

Detecting Phishing Emails Using Machine Learning Techniques

الكشف عن رسائل البريد الإلكتروني الخداع عن طريق تقنيات

تعليم الآلة

Prepared By

Sa'id Abdullah Al-Saaidah

Supervisor

Dr. Oleg Viktorov

Thesis Submitted in Partial Fulfillment of the requirements

for the Degree of Master of Computer Science

Department Of Computer Science

Faculty of Information Technology

Middle East University

January, 2017

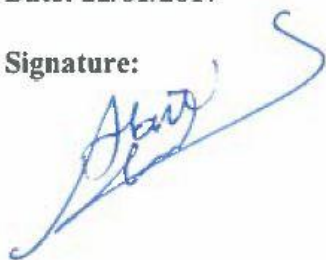
Authorization statement

I, Sa'id Abdullah Al-Saaidah, authorize the Middle East University to provide hard copies or soft copies of my Thesis to libraries, institutions or individuals upon their request.

Name: Sa'id Abdullah Al Saaidah

Date: 22/01/2017

Signature:

A handwritten signature in blue ink, appearing to read 'Sa'id', with a long, sweeping horizontal stroke extending to the right.

أقرار تفويض

أنا سعيد عبدالله عبد الربيع السعايده افوض جامعة الشرق الاوسط بتزويد نسخ من رسالتي ورقيا
والكترونيا للمكتبات، او المنظمات، او المؤسسات والهيئات او الافراد عند طلبها.

الاسم : سعيد عبدالله السعايده

التاريخ: 2017\01\22

التوقيع: 

Middle East University

Examination Committee Decision

This is to certify that the thesis entitled "Detecting Phishing Emails Using Machine Learning Techniques" was successfully defended and approved in 22 / 01 / 2017.

Examination Committee Members

Signature

Mudhafar Munir Fahmi Al-Jarrah (Internal Member & Chairman)

Assistant Professor, Department of Computer Science

Middle East University (MEU)

Dr Ahmad Adel Abu- Shareha (Supervisor)

Assistant Professor, Department of Computer Science

Middle East University (MEU)

On Behalf of Dr. Oleg Viktorov.

Dr. Mohammad Ahmad Alia (External Member)

Associate Professor, Department of Computer Information Systems

Al Zaytoonah University

Acknowledgment

Foremost, great thank for Allah, afterwards, I would like to express my sincere appreciation for my Supervisor Dr. Oleg Viktorov. A Special thanks and appreciation for Dr. Ahamed Abu Shareha for his guidance and support throughout the preparation of the study, which highly contributed to the success of this study. Special thanks for the examination committee for their valuable comments and suggestions that enriched my thesis. last but not least a special thanks to Dr. Ammar AlMomani for his great assistance and support.

Sai'd Saaidah

Dedication

To the soul of my father, my precious Mother, supportive soulmate (Lama),

lovely Son (Hamzah) and my wonderful Brothers and Sister

Table of Content

Subject	Page
Title.....	I
Authorization Statement.....	II
Examination Committee Decision.....	IV
Acknowledgment.....	V
Dedication.....	VI
Table of Contents.....	VII
List of Tables.....	IX
List of Figures.....	X
List of Abbreviations.....	XI
Abstract.....	XII

Chapter One

1.1	Introduction.....	1
1.2	Problem Statement.....	4
1.3	Objective of Study.....	5
1.4	Motivation.....	6
1.5	Scope and Limitation	6
1.6	Contribution	7
1.7	Thesis Organization	7

Chapter Two

2.1	Introduction	8
2.2	Phishing Emails Detection Techniques.....	9
	2.2.1 Traditional Methods.....	9
	2.2.2 Automated Methods.....	12
2.3	Literature Review	15

Chapter Three		
3.1	Introduction.....	25
3.2	Proposed Approach.....	25
	3.2.1 Pre-Processing	26
	3.2.2 Feature Selection	31
	3.2.3 Algorithms Evaluation.....	33
	3.2.4 Feature Clustering.....	33
	3.2.5 Multi-Classification Integration Approach for Phishing Email Detection	34
Chapter Four		
4.1	Data Set.....	36
4.2	Tools.....	38
4.3	Experimental Results.....	39
	4.3.1 Evaluation Measures.....	39
	4.3.2 Experimental Results on Features Selection	40
Chapter Five		
5.1	Conclusion.....	53
5.2	Future Work.....	54
	Reference.....	55

List of Tables

Table No.	Title	Page No.
Table (2.1)	Phishing Detection Tools.....	15
Table (3.1)	The Selected Body Features.....	27
Table (3.2)	The Selected Email Header Features.....	28
Table (3.3)	The Selected URL Features.....	29
Table (3.4)	The Selected Java Script and External Features.....	30
Table (3.5)	The Groups of Manual Features Selection.....	32
Table (3.6)	The Groups of Automated Features Selection.....	33
Table (4.1)	Confusion Matrix.....	40
Table (4.2)	The Results of Test on AI Features Together.....	40
Table (4.3)	The Results of Test on the Body Feature Only.....	41
Table (4.4)	The Results of the Test on the URL Feature Only.....	42
Table (4.5)	The Results of the Test on the Header Feature Only.....	43
Table (4.6)	The Results of the Test on Java Script and External Features.....	44
Table (4.7)	The Results of the Test on all Features Excluding the Body Features.....	45
Table (4.8)	The Results of the Test on All Features Excluding Java Script and External features.....	46
Table (4.9)	The Results of the Test on All Features Excluding Header Features.....	47
Table (4.10)	The Results of the Test on all Features Excluding URL Features	48
Table (4.11)	Accuracy for Automated Generated Groups.....	49
Table (4.12)	Accuracy for Both Manual and Automated Features Selection...	50
Table (4.13)	The Results of Test Multi- Classifier Integration.....	52

List of Figures

Figure No.	Title	Page No.
Figure (1.1)	Phishing Life Cycle.....	2
Figure (1.2)	Types of Phishing E-mails.....	2
Figure (1.3)	Automated Phishing Email Detection.....	3
Figure (2.1)	Support Vector Machine.....	14
Figure (3.1)	Architectural Design of the Proposed Approach.....	26
Figure (3.2)	Manual and Automated Feature Groups.....	31
Figure (3.3)	Sample of the Dataset with K-means Clustering	34
Figure (3.4)	Multi-Classifer Integration Model	35
Figure (4.1)	Sample of the Dataset 47 Feature.....	37
Figure (4.2)	Sample of the Dataset 47 Feature.....	38
Figure (4.3)	Accuracy of the Five Algorithms on all Features Together Test.....	41
Figure (4.4)	Accuracy of the Five Algorithms on Body Feature Only Test...	42
Figure (4.5)	Accuracy of the Five Algorithms on URL Feature Only Test...	43
Figure (4.6)	Accuracy of the Five Algorithms on Header Feature Only Test	44
Figure (4.7)	Accuracy of the Five Algorithms on Java Script and External Features Only Test	45
Figure (4.8)	Accuracy of the Five Algorithms on all Features Excluding Body Feature Test.....	46
Figure (4.9)	Accuracy of the Five Algorithms on all Features Excluding Java Script and External Features Test.....	47
Figure (4.10)	Accuracy of the Five Algorithms on all Features Excluding Header Feature Test.....	48
Figure (4.11)	Accuracy of the Five Algorithms on all Features Excluding URL Feature Test.....	49
Figure (4.12)	Accuracy for all Automated Features.....	50
Figure (4.13)	Accuracy for Five Classifier Algorithms in Both Scenarios...	51
Figure (4.14)	Sample of the Dataset with the three selected classifiers for the integrated system.....	52

List of Abbreviation

Abbreviation	Description
MCSI	Microsoft Consumer Safety Index
LR	Logistic Regression
DT	Decision Tree
SMO	Sequential minimal optimization
ISP	Internet Server Providers
CART	Classification and Regression Trees
SVM	Support Vector Machine

Detecting Phishing Emails Using Machine Learning Techniques

Prepared By: Sa'id Abdullah Al-Saaidah

Supervisor : Dr. Oleg Viktorov

Abstract

Phishing is a fraud technique used for identity theft where users receive fake e-mails from deceiving addresses that seem as belonging to legitimate and real business in an attempt to steel the receiver's personal details. This act endangers the privacy of many users and therefore, researchers work continuously on finding detection tools and developing existing ones. Classification is one of the machine learning techniques that can be effectively used to detect received phishing emails.

Through this research, varied classification algorithms are discussed and compared, such as; Naïvebayes, Decision Tree (DT), Logistic Regression, Classification and Regression Trees and Sequential Minimal Optimization (SMO). A new system was built to detect the phishing emails in an integrating between the supervised and unsupervised technique. In addition, the study compares the manual and automated feature selection groups for the Email.

The experiment was executed using WEKA Tool on a dataset of 4800 Email, 2400 phishing emails and 2400 legitimate emails represented the 47 features of the email structure.

Indicated that the best manually selected groups achieved an equal accuracy level achieved by the automated features group of 98.25 percent. Also the Decision Tree, J48 and SMO classifiers topped the previously-mentioned algorithms by providing the highest accuracy average in both manual and automated scenarios.

Moreover, an integrated system of multiple classifiers was constructed using the three top algorithms of SMO, Decision Tree, and J48 and the results showed that integrating unsupervised techniques with supervised ones before the testing provides more accurate results of detecting phishing emails with 98.37 for all the features.

