detection rate as legitimate, suspicious or phishy.

We can use the Neural Network for our solution because its working procedure is similar to our layered fuzzy structure. We need to implement a phishing feature extractor engine, which will generate the inputs for our Neural Network to give the final phishing website detection rating.

References

Abu-Nimeh, S., Nappa, D., Wang, X., and Nair, S. (2007) A comparison of machine learning techniques for phishing detection. *In eCrime '07: Proceedings of the anti-phishing working groups 2nd annual eCrime researchers summit*, (pp. 60–69), New York, NY, USA, ACM.

Adida, B., Hohenberger, S., Rivest, R. (2005) Lightweight Encryption for Email. USENIX Steps to Reducing Unwanted Traffic on the Internet Workshop (SRUTI), (pp. 93–99).

Ahmad, A. and Basir, O. (2000) Fuzzy Inferencing in the Webpage Layout Design. Working Paper. System Design Engineering University of Waterloo, (pp. 33-41), Waterloo, Canada.

Ali, K., Manganaris, S., and Srikant, R. (1997) Partial classification using association rules. In Heckerman, D., Mannila, H., Pregibon, D., and Uthurusamy, R., (eds.). *Proceedings of the Third International Conference on Knowledge Discovery and Data Mining*, (pp. 115-118). Newport Beach, CA.

Alnajim, A., and Munro, M. (2008) An Evaluation of Users' Tips Effectiveness for Phishing Websites Detection, (pp. 63-68), 978-1-4244-2917-2/08, IEEE.

APWG, (2005) Phishing Activity Trends Report, http://antiphishing.org/reports/apwg_report_DEC2005_FINAL.pdf. Access date [4/12/2005].

APWG, (2008) Phishing Activity Trends Report, http://antiphishing.org/reports/apwg_report_sep2008_final.pdf. Access date [3/09/2008].

APWG,(http://www.apwg.org/reports/APWG_GlobalPhishingSurvey_1H2009.pdf), Access date [4/8/2009].

Binxing, D., and Ruifeng, L. (2006) Phishing and Anti-phishing, Master of Science Thesis, Department of Computer and Systems Sciences, Stockholm's University.

Bridges, S., and Vaughn, R. (2001) Fuzzy Data Mining And Genetic Algorithms Applied To Intrusion Detection, *Proceedings of the 23rd National Information Systems Security Conference,* (pp. 13-31).

Broder, S., Glassman, M., Manasse, and Zweig, G. (1997) Syntactic Clustering of the Web, *Proceedings of. Sixth Int'l World Wide Web Conference.*, (pp. 391-404).

Brooks, J. (2006) Anti-Phishing Best Practices: Keys to aggressively and effectively protecting your organisation from Phishing Attacks, White Paper, Cyveillance.

Buckley, J., and Tucker, D. (1989) Second generation fuzzy expert system. Fuzzy Sets and Systems, Vol.31, No.4, (pp. 271-284).

Business Security Guidance, (2006) How to Protect Insiders from Social Engineering Threats, www.microsoft.com/technet/security/default.mspx, Access date [4/8/2006].

Callow, B. (2009) How To Protect Against Social Engineering Attacks, article, http://www.brighthub.com/computing/smb-security/articles/1313.aspx, Access date [5/7/2009]

Cendrowska, J. (1987) PRISM: An algorithm for inducing modular rules. International Journal of Man-Machine Studies. Vol. 27, No. 4, (pp.349-370).

Chandrasekaran, M. (2005) New Way to Avoid Phishing, Journal, Computer Software and Applications Conference, IEEE, Vol. 9, No. 7, (pp. 161-165).

Chandrasekaran, M., Narayanan, K., and Upadhyaya, S. (2006) Phishing email detection based on structural properties. *Proceedings of the NYS Cyber Security Conference*.

Chen, Y., Ma, W., and Zhang, H. (2003) Detecting Web Page Structure for Adaptive Viewing on Small Form Factor Devices, *Proc. 12th Int'l Conf. World Wide Web*, (pp. 225-233).

Chhabra, S. (2005) Fighting Spam, Phishing and Email Fraud, Master of Science Thesis.

Chinchani, R., and Upadhyaya, S. (2005) Analysis of Phishing, Journal, World of Wireless, Vol. 7, No. 11, (pp. 70-73).

Chinmay, S., Hrishikesh, P., Vishal, S., Aniket, P., and Amey, I. (2008) An Intelligent System for Phish Detection, using Dynamic Analysis and Template Matching, World Academy of Science, Engineering and Technology.

Chou, N., Ledesma, R., Teraguchi, Y., Boneh, D., and Mitchell, J. (2004) Client side defense against web-based identity theft. *In Proceeding of the 11th Annual Network and Distributed System Security Symposium (NDSS '04)*.

Chowdhury, O., Frieder, D., Grossman, and McCabe, M. (2002) Collection Statistics for Fast Duplicate Document Detection, ACM Trans. Information Systems, Vol. 20, No. 2, (pp. 171-191).

Ciesielski, V., and Lalani, A. (2003) Data Mining of Web Access Logs From an Academic Web Site, *Proceedings of the Third International Conference on Hybrid Intelligent Systems (HIS'03)*: Design and Application of Hybrid Intelligent Systems, (pp. 1034-1043).

Cohen, W. (1995) Fast effective rule induction. *Proceedings of the 12th International Conference on Machine Learning*, (pp. 115-123). CA, USA.

Cox, E. (2001) FL and Measures of Certainty in E-Commerce Expert System. Article. Scianta Intelligence, Chapel Hill.

Cranor, L., Egelman, S., Hong, J., and Zhang, Y. (2008) Phinding phish: An evaluation of antiphishing toolbars. In *Network & Distributed System Security (NDSS) Symposium,* CMU-CyLab-06-018.

Cryptomathic, (2004) Internet Banking & Two-Factor Authentication, White Paper, www.cryptomathic.com.