Keywords: Phishing Emails, Data mining, Clustering, Classification, Multi-classification

الكشف عن رسائل البريد الإلكتروني الخداع عن طريق تقنيات تعليم الآلة

اعداد: سعيد عبدالله السعايده

المشرف : د. اوليج فكتروف

الملخص

تعتبر رسائل البريد الإلكتروني المخادعة إحدى أساليب الاحتيال وتستخدم لسرقة البيانات الشخصية والمهمة للمستخدمين حيث يتلقى المستخدمون رسالة بريد إلكتروني وهمية من عناوين مخادعة والتي تبدو أنها تنتمي إلى الأعمال التجارية المشروعة والحقيقية في محاولة لسرقة المعلومات الشخصية للمتلقي. هذا العمل يشكل خطراً على خصوصية العديد من المستخدمين، وبالتالي، يعمل الباحثون بشكل مستمر على إيجاد أدوات الكشف عن هذا النوع من الرسائل الإلكترونية وتطوير القائم منها. التصنيف هو أحد الطرق المتبعة في تقنيات التنقيب عن البيانات التي يمكن استخدامها بشكل فعال للكشف عن رسائل البريد الإلكتروني المخادعة.

من خلال هذه الدراسة، تم بحث ومقارنة مجموعه من خوارزميات التصنيف المختلفة، مثل خوارزمية الناييف بيزين، شجرة القرارات، لوجستية الانحدار، شجرة التصنيف والانحدار و متسلسلة الحد الأدنى للتحسين. وبالإضافة إلى ذلك، تقارن الدراسة مجموعة مختارة من الخصائص للبريد الإلكتروني بالطريقة اليدوية والآلية.

لقد تم تنفيذ التجربة باستخدام أداة الويكا على قاعدة بيانات من 4800 بريد إلكتروني، 2400 رسائل التصيد والرسائل الإلكترونية المشروعة 2400 تمثل 47 ميزه من هيكل البريد الإلكتروني.

وأشارت النتائج إلى أن أفضل المجموعات المختارة يدويا حققت مستوى دقة مساوي للمجموعة التي تم اختيارها اليأ حيث حققت 98.25 في المائة. أيضا فقد تصدرت خوارزميات شجرة القرار، لوجستية الانحدار ومتسلسلة الحد الأدنى للتحسين من خلال توفير أعلى معدل دقة في كلا السيناريوهين اليدوية والآلية.

وعلاوة على ذلك، تم بناء نموذج متكامل من خوارزميات متعدد التصنيف حيث تم استخدام الخوارزميات الثلاثة الأولى والحاصلة على أعلى دقة خوارزمية وهي شجرة القرار، لوجستية الانحدار ومتسلسلة الحد الأدنى للتحسين وأظهرت النتائج أن دمج تقنيات غير خاضعة للرقابة أي لم يتم تجميعها مع تلك الخاضعة للرقابة أو التي تم تجميعها قبل الاختبار يوفر نتائج أكثر دقة للكشف عن الرسائل الإلكترونية الاحتيالية مع 98.37 في اختبار جميع خصائص البريد الإلكتروني.

كلمات البحث: رسائل البريد الإلكتروني المخادعة، التنقيب عن البيانات، التجميع، التصنيف والتصنيف المتعددة للخوارزميات

Chapter One

1.1 Introduction

In today's world, phishing is seen as a challenging threat growing rapidly every year. It is considered as a criminal act that integrates social-engineering and technical methods to steal confidential data of consumers such as usernames and passwords (Manning & Aron 2015). In that sense, Lungu and Tabusca argues that the current economic crisis is a reflection of the increasing attacks and violations of internet users' data (Lungu & Tabusca, 2010). Phishing techniques are classified into several types according to the applied channel of proliferation, these include malware, phishing emails, and bogus websites (Jain & Richariya 2011).

Phishing emails are categorized as spam messages. Users receive emails alleging to be from a legitimate company or bank and asking the user to follow an embedded link. The link will redirect the user to a fake website that requests confidential information, such as usernames, passwords or credit card numbers (Al-Momani and Gupta 2013).

Figure 1.1 illustrates the cycle of phishing technique. The process begins with sending emails to the targeted individuals' inboxes with an attempt to make them follow an included link. In that sense, online phishing is much like the traditional fishing; where in the later a fisher would use fishing bait and line to catch a fish, in the online technique, the phisher will send out as many emails as possible in an attempt to convince the biggest number of receivers to "catch" the bait and follow the embedded link (Al-Momani and Gupta 2013)..

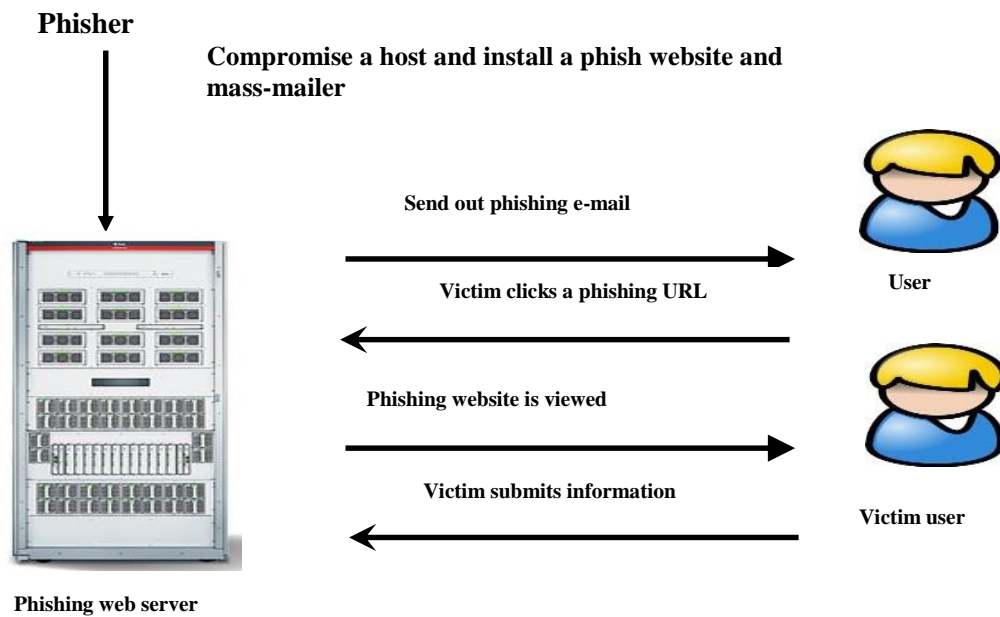


Figure (1.1) Phishing Lifecycle

Phishers rely on two techniques to achieve their goals; they either use the deceptive phishing method or the malware-based phishing (Figure 1.2). The first technique relies on social-engineering schemes by using emails to send deceptive links as these emails look a lot like coming from a real business or bank account, and direct the receiver to an affiliated fake website asking to fill in some required details that are confidential such as; usernames, passwords, credit card numbers, and personal information.

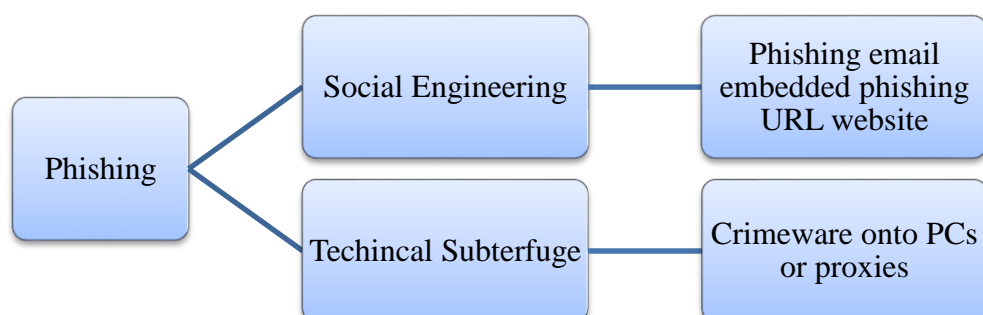


Figure (1.2) Types of Phishing E-mails

While the malware-based phishing technique does not directly ask for details, but it rather relies on malicious codes or malware and technical schemes if users click on the embedded link, or looks for security gaps in the receivers' devices to obtain their online account information directly. Sometimes, the phisher will attempt to misdirect the user to a fake website or a legitimate one monitored by substitutions (Al-Momani, 2013)

An online report was published in 2012 indicating an estimated loss of \$1.5 billion which the report attributes to the effect of phishing attacks. This huge loss and threat are on the rise which calls for finding more efficient detection techniques of such phishing emails to control the damage and reduce the risk (Akinyelu, 2014).

Phishing detection techniques function by extracting values from the examined emails by using pre-defined set of features in order to classify the email as phishing or not. The classification is achieved relying on extracted feature vectors and with reference to a trained model (Figure 1.3).