Dhamija, R., and Tygar, J. (2005) The battle against phishing: Dynamic security skins. *In Proc. ACM Symposium on Usable Security and Privacy (SOUPS 2005)*, (pp. 77–88).

Dhamija, R., Tygar, J., and Marti, H. (2006) Why phishing works, *In CHI '06: Proceedings of the SIGCHI conference on Human Factors in computing systems*, ACM Press, (pp. 581-590), New York, NY, USA.

Dickerson, J. and Dickerson, J. (2000) Fuzzy network profiling for intrusion detection, *Proceedings of NAFIPS 19th International Conference of the North American Fuzzy Information Processing Society*, (pp. 301-306), Atlanta, USA.

Ding, B., and Li, R. (2006) Phishing and Anti-phishing, Master of Science Thesis, A comprehensive research on phishing techniques and anti-phishing solutions.

Dong, X., Clark, J., and Jacob, J. (2008) User Behaviour Based Phishing Websites Detection, *Proceedings of the International Multiconference on Computer Science and Information Technology*, (pp. 783–790), ISBN 978-83-60810-14-9.

Duda, R., and Hart, P. (1973) Pattern classification and scene analysis. John Wiley & son.

Emigh, A. (2006) Online Identity Theft: Phishing Technology, Chokepoints and Countermeasures. http://www.antiphishing.org/Phishing-dhs-report.pdf, Access date [13/8/2006]

Fayyad, U., and Irani, K. (1993) Multi—interval discretisation of continues-valued attributes for classification learning. *Proceedings of IJCAI*, (pp. 1022-1027). Chambéry, France.

Fayyad, U., Piatetsky-Shapiro, G., Smith, G., and Uthurusamy, R. (1998) Advances in knowledge discovery and data mining. AAAI Press.

FDIC, (2004) Putting an end to account-hijacking identity theft, FDIC, Tech. Rep., [Online]. Available: http://www.fdic.gov/consumers/consumer/idtheftstudy/identity theft.pdf, Access date [4/18/2007]

Fette, I., Sadeh, N., and Tomasic, A. (2006) Learning to Detect Phishing Emails, Institute for Software Research International, CMU-ISRI-06-112.

Fette, I., Sadeh, N., and Tomasic, A. (2007) Learning to detect phishing emails. In WWW '07: Proceedings of the 16th international conference on World Wide Web, (pp. 649–656), New York, NY, USA, ACM Press.

FFIEC, (2003) E-Banking Introduction, Federal Financial Institutions Examination Council, Information Technology Examination Handbook (IT Handbook InfoBase), Available Online: http://www.ffiec.gov/ffiecinfobase/booklets/e_banking/ebanking_00_intro_def.html , Access date [15/6/2007]

Frank, E., and, Witten, I. (1998) Generating accurate rule sets without global optimisation. *Proceedings of the Fifteenth International Conference on Machine Learning*, (pp. 144–151). Madison, Wisconsin.

Fu, A., Wenyin, L., and Deng, X. (2006) Detecting Phishing Web Pages with Visual Similarity Assessment Based on Earth Mover's Distance (EMD), IEEE transactions on dependable and secure computing, Vol. 3, No. 4, (pp. 301-311).

Furnkranz, J. (1999) Separate-and-conquer rule learning. Artificial Intelligence Review, Vol. 13, No.1, (pp. 3-54).

Gabber, E., Gibbons, P., Kristol, D., Matias, Y., and Mayer, A. (1999) Consistent, yet anonymous, web access with LPWA. Communications of ACM, Vol. 42, No. 2, (pp. 42–47).

Garera, S., Provos, N., Chew, M. and Rubin, A. (2006) A Framework for Detection and Measurement of Phishing Attacks. Technical report, Johns Hopkins University, http://www.cs.jhu.edu/~sdoshi/index_files/phish_measurement.pdf, Access date [22/12/2006].

Gartner, (2007) (http://www.gartner.com/it/page.jsp?id=565125), Access date [9/10/2007].

Gartner, (2008) (http://www.gartner.com/it/page.jsp?id=936913), Access date [2/11/2008].

Gefen, D. (2002) Reflections on the Dimensions of Trust and Trustworthiness Among Online Consumers. ACM SIGMIS Database, Vol. 33, No. 3, (pp.38-53).

Gomez, J., and Dasgupta, D. (2002) Evolving Fuzzy classifiers for Intrusion Detection. *Proceedings of 2002 IEEE Workshop in Information Assurance*, Vol. 6, (pp. 321-323), USA NY.

Gu, X., Chen, J., Ma, W., and Chen, G. (2002) Visual Based Content Understanding towards Web Adaptation, *Proc. Second Int'l Conf. Adaptive Hyper media and Adaptive Web-Based Systems*, (pp. 164-173).

Gundel, T. (2005) Phishing and Internet Banking Security, Technical Security report, IBM Crypto Competence Center.

Hernandez, I, and Leggio, J. (2006) Combating Phishing Websites, Project Proposal.

Herzberg, A. (2005) Trustbar: "Re-establishing trust in the web", http://www.cs.biu.ac.il/˜herzbea/TrustBar/index.html, Access date [5/11/2006].

Herzberg, A., and Gbara, A. (2004) Protecting Naive Web Users, Draft of July 18, 2004.

Hoad, T., and Zobel, J. (2003) Methods for Identifying Versioned and Plagiarized Documents, J. Am. Soc. Information Science and Technology, Vol. 54, No. 3, (pp. 203-215).

Idris, N., and Shanmugam, B. (2006) Novel Attack Detection Using Fuzzy Logic and Data Mining, Security and Management Journal, (pp. 26-31).

InternetSoft, Website eXtractor (2008), http://www.esalesbiz.com/extra/download.htm, Access date [14/6/2008].

Jagatic, T., Johnson, N., Jakobsson, M., and Menczer, F. (2007) Social Phishing, Community. ACM, Vol. 50, No. 10 (pp. 94-100).

Jagatic, T., Johnson, N., Jakobsson, M., and Menczer, F. (2005) Social Phishing, School of Informatics Indiana University, Bloomington, ACM.