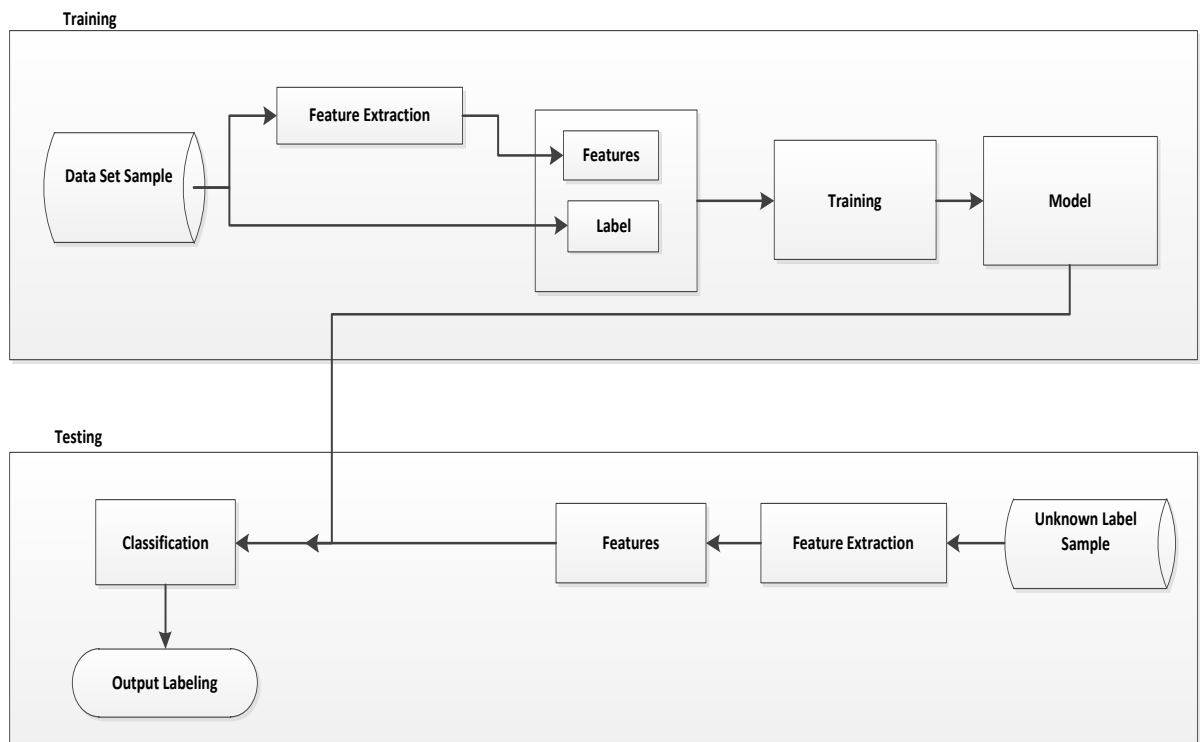


Figure (1.3) Automated Phishing Email Detection

1.2 Problem Statement

Phishing is technique used to steel personal information for the purposes of identity theft and using fake e-mail messages that appear to come from legitimate businesses. This is usually done by sending emails that seem to come from reliable source to gain access to person's confidential and private information.

Phishing emails considers as the fastest rising online crime method used for stealing personal financial data and perpetrating identity theft. Individuals who respond to phishing e-mails, and input the requested financial or personal information into e-mails, websites, or pop-up windows put themselves and their institutions at risk.

The Microsoft Consumer Safety Index survey showed that the annual worldwide impact of phishing email was US \$5 billion. On the other hand, the cost of repairing their impact is US \$6 billion (MCSI reveals the impact of poor online safety behaviors in Singapore, 2014).

With the massive work exists for phishing email detection task, there is no set of features that has been determined as the best to detected phishing. Moreover, the same nondeterministic scenario is applied for the underling classification algorithm. Finally, there is a need to keep on enhancing the accuracy of the detection techniques. Overall the problems carried out in this research are as following:

- How to determine the best set of features to be used with phishing detection.
- How to select the best classification algorithm to be used for phishing detection.

- How to enhance the performance of the best selected features and classifiers.
- How to integrate multiple classification algorithms for phishing detection and to evaluate such integration.

1.3 Objectives of the Study

The goal of this research is to conduct a comparative assessment between various classification data mining algorithms techniques, and various feature selection scenarios (manual feature selection and automated feature selection groups). Moreover, the goal includes the development of multi-classifier integration model by combining clustering and more than one classification technique to enhance detection and protecting phishing emails.

The objectives of this research are as follows:

- Determine and evaluate the best set of features to be used for phishing Emails detection using Manual feature selection based on the Email structure and automated selection techniques.
- Integrate between unsupervised Machine learning technique with the best supervised machine-learning algorithms to enhance the phishing detection.
- To determine the best classification algorithm for phishing detection.
- Design a system with integrate multiple classification algorithms for phishing Emails detection and to evaluate such integration

1.4 Motivation

The harmful effects of phishing could be extent to access the users' confidential details, which could result in financial losses for users and even prevent them from access their own accounts. Therefore, in this study, we will quantify and qualify the phishing email features to prevent and mitigate the risk of phishing emails.

In addition, this study will conduct comparative assessment between classifiers data mining algorithms techniques, manual selection feature groups and automated feature selection group. Special focus on the header based feature such as, sub-reply, sub-verify, etc. and content based feature (body) such as body suspension word and dear word etc., long URL addresses, etc. and select those which offer high quality for our study. Moreover, classification and clustering integration will be implemented for the purpose of enhancing the detection accuracy.

1.5 Scope and limitation

The scope of this research is phishing emails detection, where 47 features were selected and categorized in five groups that cover all the email components. Moreover, the LR, DT, One R, SMO and naïve base are five classification data mining algorithms were used for phishing emails detection.

For the limitation, this research will not cover the phishing websites, moreover the experiments will not cover all the available classification algorithms. However, this study will evaluate experimentally the most well-known algorithms.

1.6 Contribution

The thesis goal is to build phishing detection model that uses data mining techniques.

The contribution of this thesis are as follow:

- Selection best sets of features for phishing detection problem manually and automatically.
- Experimentally evaluate the performance of feature sets selected manually and automatically and compare between them.
- Experimentally evaluate the performance of the classification algorithm for fishing detection.
- Propose multiple classification integration system for phishing detection.

1.7 Thesis Organization

The thesis is consists of five chapters organized as the follows:

- **Chapter One:** Introduction: overview of phishing detection techniques, problem statement, the objective of the study, the motivation, the scope and limitation, thesis contribution and finally thesis organization.
- **Chapter Two:** Literature review: this chapter provides an overview of the related works in phishing emails detection and summary of articles that published by other researchers.
- **Chapter Three:** Methodology: this chapter provides an outline of the research methodology which used in this thesis. Overview of the software that used for the evaluation of the proposed method and the dataset were used in this research.
- **Chapter Four:** the implementation details of experiment and the results that were obtained for all the proposed scenarios and comparison of the results.
- **Chapter Five:** Conclusion and future work.

Chapter Two

2.1 Introduction

Detection of phishing emails has received a lot of attention recently due to their impact on users' security. Therefore, many techniques have been developed to detect phishing emails varying from communication-oriented techniques, such as authentication protocols, blacklisting, and white-listing, to content-based filtering techniques (Paaß, 2009). The blacklisting and white-listing techniques have not proven though to be sufficiently efficient when used in different domains, and thus they are not commonly used. Meanwhile, the content-based phishing filters have been widely used and have proven to be of high efficiency. In light of this, researches have focused on content-based mechanism and on developing machine learning and data mining techniques based on the header and body of emails.

In 2007, a study was conducted to measure the efficiency of the existing tools for phishing detection. This study showed that even the best phishing detection toolbars missed over 20% of the phishing websites (Kumaraguru & Rhee & Acquisti, 2007). Another study, which was conducted in 2009 concluded that most anti-phishing tools did not start blocking phishing sites before several hours or days have passed after these phishing emails sent luring users (Parmar, 2012). Therefore, we conclude that the currently implemented detection tools do not detect these phishing email and websites completely (100% percent) (Kumaraguru, 2010).

This Chapter presents different Algorithms for the detection and prediction of phishing Emails.

2.2 Phishing Emails Detection Techniques

A wide range of filters have been developed by specialists to predict and prevent phishing emails and manage occurring threats relying on either traditional techniques such as authentication protection, or on modern techniques of learning machines or mining data.

2.2.1 Traditional Methods

Traditional methods of detection fall into two categories, the network-level protection and the authentication protection. The first category of protection at a network level includes blacklist filters and white-list filters which prevent phishing by blocking suspected IP addresses or domains from accessing the network. In addition, there are the Pattern Matching filters and the Rule-based filters which rely on manually entered and updated fixed rules for detection (Ramanathan, 2012).

- **Blacklist Filter**

The blacklist filtering technique provides protection at a network level by classifying received emails based on the sender's address, IP address or DNS address. These details are extracted from the email's header and compared with a pre-defined list, and if any of these data are matched with the list; the email will be rejected. Therefore, this technique filters phishing emails to provide security at a network level. Internet Server Providers (ISP) is the responsible organization of providing and implementing this filter (Paaß, 2009).

- **Whitelist Filter**

The white-list filtering provides protection at network level as well, but in contrary to blacklists; this technique compares the email's data with a pre-defined list containing

static IP addresses of legitimate domains and IP addresses (Cao, 2008). In this regard, only emails with data matching the list will be allowed to access the network to the user's inbox.

Email addresses and IP addresses are included in the white-list if they belong to legitimate users or companies who have agreed to add their addresses to this list. Emails with data matching to this list will only be classified as legitimate based on this filter, while other emails are considered phishing and prevented from accessing the network for which this filter is called also legitimate emails classifier.

– **Pattern Matching filter**

The pattern matching technique filters emails based on specified patterns, including words, text strings, and character sets mentioned in the email's content, subject, or sender. The filter searches through the email for these specified patterns to classify the email into phishing or legitimate. Although this technique provides protection at a network level, it still provides some invaluable and false results due to the huge number of received emails which may include banned words or text strings but shall not be prevented. (Chhabra, 2005).

The second category, authentication protection, provides security on both user and domain levels. For a user-level protection, users will have to provide authentications before sending their messages such as verified email and password, while the authentication protection on a domain-level is created for emails servers (Ramanathan, 2012).

– **Email Verifications**

Email verification is a user-level authentication method that requires verification from the sender and the receiver. Once the sender accepts the notification message, the email will be certified and classified as legitimate to be passed into the receiver's inbox. Otherwise, the email will be considered as phishing and thus prevented from accessing the inbox (Adida, 2006).