Jain, V. and Krishnapuram, R. (2001) Applications of Fuzzy Sets in Personalization for E-Commerce, *Proceeding of 20th Intl. Conference of North America Fuzzy Information Processing Society (NAFIS)*, (pp. 263-268).

Jakobsson, M. (2005) Modeling and Preventing Phishing Attacks, School of Informatics Indiana University at Bloomington.

Jakobsson, M., and Young, A. (2005) Distributed Phishing Attacks, Cryptology ePrint Archive, http://eprint.iacr.org/2005/091.pdf. In: *Proceedings of DIMACS Workshop on Theft in E-Commerce: Content, Identity, and Service*.

Jakobsson, M., Tsow, A., Shah, A., Blevis, E., and Lim, Y. (2007) What Instills Trust? A Qualitative Study of Phishing, (pp. 356-361), Indiana University, Bloomington.

Jaleshgari, R. (1999) Document trading online, Information Week, 755, 136.

James, L. (2006) Phishing Exposed, Tech Target Article sponsored by: Sunbelt software, searchexchange.com.

Kirda, E., and Kruegel, C. (2005a) Filching Attack of on-line Status, Journal, Network Security Technology and Application, Vol. 6, No. 4, (pp. 17-20).

Kirda, E., and Kruegel, C. (2005b) Protecting users against phishing attacks with antiphishing. *In Proceedings of the 29th Annual International Computer Software and Applications Conference (COMPSAC)*, (pp. 517–524).

Kosko, B. (1992) Neural Networks and Fuzzy Systems, Prentice-Hall, Upper Saddle River, N.J.

Lan Yu, Davy Janssens, Guoqing Chen and Geert Wets, Improving Associative Classification by Incorporating Novel Interestingness Measures, *IEEE International Conference on e-Business Engineering (ICEBE'05)*, Beijing, China, Vol. 31, No. 1, (pp. 184-192), ISBN: 0-7695-2430-3.

Li, W., Han, J., and Pei, J. (2001) CMAR: Accurate and efficient classification based on multiple-class association rule. *Proceedings of the ICDM'01* (pp. 369-376). San Jose, CA.

Liao, Z. and Cheung, M. (2002) Internet-based e-banking and consumer attitudes: an empirical study, Elsevier Science Information & Management, Vol. 39, No. 4, (pp. 283–295).

Liu, B., Hsu, W., and Ma, Y. (1998) Integrating classification and association rule mining. Proceedings of the KDD, (pp. 80-86). New York, NY.

Liu, W., Deng, X., Huang, G., and Fu, A. (2006) An Antiphishing Strategy Based on Visual Similarity Assessment, Published by the IEEE Computer Society, Vol. 10 , No. 2, (pp. 58-65), 1089-7801/06 IEEE, Internet Computing.

Liu, W., Guanglin, H., Liu, X., Xiaotie, D. and Zhang, M. (2005) Phishing Webpage Detection, *Proceedings of the 2005 Eight International Conference on Document Analysis and Recognition (ICDAR'05)*, (pp. 560-564), IEEE.

Ludl, C., McAllister, S., Kirda, E., and Kruegel, C. (2007) On the Effectiveness of Techniques to Detect Phishing Sites, in Conference on Detection of Intrusions and Malware and Vulnerability Assessment, LNCS 4579, (pp. 20–39), ISBN: 978-3-540-73613-4, Springer-Verlag Berlin Heidelberg.

MAAWG and APWG, (2006) Anti-Phishing Best Practices for ISPs and Mailbox Providers, A document jointly produced by the MAAWG and APWG, Version 1.01.

Mahant, N. (2004) Risk Assessment is Fuzzy Business – Fuzzy Logic provides the Way to Assess Off-site Risk from Industrial Installations, Bechtel, Australia.

Meretakis, D., and Wüthrich, B. (1999) Extending naïve Bayes classifiers using long itemsets. *Proceedings of the fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, (pp. 165 – 174). San Diego, California.

Merz, C., and Murphy, P. (1996) UCI repository of machine learning databases. Irvine, CA, University of California, Department of Information and Computer Science.

Microsoft Corporation, (2008) Microsoft Phishing Filter: A New Approach to Building Trust in E-Commerce Content, White Paper.

Ming, Q., and Chaobo, Y. (2006) Research and Design of Phishing Alarm System at Client Terminal, *Proceedings of the 2006 IEEE Asia-Pacific Conference on Services Computing (APSCC'06)*, (pp. 597-600), IEEE.

Misch, S. (2006) Content Negotiatio in Internet Mail, Diploma Thesis, University of Applied Sciences Cologne, Mat.No.: 7042524.

Nanno, T., Saito, S., and Okumura, M. (2003) Structuring Web Pages Based on Repetition of Elements, *Proc. Seventh Int'l Conf. Document Analysis and Recognition*.

Negnevitsky, M. (2002) Artificial Intelligence: A Guide to Intelligent System, Pearson Education, Sydney.

Ollmann, G. (2004) The Phishing Guide, Understanding and Preventing Phishing Attacks (Online Available) :http://www.nextgenss.com/papers/NISR-WP-Phishing.pdf.

Olsen, S. (2004) AOL tests caller ID for e-mail, CNET News.com, January 22, 2004.

Orr, W., (1999) ABAecom poised for e-commerce growth, ABA Banking Journal, Vol. 91, No. 10, (pp. 84–88).

Pan, Y., and Ding, X. (2006) Anomaly Based Web Phishing Page Detection, *Proceedings of the 22nd Annual Computer Security Applications Conference (ACSAC'06)*, (pp. 381-392), Computer Society.

PassMark, (2005) Two-Factor Two-Way Authentication, PassMark Security. http://www.passmarksecurity.com.

Perez, J. (2003) Yahoo airs antispam initiative, ComputerWeekly.com, December 8, 2003.

Persson, A. (2007) Exploring Phishing Attacks and Countermeasures", Master Thesis in Computer Science, Thesis No: MCS-2007:18.

Pettersson, J., Fischer-Hübner, S., Danielsson, N., Nilsson, J., Bergmann, M., Clauss, S., Kriegelstein, T., and Krasemann, H. (2005) Making prime usable, *in Proceedings of SOUPS '05*, (pp. 53-64). ACM Press, Pittsburgh, PA, USA.

Phishtank, (2008),http://www.phishtank.com/phish_archive.php, Access date [14/11/2008].

Qaddoum S. Kifaya., D. (2009) Mining Student Evolution Using Associative Classification and Clustering, Communications of the IBIMA, Volume 11, ISSN: 1943-7765.

Qi, M., Yang, C. (2006) Research and Design of Phishing Alarm System at Client Terminal, *Proceedings of the IEEE Asia-Pacific Conference on Services Computing (APSCC'06)*, (pp. 597-600), IEEE.

Quinlan, J. (1979) Discovering rules from large collections of examples: a case study. In D. Michie, editor, Expert Systems in the Micro-electronic Age, (pp.168-201). Edinburgh.

Quinlan, J. (1986) Induction of decision trees. Machine Learning, Vol. 1, No. 1, (pp. 81 – 106).

Quinlan, J. (1987a) Generating production rules from decision trees. *Proceedings of the 10th Int. Joint Conferences on Artificial Intelligence*, (pp. 304-307). Milan, Italy.

Quinlan, J. (1993) C4.5: Programs for machine learning. San Mateo, CA: Morgan Kaufmann.

Quinlan, J. (1996) Improved use of continuous attributes in c4.5. Journal of Artificial Intelligence Research, Vol. 4, No. 1, (pp. 77-90).

Quinlan, J. (1998) Data mining tools See5 and C5.0. Technical Report, RuleQuest Research.

Quinlan, J., and Cameron-Jones, R. (1993) FOIL: A midterm report. *Proceedings of the European Conference on Machine Learning*, (pp. 3-20), Vienna, Austria.

Rissanen, J., (1985) The minimum description length principle. In: Kotz, S., Johnson, N. (Eds.), Encyclopedia of Statistical Sciences, Vol. 5, (pp. 523–527).

Ross, B., Jackson, C., Miyake, N., Boneh, D., and Mitchell, J. (2005) Stronger Password Authentication Using Browser Extensions. *Proceedings of the 14th Usenix Security Symposium*.

Salton, G., Wong, A., and Yang, C. (1975) A Vector Space Model for Information Retrieval, J. Am. Soc. Information Science, Vol. 18, No. 11, (pp. 613-620).

Schneider, F., Provos, N., Moll, R., Chew, M., and Rakowski, B. (2007) Phishing Protection Design Documentation. http://wiki.mozilla.org/Phishing_Protection:_Design_Documentation, Access date [23/10/2007].

Seker, R. (2006) Protecting Users against Phishing Attacks with AntiPhish, Journal, Computer Software and Applications, Vol. 13, No. 8, (pp. 517-524).

Shah, S., (2003) Measuring Operational Risks using Fuzzy Logic Modeling, Article, Towers Perrin, JULY 2003.

Sharif, T. (2005) Phishing Filter in IE7, http://blogs.msdn.com/ie/archive/2005/09/09/463204.aspx, Access date [4/6/2007].

Sodiya, A., Onashoga, A., and Oladunjoye, B. (2007) Threat modeling using fuzzy logic paradigm. Journal of Issues in Informing Science and Information Technology. U. S. A, Vol. 15, No. 4, (pp. 53.61).

Srisawat, A., and Kijsirikul, B. (2004) Using Associative Classification for Predicting HIV-1 Drug Resistance, *Proceedings of the Fourth International Conference on Hybrid Intelligent Systems*, (pp. 280-284), 0-7695-2291-2/04, (HIS'04).

Stepp, M. (2005) Phishhook: A tool to detect and prevent phishing attacks. In DIM ACS Workshop on Theft in E-Commerce: Content, Identity, and Service.

Suh, B., and Han, I. (2002) Effect of trust on customer acceptance of Internet banking. Electronic Commerce Research and Applications, Vol. 1, No. 3, (pp. 247-263).

Sukkar, A., and Hasan, H. (2005) Toward a Model for the Acceptance of Internet Banking in Developing Countries, Information Technology for Development, Vol. 11, No.4, (pp. 381–398).

Thabtah, F., Peter, C., and Peng, Y. (2005) MCAR: Multi-class Classification based on Association Rule. IEEE International Conference on Computer Systems and Applications, (pp. 127-133).

Watson, D., Holz, T., and Mueller, S. (2005) Know your enemy: Phishing, behind the scenes of Phishing attacks, The Honeynet Project & Research Alliance.

Weider, D., Nargundkar, Y., and Tiruthani, N. (2008) Phishing Vulnerability Analysis of Web Based Systems, 978-1-4244-2703-1/08, IEEE.

WEKA, (2006) Data Mining with Open Source Machine Learning Software in Java, http://www.cs.waikato.ac.nz/ ml/weka, Access date [31/01/2007].

Willem, D., and Eloff, J. (1997) Enhanced Password Authentication through Fuzzy Logic. IEEE Expert Nov-Dec num. 1. (pp. 38-44).

Witten, I., and Frank, E. (2000) Data mining: practical machine learning tools and techniques with Java implementations. San Francisco: Morgan Kaufmann.

Wood, L. (2005) Document Object Model Level 1 Specification, http://www.w3.org.

Wu, M. (2006) Fighting Phishing at the User Interface, PhD Thesis in Computer Science and Engineering.

Wu, M., Miller, R., and Garfinkel, S. (2006) Do Security Toolbars Actually Prevent Phishing Attacks?, CHI.

Wu, M., Miller, R., and Little, G. (2006a) Web Wallet: Preventing Phishing Attacks by Revealing User Intentions, MIT Computer Science and Artificial Intelligence Lab.

Wu, M., Miller, R., and Little, G. (2006b) Do Security Toolbars Actually Prevent Phishing Attacks?, CHI.