This filter has its pros and cons. Although this filtering process has proven to be efficient in detecting phishing emails completely (100%), it still needs a lot of time relatively as the receiver has to respond before receiving the message, and there is a risk of losing the email if the verification process generated traffic over the network or the same challenge has not been recognized.

– **Password Filter**

Password filters also provide protection through a user-level authentication. Using this filter allows for receiving any email in the subject line, the email address, the header field, or in any part of the email only if the filter was able to detect the determined password. Therefore, if the filter was not able to find the password or detect a wrong password, the email will be rejected. These passwords are not created by default, therefore; first-time users of this filter will have to start a conversation with each other to set and activate a password and then be classified as legitimate by the filter. This type of filters still has its shortcoming in terms that some legitimate emails might be lost if the password was not recognized, in addition that the process requires time (Ramanathan, 2012).

2.2.2 Automated Methods

This method applies automated classifiers that rely on machine learning and data mining. These classifiers work beside the server and filter the received emails into phishing or legitimate by examining different features if the email's header and body (Abu-Nimeh, 2007).

– Logistic Regression

The logistic regression is a widely-used method due to its easily-interpretable and practical results. This model is functional in predicting binary data (0/1 response) as it relies on statistical data and applies a generalized linear model.

Despite of this method's simplicity, it has three shortcomings; first, it requires more statistical assumptions before being applied. Second, it its more functional with variables that have linear relation than those with a complex relation. Last, the accurateness of the predication rate is sensitive to the completeness of the data (Abu-Nimeh, 2007).

– Classification and Regression Trees (CART)

The Classification and Regression Trees (CART) model developed in 80's is used to represent the distribution of Tree that splits using two components, and the T tree that splits into two nodes Decision trees are represented by a set of Yes or No questions which splits the learning sample into smaller and smaller parts.

Unlike logistic regression method, this model is used for complex relations between variables rather than linear relations

A binary tree is created by continuously partitioning the predictor space into different homogenous groups. The partition occurs depending on defined splitting rules associated

to the internal nodes of the tree, where each homogenous group is associated by a terminal node.

This model leads to generating a big binary tree which, although is practical for complex relations and provides easily-read interactions among predictors; it still makes it hard to predict the additive effects due to its huge. (Steinberg & Colla, 2009)

– **Decision Trees Filter (DT)**

Decision Trees Filter is a graphical model of classification that is comprised of nodes and arrows. The base node is called the Root from which the DT is initiated. Each node within the network contains an “If-then” rule, a class, and a feature, and leads to the next one using the arrows, referred to as edges. The decision tree ends with a leaf node called the terminator. The tree could include one or more classifier stages and the internal nodes are bounded by the root and terminating nodes (Safavian, 1990).

Different algorithms have been suggested to generate decision trees including the ID3 model which calculates information of entropy as a heuristic function to evaluate the target. In 1992, this algorithm was developed to C4.5 algorithm.

In that sense, the decision tree will generate sub-trees, each node in the tree has a parent node leading to it (except for the root), and each one also leads to a child node (except for the terminating node), while the tree will end with the terminating node (leaf node) that represents the final solution of the suggested problem.

– **Support Vector Machine (SVM)**

SVM is widely applied by researchers in the medical diagnoses, text categorization, image classification, bio sequences analysis, and other fields. Using this technique, data

is divided into two categories using statistics, Quadratic equations, and fixed rules. The binary classification of the data is created by using a separating hyper plane to maximize the space of the margin base on kernel functions, and extracting data and storing it in the vector, to reach the best solution of the problem and finding the suitable classification. This technique is beneficial for finding solutions of problems with unfamiliar history, but fails at analyzing big data.

In (Figure 2.1) the support vectors are illustrated on the boundaries, and the separating hyperplane is located in the middle of the margin maximize the separation margin.

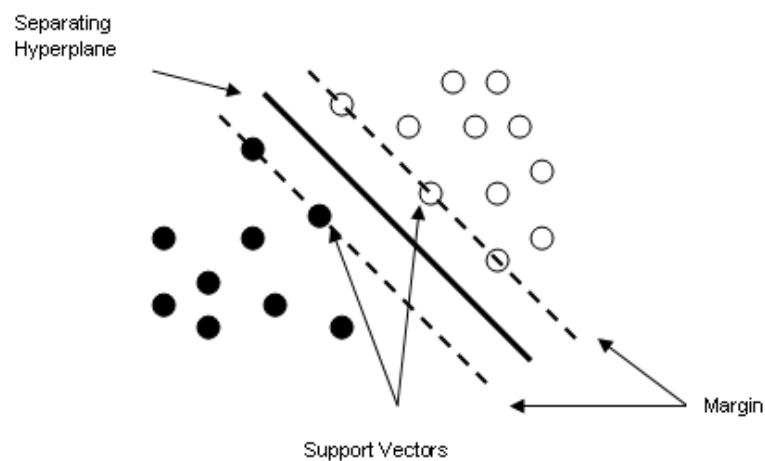


Figure (2.1) Support Vector Machine (Al-Momani and Gupta 2013).

Table 2.1 summarizes the well-known phishing detection tools such as CloudMark, Netcraft, FirePhish, eBay Account Guard and IE Phishing Filter (Ramanathan, 2012). The authors pointed out the main disadvantages of the popular tools that are widely used.

Table (2.1) Phishing detection tools

Tool	Type	Description	Advantage	Disadvantage
Snort	Network level	Heuristic tool	Good at detecting level attacks	Rules require manual adjustments. Does not look at content
Spam Assassin	Server Side Filter	Heuristic engine uses specific features	Good at detecting email header spoofing	High false positives
PILFER	Server Side Filter	Utilize 10 features	Better performance than spam assassin	Did not use content from the body of the email. Used with short lived phish domains.
Spoof Guard	Client Side Tool	Plug-in to a browser	Warns user if link points to phishing site	Users do not pay attention to warnings. Not all email clients are browser based.
Calling ID, Cloud Mark, Netcraft, and Fire Phish	Client Side Tool	Utilizes blacklist of domains	Good for domains that employ domain level authentication	Phish domains are short lived. Does not look at email content.
eBay Account Guard	Client Side Tool	Utilizes blacklist of eBay URLs	Protects eBay users.	Specific website tool.
IE Phishing Filter	Client Side Tool	Records specific user website visiting patterns.	Adapts to user website visit pattern	Works only on internet explorer.
Catching Phish	Client Side Tool	Detects fake website based on rendered images	Browser independent. Good results on small data sets.	Processing time is high. Susceptible to screen resolution

2.3 Literature Review

This section provides an overview on some of the main studies conducted on data mining techniques and algorithms to detect phishing emails:

Chandrasekaran depended on the distinctive structural features of the email to detect phishing emails. These features work in cooperation with the SVM to predict phishing emails and prevent them from originally reaching the user. (Chandrasekaran & Narayana & Upadhyaya, 2006)

In 2007, Abu-Nimeh focused on examining different machine learning methods and comparing the accuracy of their predictions using a total of 2889 phishing and legitimate emails. All of the following methods were included in the study: Logistic Regression (LR), Classification and Regression Trees (CART), Bayesian Additive Regression Trees (BART), Support Vector Machines (SVM), Random Forests (RF), and Neural Networks (NNet) which were tested also using 43 features. According to the results, RF outperforms all other classifiers, with an error rate of 07.72%, followed by CART, LR, BART, SVM, and NNet respectively, providing that the legitimate and phishing emails are given equal weights. In terms of false positive rate, the best results were achieved by the LR with a percentage of 4.89%, followed by BART, NNet, CART, and SVM respectively, while the worst false positive rate was achieved by the RF with percentage of 08.29. (Abu-Nimeh & Nappa & Wang 2007)

Furthermore, the two methods of adaptive Dynamic Markov Chains (DMC) and latent Class-Topic Models (CLTOM) were proposed by Bergholz to classify emails where two new features were produced. The adaptive version of the DMC succeeded to provide the same quality performance in comparison to the standard version while using two-thirds less of the memory. As for the CLTOM, the adaptive version has shown higher performance than the standard LDA as the first incorporates class-specific information into the topic model and has achieved a total of topic numbers of up to 100.(Bergholz & Chang & Paass, 2008)

Toolan developed a new C5.0 algorithm to filter into Phishing / non-Phishing categories by selecting 5 features. The sampled data included 8,000 emails where half of them were phishing and the other half was legitimate. This approach outperformed any

other individual classifier or collection of classifiers in terms that is achieved higher recall efficiency.(Toolan & Carthy, 2009)

Abu-Nimeh and others founded a detection tool for protecting mobile platforms against attacks. The client-server distributed server relies on Additive Regression Trees beside the server with the assistance of the automatic variable selection to improve their predictive accuracy and eliminate the overhead of variable selection is applied.(Abu-Nimeh & Nappa & Wang, 2009)

Gansterer proposed a filtering system that classifies received emails into three categories; legitimate (solicited e-mail), spam, and phishing emails, relying on newly developed features from these emails. The system comprises different classifiers to be able to categorize received messages. A classification accuracy of 97% was achieved among the three groups, which is considered better than solving the ternary classification problem by a sequence of two binary classifiers.(Gansterer & Polz, 2009)

Dr. Ma used an algorithm with a set of orthographic features to cluster phishing emails automatically and eliminating redundant features. This clustering and feature selection technique succeeded in providing highly efficient results. Ma applied the global k-mean model with a little modification and generated the values of the objective function over a range of tolerance values of selected features subsets. The objective function values assisted in recognizing the suitable clusters based on the distribution of these values. (Ma & Yearwood & Watter 2009)

Basnet studies a detection approach that utilizes readily acquired features from the email's content without resorting to heuristic-based phishing features. This approach relied on Confidence-Weighted Linear Classifiers proposed by Basnet. Images are generated by Phishers from the message's text that only graphical data passes the phishing filter. (Basnet & Sung, 2010)

Dr. Wu focused on spoofing emails and Microsoft Outlook™ services by developing a sender authentication protocol (SAP). This authentication protocol verifies the authenticity of the sender by testing the claimed-sender1 with the archived emails. The enhanced Outlook™ has an add-in that tests feasibility while it remained the same user-friendly interface of the original version, and this the SAP add-in will be started automatically once the Outlook™ operates. (Wu & Zhao & Qiu, 2010)

In 2011 Khonji and Jones and Iraqi they listed the 47 features for the Email that were used to classify the phishing emails in the study and they gave a brief description on each feature, the list covers all the structures of the Email. (Khonji & Iraqi, 2011)

A new genetic algorithm was developed by Alguliev for clustering spam messages and solving clustering problems. The proposed algorithm uses the strategy of maximizing the similarity between messages in clusters, and the objective function is defined by k-nearest neighbor algorithm. However, such algorithms are limited by the constant support of chromosomes which reduces convergence process when trying to solve constrained problems. Therefore, a penalty function is applied to expedite the convergence process and preventing infeasible chromosomes

Thereafter, a detailed examination is conducted on the resulting classification to conclude information about the classes, and an informative portrait is shaped through documentation to achieve better understanding of these clusters and spam messages. This anti-spam system will help in predicting targeting information attacks, in addition to analyzing the origins of spam messages which will help in finding organized social networks of spammers. (Alguliev & Nazirova, 2011)

Azad has focused on testing different existing algorithms in terms of their accuracy, such as Naive Bayes, logistic regression, and support vector machine (SVM) classifiers. He used bag of words and augmented bag of words models. In general, the tested classifiers achieved high results indicating an accuracy rate of 95% with the SVM with the linear kernel and Bayes topping the other classifiers, as they only missed 10 and 2.66 percent of phishing emails respectively. When in comparison with the Naive Bayes and logistic regression, the SVM showed equal results being tested with less features. Meanwhile, the linear SVM was tested as well with removing additional features to result in lower detection rates as it misclassified 5.86 percent of phishing emails, meaning that additional features enhance the accuracy of the results. In conclusion, the study showed that linear SVM is beneficial for detecting phishing emails before they even reach the user's inbox. (Azad, 2011)

A new method for clustering of spam messages collected in antispam system is offered by Alguliev, through the development of Genetic algorithm including penalty function for solving clustering problem. In addition to, the classification of new spam messages coming to the bases of antispam system. The proposed system is not only

capable to detect purposeful information attacks but also to analyze origins of the spam messages from collection, it is possible to define and solve the organized social networks of spammers (Alguliev & Nazirova, 2011).

Meanwhile, a new version of neural networks was developed by Al-Momani that achieved a zero-day detection of unknown phishing emails. The new framework was named PENFF (Phishing Evolving Neural Fuzzy Framework) which relies on adaptive evolving fuzzy neural network (EFNN). As a performance indicator; the Root Mean Square Error (RMSE) and Non-Dimensional Error Index (NDEI) are 0.12 and 0.21 respectively which indicate low error rates compared to other approaches. (Al-Momani & Altaher 2012)

Kumar used TANAGRA data mining tool on a sampled spam dataset to evaluate the efficiency of the emails classifier where several algorithms were applied on that data set.

At the end, the features selections by Fisher spam filters and Rnd filtering achieved better classifications. After fisher filtering has acheived more than 99% accuracy in detecting spam, The Rnd tree classification algorithm was applied on relevant features. (Kumar, Poonkuzhali, Sudhakar, 2012)

Altaher relied on Adoptive Evolving Fuzzy Neural Network (EFuNN) to create Phishing Evolving Neural Fuzzy Framework (PENFF) to detect of unknown “zero-day” phishing emails by handling all similar feature vectors to establish rules for prediction. Therefore, PENFF approach relies on the similarity of features included in the email’s body and URL.(Altaher & Al-Momani & Wang, 2012)

Pandey classified phishing emails by applying several methods, such as; Multilayer Perceptron (MLP), Decision Trees (DT), Support Vector Machine (SVM), Group Method of Data Handling (GMDH), Probabilistic Neural Net (PNN), Genetic Programming (GP) and Logistic Regression (LR). This combination aimed at using text and data mining in parallel for detection where 23 keywords were extracted from the email body and were already included sampled data set, in addition to a total 2500 phishing and non-phishing emails were analyzed

A t-statistic based feature selection was applied to conclude the 12 of the most effective features in predicting phishing emails with accuracy and minimum feature number. The study compared results of processes with features selection and without them. As a result, the selection of features showed no effect on the classifiers and on the detection process. This result was reasonable as the GP and DT do not differ statistically in a distinctive manner, either with or without feature selection. The DT however applies the “if-then” rule which acts like an early warning expert system; therefore, this system shall be preferred and widely used. .(Pandey & Ravi, 2012)

Jameel and George used a feedforward neural network to identify the phishing email by extracting features from the email's header and HTML body. Their suggested algorithm was tested on 18 features using 5 hidden neurons. For this algorithm, a training is required before implementing it which takes 173.55 msec. The time for testing a single email is 0.00069 msec. The consumed time will increase with the increase in the neurons number while it is still considered low. With regard to the results, the algorithm proved high accuracy of 98.72%, and a learning rate of 0.01. (Jameel & Loay and George 2013)

Zhang aimed at estimating the accuracy of the cross validation approach in detecting phishing emails. He used multilayer feedforward neural networks (NN) systems with different numbers of hidden units and activation functions to prove that NNs can provide fairly accurate and efficient results with an estimated number of hidden units. It is worth mentioning that he proved these results even with few training while selecting the features set will achieve better results (Zhang & Yuan, 2013)

In 2013, Al Momani found a new model that proved excellent results in terms of true positive, true negative, sensitivity, precision, F-measure and overall accuracy compared with other approaches. In addition, the system showed efficiency in predicting the values of these emails in online mode, and long-life working with footprint consuming memory. The model Al Momani developed is called Phishing Dynamic Evolving Neural Fuzzy Framework (PDENF) for predicting unknown phishing emails and detecting them in zero day (Almomani & Gupta & Wan, 2013)

Regarding websites classification, Khonji examined the modified technique for preventing phishing emails and enhancing the filters efficiency. The previously proposed technique relied on analyzing the website's URLs lexically which enhanced the accuracy of the filters by 97%. Lexical URL analysis indicated higher accuracy of anti-phishing classification. (Khonji & Jones & Iraqi, 2013)

Later in 2013, Rathi aimed at comparing the performance between algorithms with a feature selection and algorithms without a feature selection. At first, the sampled data was examined without any filters or features selection, then the classifiers were tested

each at time beginning with the best-first feature selection to be able to elect the most beneficial features and then apply various classifiers for classification

The Random Tree classifier proved a 99.72% accuracy which means it works best to detect spam emails. In conclusion, the accuracy of email filters was enhanced incredibly when the algorithm with feature selection was applied into the entire process and that classifiers of tree shape are more efficient in detecting spam emails (Rathi & Pareek, 2013)

Another framework was found by Al Momani and others that also detects unknown zero-day phishing emails relying on a the “evolving connectionist system”. The new system was named the phishing dynamic evolving neural fuzzy framework (PDENFF) and follows a hybrid learning approach (supervised/ unsupervised) and is supported by an offline learning feature to achieve the intended purpose. Using this system helped in enhancing the detection of zero-day phishing e-mails was improved between 3% and 13%. Moreover, it used rules, classes or features to enhance the learning process using ECOS which provided the system with the advantage of distinguishing phishing emails from legitimate one. (Al-Momani & Gupta & Wan, 2013)

Another mechanism was developed later in 2014 by Akinyelu to better classify phishing emails using forest machine learning mechanism. This mechanism was tested on data comprising around 2000 phishing emails with advanced features (as identified from the literature), and it was able to classify phishing emails with high efficiency (99.7%) with low false negative (FN) and false positive (FP) rates. Therefore, Akinyelu’s algorithm is more efficient in terms that it requires fewer features to detect phishing and provides more accurate results. (Akinyelu & Adewumi, 2014)

A fraudulent detection model was proposed by Nizamani (2014) using an advanced selection of features where the different categories were compared in terms of the fraudulent email detection rate. The study was conducted applying several classification approaches and algorithms, such as SVM, NB, J48 and CCM, in addition to different features sets. An accuracy percentage of 96% was achieved and the results indicated that the level of accuracy was affected by the type of selected features rather than the classifiers' type (Nizamani & Memon & Glasdam, 2014)

In 2015, Kathirvalavakumar and others proposed a multilayer neural network to detect phishing emails. His suggested network relies on a feedforward pruning algorithm that extracts distinguished data and features from the email and applies a weight trimming strategy. This pruning strategy helps in minimizing the number of features through the algorithm resulting in minimum computation required for classification of emails into phishing or not. The network has provided fair results in terms of false positives and false negatives. As this network has been tested on data from 2007, using this network for current data requires identifying the new features to the algorithm incorporating them into input domain for training in order to be useful. (Kathirvalavakumar & Kavitha & Palaniappan, 2015)

Chapter Three

3.1 Introduction

This Chapter presents the proposed work for phishing detecting. The work contribute to the field by developing a multi-classifier integration model by combining clustering and classifications techniques to enhance the detection accuracy, Moreover, a comparative assessment between various classification algorithms, and feature selection scenarios (manual and automated feature selection groups) are proposed .