Ye, Z., and Smith, S. (2005) Trusted paths for browsers. ACM Transactions on Information and System Security, Vol. 8, No. 2, (pp. 153–186).

Yin, X., and Han, J. (2003) CPAR: Classification based on predictive association rule. *Proceedings of the SDM* (pp. 369-376). San Francisco, CA.

Yu, S., Cai, D., Wen, J., and Ma, W. (2003) Improving Pseudo-Relevance Feedback in Web Information Retrieval Using Web Page Segmentation, *Proc. 14th Int'l Conf. World Wide Web*, (pp. 11-18).

Zadeh, L. (1965) Fuzzy Sets, Information and Control, Vol. 8, New York: Academic Press, (pp. 338-353).

Zadhe, L. (1973) Outline of a New Approach to the Analysis of Complex Systems and Decision Process, IEEE Trans. Systems, Man, and Cybernetics, SMC-3, (pp. 28-44).

Zdziarski, J., Yang, W., and Judge, V. (2003) Approaches to Phishing Identification Using Match and Probabilistic Digital Fingerprinting Techniques, [Online]. Available: http://zdziarski.com/papers/phishing.pdf, CipherTrust, Inc.

Zhang, Y., Egelman, S., Cranor, L., and Hong, J. (2006) Phinding Phish: Evaluating Anti-Phishing Tools, Carnegie Mellon University, White Paper.

Zimmermann, H. (1996) Fuzzy Set Theory and Its Applications, Third Edition, Kluwer Academic Press, Boston.

Zin, A., and Yunos, Z. (2005) How To Make Online Banking Secure, article published in The Star InTech.

Appendix A

Rules for Source Code & Java Script Criteria

CBA Rules:

Num of Test Case : 2178; Correct Prediction : 2025; Error Rate : 7.024%
MinSup: 10.000%, MinConf: 80.000%

Rule 1: Using_onMouseOver_to_hide_the_Link = High
Redirect_pages = High -> class = Fraud
Rule 2: Using_onMouseOver_to_hide_the_Link = High
Straddling_attack = High -> class = Fraud
Rule 3: Server_Form_Handler_[SFH] = Low
Using_onMouseOver_to_hide_the_Link = Low
Pharming_Attack = Low -> class = Genuine
Rule 4: Server_Form_Handler_[SFH] = Low
Using_onMouseOver_to_hide_the_Link = Low -> class = Genuine
Rule 5: Server_Form_Handler_[SFH] = Low
Pharming_Attack = Low -> class = Genuine
Rule 6: Using_onMouseOver_to_hide_the_Link = Low
Straddling_attack = Low -> class = Genuine
Rule 7: Server_Form_Handler_[SFH] = Low
Straddling_attack = Low -> class = Genuine
Rule 8: Using_onMouseOver_to_hide_the_Link = High
Pharming_Attack = High -> class = Fraud
Rule 9: Pharming_Attack = Moderate -> class = Doubtful
Rule 10: Pharming_Attack = Low -> class = Genuine
Rule 11: Using_onMouseOver_to_hide_the_Link = Moderate -> class = Doubtful

Num of Rules: 11

Sample of JRIP rules:
==========

Correctly Classified Instances	2156	98.9899 %
Incorrectly Classified Instances	22	1.0101 %
Mean absolute error	0.0133	

=== Confusion Matrix ===

a	b	c	<-- classified as
704	7	0	a = Genuine
2	885	0	b = Doubtful
0	13	567	c = Fraud

(Using_onMouseOver_to_hide_the_Link = High) and (Redirect_pages = High) =>
Source_Code_&_Java_script_Phishing_Risk=Fraud (312.0/0.0)
(Using_onMouseOver_to_hide_the_Link = High) and (Straddling_attack = High) =>
Source_Code_&_Java_script_Phishing_Risk=Fraud (155.0/0.0)
(Pharming_Attack = Low) and (Server_Form_Handler_(SFH) = Low) =>
Source_Code_&_Java_script_Phishing_Risk=Genuine (308.0/0.0)

(Straddling_attack = Low) and (Using_onMouseOver_to_hide_the_Link = Low) and
(Server_Form_Handler_(SFH) = Moderate) => Source_Code_&_Java_script_Phishing_Risk=Genuine
(115.0/0.0)
(Straddling_attack = Low) and (Server_Form_Handler_(SFH) = Low) and
(Using_onMouseOver_to_hide_the_Link = Low) =>
Source_Code_&_Java_script_Phishing_Risk=Genuine (69.0/0.0)

Number of Rules: 14

Sample of PART decision list

Using_onMouseOver_to_hide_the_Link = High AND
Pharming_Attack = Low: Fraud (156.0)
Using_onMouseOver_to_hide_the_Link = High AND
Pharming_Attack = Moderate: Doubtful (156.0/3.0)
Using_onMouseOver_to_hide_the_Link = High AND
Straddling_attack = High: Fraud (155.0)
Server_Form_Handler_(SFH) = Low AND
Pharming_Attack = Low: Genuine (308.0)
Server_Form_Handler_(SFH) = Moderate AND
Straddling_attack = Low: Genuine (115.0)

Number of Rules: 22

Sample of Prism rules

If Server_Form_Handler_(SFH) = High
  and Redirect_pages = High
  and Using_onMouseOver_to_hide_the_Link = Moderate then Fraud
If Server_Form_Handler_(SFH) = High
  and Pharming_Attack = High
  and Straddling_attack = High then Fraud
If Pharming_Attack = Moderate
  and Redirect_pages = Low
  and Server_Form_Handler_(SFH) = Moderate then Doubtful
If Using_onMouseOver_to_hide_the_Link = Low
  and Server_Form_Handler_(SFH) = Low
  and Pharming_Attack = Low then Genuine
If Straddling_attack = Low
  and Using_onMouseOver_to_hide_the_Link = Low
  and Server_Form_Handler_(SFH) = Moderate then Genuine

Number of Rules: 59

Sample of J48 pruned tree

```
Using_onMouseOver_to_hide_the_Link = Low
|   Server_Form_Handler_(SFH) = Low
|   |   Pharming_Attack = Low: Genuine (274.0)
|   |   Pharming_Attack = Moderate
|   |   |   Redirect_pages = Low: Genuine (0.0)
|   |   |   Redirect_pages = Moderate: Genuine (69.0)
|   |   |   Redirect_pages = High: Doubtful (35.0)
|   |   Pharming_Attack = High
|   |   |   Straddling_attack = Low: Genuine (0.0)
|   |   |   Straddling_attack = Moderate: Genuine (36.0)
|   |   |   Straddling_attack = High: Doubtful (35.0/1.0)
|   Server_Form_Handler_(SFH) = Moderate
|   |   Straddling_attack = Low: Genuine (115.0)
|   |   Straddling_attack = Moderate
|   |   |   Pharming_Attack = Low: Genuine (22.0)
|   |   |   Pharming_Attack = Moderate: Doubtful (23.0/1.0)
|   |   |   Pharming_Attack = High: Doubtful (22.0/1.0)
|   |   Straddling_attack = High
```

Number of Leaves: 43
Size of the tree: 64

Rules for Web Address Bar Criteria

CBA Rules:

Num of Test Case : 2178; Correct Prediction :2034; Error Rate : 6.611%
MinSup: 10.000%, MinConf: 80.000%

Rule 1:	Using_the_@_Symbol_to_Confuse = High Long_URL_address = High	->	class = Fraud
Rule 2:	Using_the_@_Symbol_to_Confuse = High Replacing_similar_characters_for_URL = High	->	class = Fraud
Rule 3:	Using_Hexadecimal_Character_Codes = Low Using_the_@_Symbol_to_Confuse = Low Adding_a_prefix_or_suffix = Low	->	class = Genuine
Rule 4:	Using_Hexadecimal_Character_Codes = Low Using_the_@_Symbol_to_Confuse = Low	->	class = Genuine
Rule 5:	Using_Hexadecimal_Character_Codes = Low Adding_a_prefix_or_suffix = Low	->	class = Genuine
Rule 6:	Using_the_@_Symbol_to_Confuse = Low Replacing_similar_characters_for_URL = Low	->	class = Genuine
Rule 7:	Using_Hexadecimal_Character_Codes = Low Replacing_similar_characters_for_URL = Low	->	class = Genuine
Rule 8:	Adding_a_prefix_or_suffix = Moderate	->	class = Doubtful
Rule 9:	Using_the_@_Symbol_to_Confuse = Moderate	->	class = Doubtful
Rule 10:	Replacing_similar_characters_for_URL = Moderate	->	class = Doubtful
Rule 11:	Using_Hexadecimal_Character_Codes = Moderate	->	class = Doubtful

Num of Rules: 11

Sample of JRIP rules:

==========

Correctly Classified Instances	2156	98.9899 %
Incorrectly Classified Instances	22	1.0101 %
Mean absolute error	0.0133	

=== Confusion Matrix ===

a	b	c	<-- classified as
695	0	0	a = Genuine
12	895	1	b = Doubtful
0	9	566	c = Fraud

(Adding_a_prefix_or_suffix = Low) and (Using_Hexadecimal_Character_Codes = Low) => Web_Address_Bar_Phishing_Risk=Genuine (308.0/5.0)
(Replacing_similar_characters_for_URL = Low) and (Using_the_@_Symbol_to_Confuse = Low) and (Using_Hexadecimal_Character_Codes = Moderate) => Web_Address_Bar_Phishing_Risk=Genuine (115.0/1.0)
(Using_the_@_Symbol_to_Confuse = High) and (Long_URL_address = High) => Web_Address_Bar_Phishing_Risk=Fraud (312.0/1.0)
(Using_the_@_Symbol_to_Confuse = High) and (Replacing_similar_characters_for_URL = High) => Web_Address_Bar_Phishing_Risk=Fraud (155.0/0.0)
(Using_Hexadecimal_Character_Codes = High) and (Adding_a_prefix_or_suffix = High) and (Using_the_@_Symbol_to_Confuse = High) => Web_Address_Bar_Phishing_Risk=Fraud (41.0/0.0)

Number of Rules: 14

Sample of PART decision list

Using_the_@_Symbol_to_Confuse = High AND
Long_URL_address = High: Fraud (312.0/1.0)
Using_the_@_Symbol_to_Confuse = High AND
Replacing_similar_characters_for_URL = High: Fraud (155.0)
Using_Hexadecimal_Character_Codes = Low AND
Using_the_@_Symbol_to_Confuse = Low AND
Adding_a_prefix_or_suffix = Low: Genuine (274.0/5.0)
Using_Hexadecimal_Character_Codes = Moderate AND
Adding_a_prefix_or_suffix = High: Doubtful (160.0/2.0)
Using_Hexadecimal_Character_Codes = Moderate AND
Using_the_@_Symbol_to_Confuse = Low AND
Replacing_similar_characters_for_URL = Low: Genuine (115.0/1.0)

Number of Rules: 24

Sample of Prism rules

If Using_the_@_Symbol_to_Confuse = Low
  and Using_Hexadecimal_Character_Codes = Low
  and Replacing_similar_characters_for_URL = Low
  and Adding_a_prefix_or_suffix = Low then Genuine
If Using_the_@_Symbol_to_Confuse = Low
  and Using_Hexadecimal_Character_Codes = Low
  and Replacing_similar_characters_for_URL = Low
  and Long_URL_address = Moderate
  and Adding_a_prefix_or_suffix = Moderate then Genuine
If Replacing_similar_characters_for_URL = Moderate
  and Using_the_@_Symbol_to_Confuse = Low

and Adding_a_prefix_or_suffix = High
and Long_URL_address = Low
and Using_Hexadecimal_Character_Codes = Low then Doubtful
If Using_the_@_Symbol_to_Confuse = High
and Long_URL_address = High
and Adding_a_prefix_or_suffix = High then Fraud
If Using_the_@_Symbol_to_Confuse = High
and Replacing_similar_characters_for_URL = High
and Long_URL_address = Low then Fraud