3.2 Proposed approach

Initially, a comprehensive literature review on the features that were used in phishing emails detection as well as the data mining techniques were presented in Chapter two. As such, the proposal work starts by investigate the phishing email detection accuracy with a complete feature set. Subsequently, reduce the time and space required for extracting and using these features. Then, the performance of different classification algorithms on the extracted feature set is investigated. Finally, a multiple approach for combining multiple classifiers is proposed. The steps of the proposed work is presented in Figure 3.1.

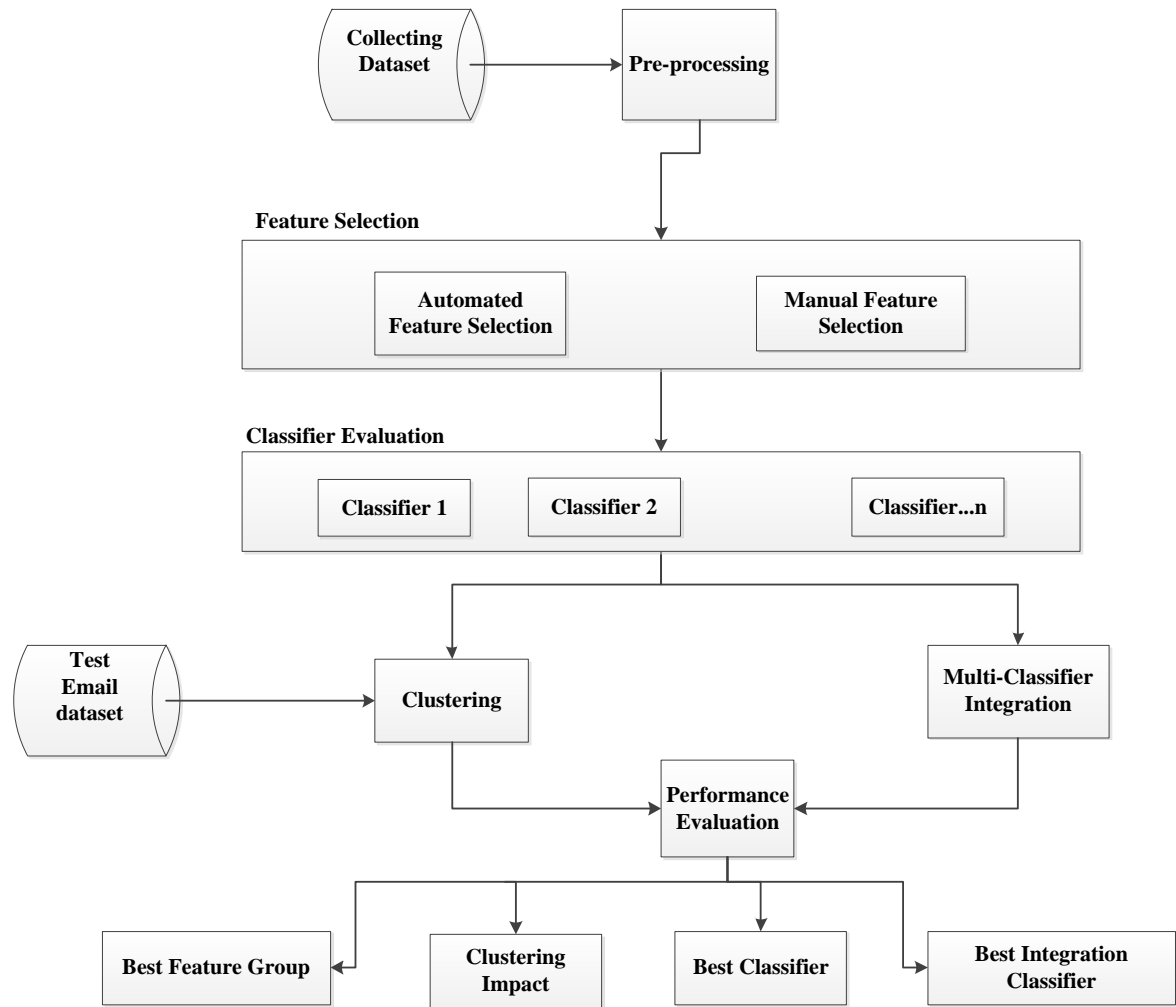


Figure (3.1) Architectural Design of the Proposed Approach

3.2.1 Pre-processing:

In the pre-processing step, phishing emails dataset was collected. Features are then extracted from each email and all the features for all emails are presented by a matrix, each row represent one email along with columns corresponding to 47 selected features, in addition to a column that represents the class of the email (whether it is phishing or legitimate email) as shown in Figure 3.2 and 3.3. The set of features were categorized into four groups; email Body group (contains 11 features), Email Header Group (contains 11 features), URL features group (contains 18 features) and java script

features was combined with the external features as one group (contains 7 features). The 47 features are listed in Table 3.1, 3.2, 3.3 and 3.4 respectively.

Table (3.1) The Selected Body Features

No .	Feature Name	Description
1.	Body dear word	A binary feature that returns 1 if the word “dear” was found in the body of a message, and 0 otherwise.
2.	Body form	A binary feature that returns 1 if the email message contains a html form, and 0 otherwise.
3.	Body html	A binary feature that returns 1 if the email message has html content, and 0 otherwise.
4.	Body multipart	A binary feature that returns 1 if the email message has a multipart MIME type and 0 otherwise.
5.	Body no. characters	A continuous feature that returns total number of characters found in the body of a given email.
6.	Body no. words	A continuous feature that returns total number of words found in the body of a given email.
7.	Body no. unique words	A continuous feature that returns total number of unique words found in the body a given email message.
8.	Body richness	A continuous feature that returns the result of dividing total number of words by total number of characters found in the body of a given email.
9.	body no. function words	<div><div>A continuous feature that returns total number of function words found in the body of a given email. Function words are:</div><div><div><ul style="list-style-type: none">• Account• Access• Bank• Credit• Click• Identity• Inconvenience• Information• Limited</div><div><ul style="list-style-type: none">• Log• Minutes• Password• Recently• Risk• Social• Security• Service• Suspended</div></div></div>
10.	Body suspension word	A binary feature that returns 1 if the word “suspension” is found in the body of an email, and 0 otherwise.
11.	Body verify your account phrase	A binary feature that returns 1 if the phrase “verify your account” is found in the body of an email, and 0 otherwise.

Table (3.2) The Selected Email Header Features

No.	Feature Name	Feature Description
1.	Subject bank word	A binary feature that returns 1 if the “bank” word is found in the subject field of a given email message, and 0 otherwise.
2.	Subject debit word	A binary feature that returns 1 if the “debit” word is found in the subject field of a given email message, and 0 otherwise.
3.	Subject fwd word	A binary feature that returns 1 if the “Fwd:” word is found in the subject field of a given email message, and 0 otherwise.
4.	Subject reply word	A binary feature that returns 1 if the “Re:” word is found in the subject field of a given email message, and 0 otherwise.
5.	Subject verify word	A binary feature that returns 1 if the “verify” word is found in the subject field of a given email message, and 0 otherwise.
6.	Subject no. characters	A continuous feature that returns the total number of characters found in the subject field of a given email.
7.	Subject no. words	A continuous feature that returns the total number of words found in the subject field of a given email.
8.	Subject richness	A continuous feature that returns the result of dividing total number of words by total number of characters found in the subject field of a given email.
9.	Send no. words	A continuous feature that returns the total number of words found in the “sender” field of a given email.
10.	Send different reply to	A binary feature that returns 1 in case a difference between the sender and reply-to email addresses was found, and 0 otherwise.
11.	Send unmodal domain	A binary feature that returns 1 in case the sender email address uses an unmodal domain name, and 0 otherwise. A modal domain name is defined as the most frequently referred to domain name in the body of a given email.

Table (3.3) The Selected URL Features

No.	Feature Name	Feature description
1.	url at char	If the Email contain a URL with "@" Returns 1, else 0
2.	url bag link	If the following words found in the email returns 1 and 0 otherwise (Click, Here, Login, Update)
3.	Url IP	If the Email contain a URL with IP, address in its authority portion returns 1 and 0 otherwise.
4.	url no. domains	A continuous feature that returns total number of domains found in URLs in a given email.
5.	url no. external link	A continuous feature that returns total number of external links found in a given email. An external link is a link that points to a resource that is accessible out of the email.
6.	url no. internal link	A continuous feature that returns total number of internal links found in a given email. An internal link is a link that points to a resource that is accessible in the email
7.	url no. image link	Returns total number of image links found in a given email.
8.	urlnumip	A continuous feature that returns total number of URLs that contain an IP address in their authority section as opposed to a domain name.
9.	url no. link	A continuous feature that returns total number of links found in the body of a given email.
10.	url no. periods	A continuous feature that returns total number of periods in the body of a given email.
11.	url no. port	A continuous feature that returns total number of URLs with port numbers in their authority section in a given email.
12.	url port	A binary feature that returns 1 if a URL with a port number is found in the body of a given email, and 0 otherwise.
13.	url two domains	A binary feature that returns 1 if a URL is found that has two domain names and 0 otherwise.
14.	url unmodal bag link	A binary feature that return 1 if an unmodal link is founded with certain words (Click, Link, Here) in its link text, and 0 otherwise. The particular words are:
15.	url word click link	If found word "click" in the link text returns 1, 0 otherwise
16.	url word here link	If found word "here" in the link text returns 1, 0 otherwise
17.	url word login link	If found word "login" in the link text returns 1, 0 otherwise.
18.	word update link	If found word "update" in the link text returns 1, 0 otherwise

Table (3.4) The Selected Java Script and External Features

Java Script Feature		
No.	Feature Name	Feature Description
1.	Script java script	A binary feature that returns 1 if the body of a given email message contained JavaScript, and 0 otherwise.
2.	Script on click	A binary feature that returns 1 if an “on Click” JavaScript event was found in the body of a given email, and 0 otherwise
3.	Script popup	A binary feature that returns 1 if a given email message contained JavaScript code to open pop-up windows, and 0 otherwise.
4.	Script status change	A binary feature that returns 1 if a given email message contained JavaScript code to modify the status bar, and 0 otherwise.
5.	Script unmodal load	A binary feature that returns 1 if a given email message contained JavaScript that is loaded from an external website which is not a modal domain name, and 0 otherwise.
External Feature:		
6.	Externals a binary	A binary feature that returns 1 if a given email is labeled as a phishing message by Spam Assassin and 0 otherwise.
7.	Externals a score	A continuous feature that returns the score of a given email as returned by Spam Assassin

3.2.2 Features Selection

In feature selection, as illustrated in Figure 3.2, two main scenarios were developed, the first is manual feature selection and the second is automated features selection.