Number of Rules: 59

Sample of J48 pruned tree

```
Using_the_@_Symbol_to_Confuse = Low
|  Using_Hexadecimal_Character_Codes = Low
|  |  Adding_a_prefix_or_suffix = Low: Genuine (274.0/5.0)
|  |  Adding_a_prefix_or_suffix = Moderate
|  |  |  Long_URL_address = Low: Genuine (0.0)
|  |  |  Long_URL_address = Moderate: Genuine (70.0/1.0)
|  |  |  Long_URL_address = High: Doubtful (35.0)
|  |  Adding_a_prefix_or_suffix = High
|  |  |  Replacing_similar_characters_for_URL = Low: Doubtful (0.0)
|  |  |  Replacing_similar_characters_for_URL = Moderate: Genuine (36.0/1.0)
|  |  |  Replacing_similar_characters_for_URL = High: Doubtful (35.0)
|  Using_Hexadecimal_Character_Codes = Moderate
|  |  Replacing_similar_characters_for_URL = Low: Genuine (115.0/1.0)
|  |  Replacing_similar_characters_for_URL = Moderate
|  |  |  Adding_a_prefix_or_suffix = Low: Genuine (22.0)
|  |  |  Adding_a_prefix_or_suffix = Moderate: Doubtful (23.0)
|  |  |  Adding_a_prefix_or_suffix = High: Doubtful (22.0/1.0)
|  |  Replacing_similar_characters_for_URL = High
```

Number of Leaves: 43
Size of the tree: 64

Rules for Page Style & Contents Criteria

CBA Rules:

Num of Test Case : 2178; Correct Prediction :2072; Error Rate :4.866%
MinSup: 10.000%, MinConf: 80.000%

Rule 1: using_Pop-Ups_windows = High
Spelling_errors = High -> class = Fraud
Rule 2: using_Pop-Ups_windows = High
Copying_website = High -> class = Fraud
Rule 3: Disabling_Right-Click = Low
using_Pop-Ups_windows = Low
Using_forms_with_“Submit”_button = Low -> class = Genuine
Rule 4: Disabling_Right-Click = Low

using_Pop-Ups_windows = Low -> class = Genuine
Rule 5: Disabling_Right-Click = Low
Using_forms_with_“Submit”_button = Low -> class = Genuine
Rule 6: using_Pop-Ups_windows = Low
Copying_website = Low -> class = Genuine
Rule 7: Disabling_Right-Click = Low
Copying_website = Low -> class = Genuine
Rule 8: using_Pop-Ups_windows = High
Using_forms_with_“Submit”_button = High -> class = Fraud
Rule 9: Using_forms_with_“Submit”_button = Moderate -> class = Doubtful
Rule 10: using_Pop-Ups_windows = Moderate -> class = Doubtful
Rule 11: Copying_website = Moderate -> class = Doubtful
Rule 12: Disabling_Right-Click = Moderate -> class = Doubtful

Num of Rules: 12

Sample of JRIP rules:

==========

Correctly Classified Instances	2164	99.3572 %
Incorrectly Classified Instances	14	0.6428 %
Mean absolute error	0.0084	

=== Confusion Matrix ===

a	b	c	<-- classified as
702	1	0	a = Genuine
3	896	1	b = Doubtful
0	9	566	c = Fraud

(Using_forms_with_“Submit”_button = Low) and (Disabling_Right-Click = Low) => Page_Style_&_Contents_Phishing_Risk=Genuine (308.0/0.0)
(Copying_website = Low) and (using_Pop-Ups_windows = Low) and (Disabling_Right-Click = Moderate) => Page_Style_&_Contents_Phishing_Risk=Genuine (115.0/1.0)
(Disabling_Right-Click = High) and (using_Pop-Ups_windows = Moderate) and (Spelling_errors = High) => Page_Style_&_Contents_Phishing_Risk=Fraud (39.0/0.0)
(using_Pop-Ups_windows = High) and (Spelling_errors = High) => Page_Style_&_Contents_Phishing_Risk=Fraud (312.0/0.0)
(using_Pop-Ups_windows = High) and (Copying_website = High) => Page_Style_&_Contents_Phishing_Risk=Fraud (155.0/0.0)

Number of Rules: 14

Sample of PART decision list

----

using_Pop-Ups_windows = High AND
Using_forms_with_“Submit”_button = Moderate: Doubtful (158.0/1.0)
using_Pop-Ups_windows = High AND
Copying_website = High: Fraud (311.0)
using_Pop-Ups_windows = High AND
Spelling_errors = High: Fraud (156.0)
Disabling_Right-Click = Low AND
Using_forms_with_“Submit”_button = Low: Genuine (308.0)
Copying_website = Moderate AND
Disabling_Right-Click = Low: Genuine (36.0)

Number of Rules: 21

Sample of Prism rules

```
If using_Pop-Ups_windows = Low
   and Disabling_Right-Click = Low
   and Using_forms_with_“Submit”_button = Low then Genuine
If Copying_website = Low
   and using_Pop-Ups_windows = Low
   and Spelling_errors = Low then Genuine
If Using_forms_with_“Submit”_button = Moderate
   and using_Pop-Ups_windows = High
   and Disabling_Right-Click = Low then Doubtful
If Disabling_Right-Click = High
   and Using_forms_with_“Submit”_button = High
   and Copying_website = High
   and Spelling_errors = Low
   and using_Pop-Ups_windows = Low then Fraud
If Using_forms_with_“Submit”_button = High
   and Spelling_errors = High
   and Disabling_Right-Click = Low
   and Copying_website = Moderate
   and using_Pop-Ups_windows = Moderate then Fraud
```