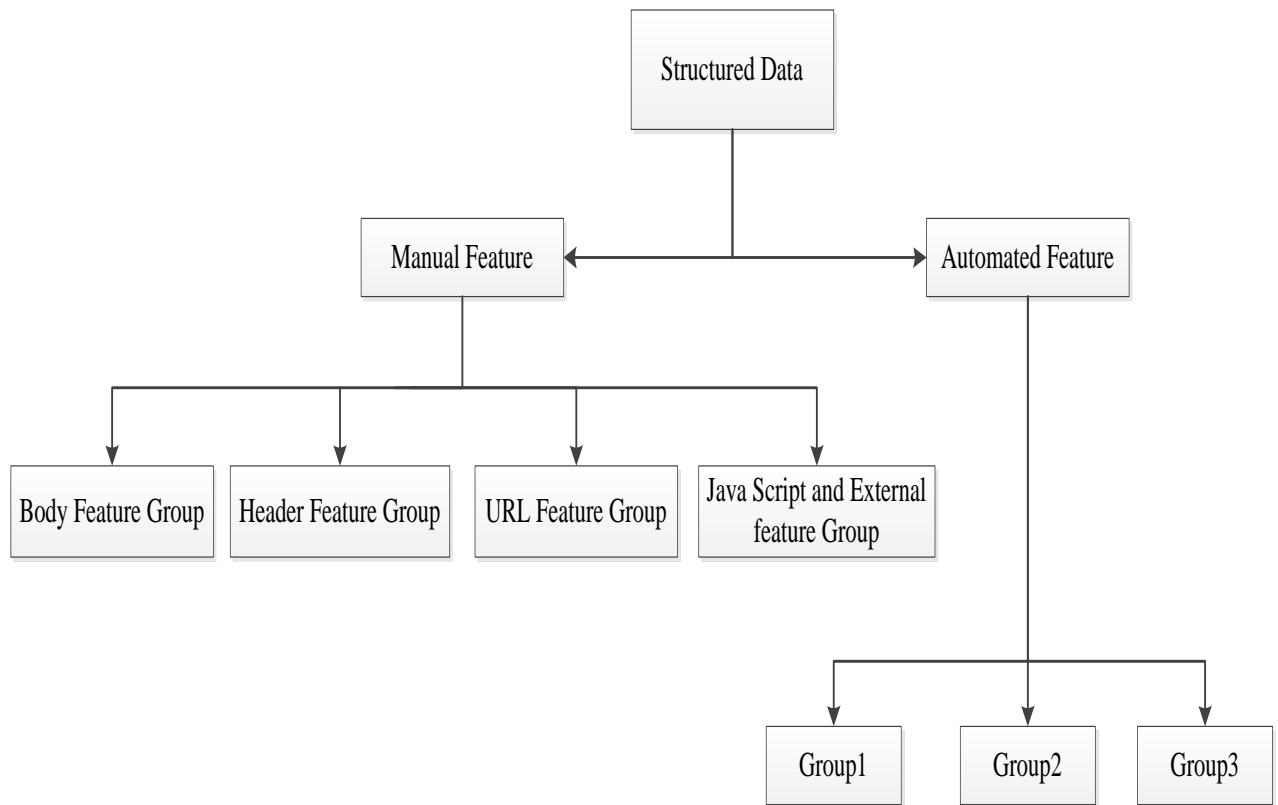


Figure (3.2) Manual and Automated Feature Groups

- 1) Manual Feature Selection Scenario: as mentioned above the selection of four groups were based on the content and the structure of the email. Accordingly, the manual feature selection scenario was based on the feature groups. This generates a groups as given in Table 3.5.

Table (3.5) The Groups of Manual Features Selection

The Features Group	Number of Features
All the features	47
Only the Email body features	11
Only Email Header Features	11
Only URL features	18
Only Java Script and External features	7
All feature excluding Email body features	36
All feature excluding Email Header Features	36
All feature excluding URL features group	30
All feature excluding Java Script and External features	40

2) Automated Feature Selection Scenario: using feature selection algorithms generate three groups. The features selection algorithms that are used are :

- Correlation-based feature Subset evaluator, which evaluates the worth of a subset of attributes by considering the individual predictive ability of each feature along with the degree of redundancy between them.
- Consistency Subset Evaluator, which evaluates the worth of a subset of attributes by the level of consistency in the class values when the training instances are projected onto the subset of attributes.
- Principle component which Performs a principal components analysis and transformation of the data

The Table 3.6 shows the automated selected features with the algorithms that used.

Table (3.6) The Groups of Automated Features Selection

Selection technique	Search method	Number of Features
Correlation-based feature Subset	Greedy Stepwise	14
Consistency Subset evaluate	Greedy Stepwise	3
Principle component	Ranker	33

3.2.3 Algorithms Evaluation

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

- Naive base
- Decision tree (J48)
- Logistic regression
- Classification and regression tree (CART)
- Sequential minimal optimization (SMO)

3.2.4 Feature Clustering

The first step after selecting the features and the dataset is define a groups and put objects in them this is called Clustering and it is similar to classification but in an unsupervised way. while in classification objects are assigned into predefined classes that makes significant group of objects share similar characteristics(Al-Momani et al., 2011). For further explanation about clustering here is the most popular example about clustering the Library, in the library the books have a wide range of topics. The challenge

is how to gather those books in a way that readers can take several books in a specific topic without extensive search or effort. Clustering introduces some kind of similarities in one cluster or one shelf and arranges it with a meaningful class. According to that, readers just go to that shelf instead of looking in the whole library.

In this experiment the K-means algorithm used to make the clustering and to categorized all the Emails into five groups (0, 1, 2, 3, and 4) as shown in Figure 3.3

body Dear Word	url word login link	Url word Update link	Kmean Cluster
0	0		0	0
1	0		0	2
0	0		0	4
1	1		0	2
0	0		0	4
0	0		0	3
0	0		0	3
0	0		0	2

Figure (3.3) Sample of the Dataset with K-means Clustering

3.2.5 Multi-Classification Integration Approach for Phishing Email Detection

Three algorithms been used to build the Multi- classifier model Logistic regression, Decision Tree and Sequential minimal optimization, the first two algorithms will test the email synchronization whether it's fishing or legitimate then analyze the result if the labels are equal it will be assigned otherwise it will be tested by the 3rd algorithm to decide and label the email as shown in Figure 3.4.