Number of Rules: 49

Sample of J48 pruned tree

```
using_Pop-Ups_windows = Low
|   Disabling_Right-Click = Low
|   |   Using_forms_with_“Submit”_button = Low: Genuine (274.0)
|   |   Using_forms_with_“Submit”_button = Moderate
|   |   |   Copying_website = Low: Genuine (69.0/1.0)
|   |   |   Copying_website = Moderate: Doubtful (35.0)
|   |   |   Copying_website = High: Genuine (0.0)
|   |   Using_forms_with_“Submit”_button = High
|   |   |   Copying_website = Low: Genuine (0.0)
|   |   |   Copying_website = Moderate: Genuine (36.0)
|   |   |   Copying_website = High: Doubtful (35.0/1.0)
|   Disabling_Right-Click = Moderate
```

Number of Leaves: 43
Size of the tree: 64

Rules for Social Human Factor Criteria

CBA Rules:

Num of Test Case : 2178; Correct Prediction : 2063; Error Rate : 5.280%
MinSup: 10.000%, MinConf: 80.000%

Rule 1: Public_generic_salutation = Low
Much_emphasis_on_security_and_response = Low -> class = Genuine
Rule 2: Public_generic_salutation = High
Much_emphasis_on_security_and_response = High -> class = Fraud
Rule 3: Buying_Time_to_Access_Accounts = Moderate
Public_generic_salutation = Low -> class = Genuine
Rule 4: Buying_Time_to_Access_Accounts = Moderate
Much_emphasis_on_security_and_response = Low -> class = Genuine
Rule 5: Buying_Time_to_Access_Accounts = High
Public_generic_salutation = High -> class = Fraud
Rule 6: Buying_Time_to_Access_Accounts = Low
Much_emphasis_on_security_and_response = Low -> class = Genuine
Rule 7: Buying_Time_to_Access_Accounts = Low
Much_emphasis_on_security_and_response = Moderate -> class = Genuine
Rule 8: Public_generic_salutation = Low
Much_emphasis_on_security_and_response = Moderate -> class = Genuine
Rule 9: Buying_Time_to_Access_Accounts = Low
Public_generic_salutation = Low -> class = Genuine
Rule 10: Much_emphasis_on_security_and_response = Low -> class = Genuine
Rule 11: Buying_Time_to_Access_Accounts = Low -> class = Genuine
Rule 12: Buying_Time_to_Access_Accounts = High -> class = Fraud

Num of Rules: 12

Sample of JRIP rules:

==========

Correctly Classified Instances	2109	96.832 %
Incorrectly Classified Instances	69	3.168 %
Mean absolute error	0.032	

=== Confusion Matrix ===

a	b	c	<-- classified as
1450	50	0	a = Genuine
19	98	0	b = Doubtful
0	0	561	c = Fraud

(Public_generic_salutation = Moderate) and (Buying_Time_to_Access_Accounts = Moderate) and (Much_emphasis_on_security_and_response = High) => Social_Human_Factor_Phishing_Risk=Doubtful (52.0/19.0)
(Much_emphasis_on_security_and_response = Moderate) and (Buying_Time_to_Access_Accounts = Moderate) and (Public_generic_salutation = High) => Social_Human_Factor_Phishing_Risk=Doubtful (57.0/19.0)
(Public_generic_salutation = Moderate) and (Much_emphasis_on_security_and_response = Moderate) and (Buying_Time_to_Access_Accounts = High) => Social_Human_Factor_Phishing_Risk=Doubtful (39.0/12.0)
(Buying_Time_to_Access_Accounts = High) and (Public_generic_salutation = High) => Social_Human_Factor_Phishing_Risk=Fraud (226.0/0.0)
(Much_emphasis_on_security_and_response = High) and (Public_generic_salutation = High) => Social_Human_Factor_Phishing_Risk=Fraud (207.0/0.0)

Number of Rules: 7

Sample of PART decision list

Public_generic_salutation = Low AND
Buying_Time_to_Access_Accounts = Low: Genuine (424.0/3.0)
Much_emphasis_on_security_and_response = Low AND
Buying_Time_to_Access_Accounts = Moderate: Genuine (233.0)
Public_generic_salutation = Low AND
Buying_Time_to_Access_Accounts = High AND
Much_emphasis_on_security_and_response = High: Fraud (86.0)
Public_generic_salutation = High: Fraud (139.0)
Much_emphasis_on_security_and_response = High: Fraud (42.0)

Number of Rules: 15

Sample of Prism rules

If Much_emphasis_on_security_and_response = Low
  and Public_generic_salutation = Low then Genuine
If Public_generic_salutation = Low
  and Buying_Time_to_Access_Accounts = Moderate then Genuine
If Buying_Time_to_Access_Accounts = Low
  and Public_generic_salutation = Moderate then Genuine
If Public_generic_salutation = High
  and Buying_Time_to_Access_Accounts = Low
  and Much_emphasis_on_security_and_response = Low then Doubtful
If Buying_Time_to_Access_Accounts = High
  and Public_generic_salutation = High then Fraud
If Much_emphasis_on_security_and_response = High
  and Public_generic_salutation = High then Fraud

Number of Rules: 26

Sample of J48 pruned tree

Public_generic_salutation = Low
| Buying_Time_to_Access_Accounts = Low: Genuine (424.0/3.0)
| Buying_Time_to_Access_Accounts = Moderate: Genuine (273.0)
| Buying_Time_to_Access_Accounts = High
| | Much_emphasis_on_security_and_response = Low: Genuine (79.0)
| | Much_emphasis_on_security_and_response = Moderate: Genuine (59.0/1.0)
| | Much_emphasis_on_security_and_response = High: Fraud (86.0)
Public_generic_salutation = Moderate
| Buying_Time_to_Access_Accounts = Low: Genuine (185.0)
| Buying_Time_to_Access_Accounts = Moderate
| | Much_emphasis_on_security_and_response = Low: Genuine (51.0)
| | Much_emphasis_on_security_and_response = Moderate: Genuine (43.0/12.0)
| | Much_emphasis_on_security_and_response = High: Doubtful (52.0/19.0)
| Buying_Time_to_Access_Accounts = High

Number of Leaves: 19
Size of the tree: 28