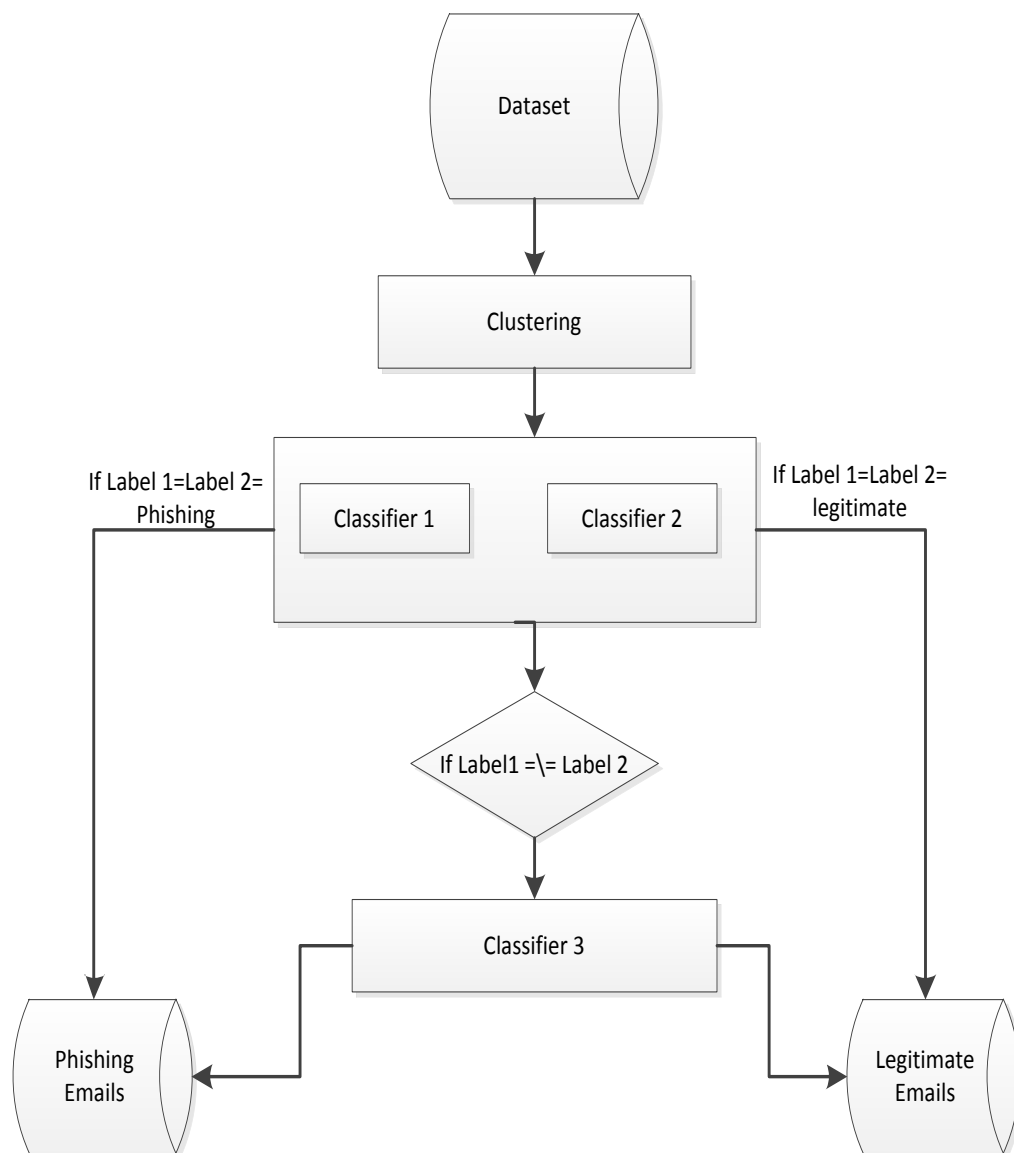


Figure (3.4) Multi-Classifier Integration Model

Chapter Four

In this chapter, the experiment will be presented along with the result and the evaluation, divided into sections as the follows:

- Section 4.1: presented the dataset which is a group of 4800 emails (phishing and legitimate),
- Section 4.2: describes the utilized tools the WEKA tool is used test the datasets with the built-in machine-learning algorithm and 4.3 section presents the experiment results.

4.1 Data Set

The utilized date set contains 4800 emails, 2400 phishing emails and 2400 legitimate emails. The emails are obtained from two sources, firstly, is the monkey website for phishing emails (Monkey, 2016), While, the legitimate emails were collected from the spam Assassin website for the data mining competition (Apacheorg, 2016).

The spam Assassin resource offers, legitimate emails that contains two categories: easy legitimate emails and hard legitimate emails which are very close to spam then the whole.

Feature extraction is implemented in the data set representation, were each email is converted into feature vector of 47 selected features and a column which represent the type of the email (whether it is phishing or legitimate email) as shown in Figure 4.1 and 4.2

A	B	C	D	E	F	G
Email Number	body Dear Word	Body Form	Body HTML	Body Multipart	Body NumberChart	Body Num Function Words
1	0	0	0	0	4522	0
2	0	0	0	0	890	1
...	0	0	0	0	3931	12
	0	0	1	0	2995	19
4801	1	0	1	0	1382	15

H	I	J	K	L
Body Num Uniq Words	Body Num words	Body Richness	Body Suspension Word	Body verify your account phrase
374	931	0.205882353	0	0
124	198	0.22247191	0	0
499	864	0.219791402	0	0
195	642	0.214357262	1	0
164	327	0.236613603	0	0

M	N	O	P	Q	R
Externals a Binary	Externals a Score	Script Java Script	Script on Click	Script Popup	Script Status Change
0	0	0	0	0	0
0	0	0	0	0	0
0	3.5	0	0	0	0
0	2.7	1	1	1	1
1	15.3	0	0	0	0

S	T	U	V	W	X
Script Unmodal load	Send Diff Reply To	Send NumWords	Send Unmodal Domain	Subject Bank Word	Subject Debit Word
0	1	4	0	0	0
0	1	4	0	0	0
0	1	0	0	0	0
0	1	0	0	0	0
0	1	0	0	0	0

Y	Z	AA	AB	AC	AD
Subject fwd Word	Subject Num Chars	Subject Num words	Subject Reply word	Subject Richness	Subject Verify word
0	21	4	1	0.19047619	0
0	21	4	1	0.19047619	0
0	37	6	0	0.162162162	0
0	21	3	0	0.142857143	0
0	51	7	0	0.137254902	0

Figure (4.1) Sample of the Dataset 47 Feature

AE	AF	AG	AH	AI	AJ	AK
Url at Char	url Bag link	url IP	url num Domains	url num External link	url num Imagelink	url num Internal link
0	0	0	2	0	0	0
0	0	0	2	0	0	0
0	0	0	2	0	0	0
0	1	0	2	2	0	0
0	1	0	4	1	0	0

AL	AM	AN	AO	AP	AQ	AR
Url num IP	Url Num Link	Url Num Periods	url Num Port	url Port	url Two Domains	url unmodal baglink
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	1	0
0	2	1	0	0	0	0
0	1	3	0	0	0	0

AS	AT	AU	AV	AW
url word click link	url word here link	url word login link	Url word Update link	Class
0	0	0	0	ham
0	0	0	0	ham
0	0	0	0	ham
0	0	1	0	phish
0	0	1	0	phish

Figure (4.2) Sample of the Dataset 47 Feature

4.2 Tools

The file was converted to CSV format in order to be compatible to be tested with the selected five algorithms through WEKA tool.

Weka is a group of machine learning algorithms used for data mining tasks, and it's contain several tools for data pre-processing, regression, classification, association rules, visualization and clustering. The algorithms can either be implemented on the dataset directly or called from a Java code. It is also quite suitable for developing new machine learning schemes. (Weka, 2016)

4.3 Experimental Results

Several experiment are implemented in different scenarios, the experiment and the result were evaluated using several measurements, the performance of several experiments were compared and the results were highlighted

4.3.1 Evaluation Measures

Accuracy is the rate of correct predictions that the model achieving when compared with the actual classifications in the dataset. On the other hand, Precision and recall are two evaluation techniques, which calculated based on confusion matrix as shown in Table 4.1 and computed according to Equations 4.1, 4.2 and 4.3:

$$Precision = \frac{TP}{TP + FP} \quad (4.1)$$

$$Recall = \frac{TP}{TP + FN} \quad (4.2)$$

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (4.3)$$

Where,

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

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

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

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

Table (4.1) Confusion Matrix

	Classified Phishing	Classified legitimate
Actual Phishing	TP	FN
Actual Legitimate	FP	TN

4.3.2 Experimental Results on Features Selection

In this experiment the accuracy, precision and recall were calculated for the both scenarios. This will comparably evaluate the manual feature selection and the automated feature selection. Moreover, it will comparably evaluate the influence of each diagnostic selected feature group. Finally, average results were calculated and compared with the results of the test on all features in order to recommend new filtering approach for phishing detection.

Initially, the experiment is carried out on entire features set the results is summarized in Table 4.2 and plotted in Figure 4.3

The results show that the DT and SMO classifiers algorithm achieved the highest accuracy, precision and recall as of 98. While, the NaiveBayes classifier algorithm obtained the worst result as of 97.37.

Table (4.2) The Result of Test on All Features Together

Algorithm	Accuracy	Precision	Recall
LR	97.75	0.97	0.97
DT , J48	98	0.98	0.98
CART, One R	97	0.97	0.97
SMO	98	0.98	0.98
NaiveBayes	97.37	0.974	0.974

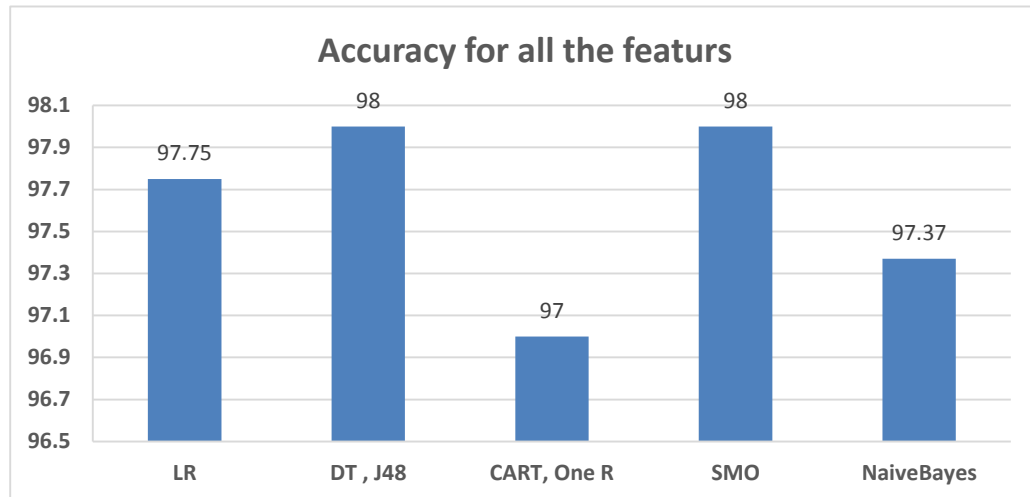


Figure (4.3) Accuracy of the Five Algorithms on All Features Together Test

Then the experiment was carried out on the body feature only, the results is summarized in Table 4.3 and plotted in Figure 4.4.

The results show the NaiveBayes classifier algorithm achieved the highest accuracy, precision and recall as of 96.75. While, the LR classifier algorithm was the lowest result as 95.62.

Table (4.3) The Result of Test on the Body Feature Only

Algorithm	Accuracy	Precision	Recall
LR	95.62	0.958	0.956
DT , J48	96.50	0.965	0.965
CART, One R	95.75	0.959	0.959
SMO	96.62	0.967	0.966
NaiveBayes	96.75	0.968	0.968

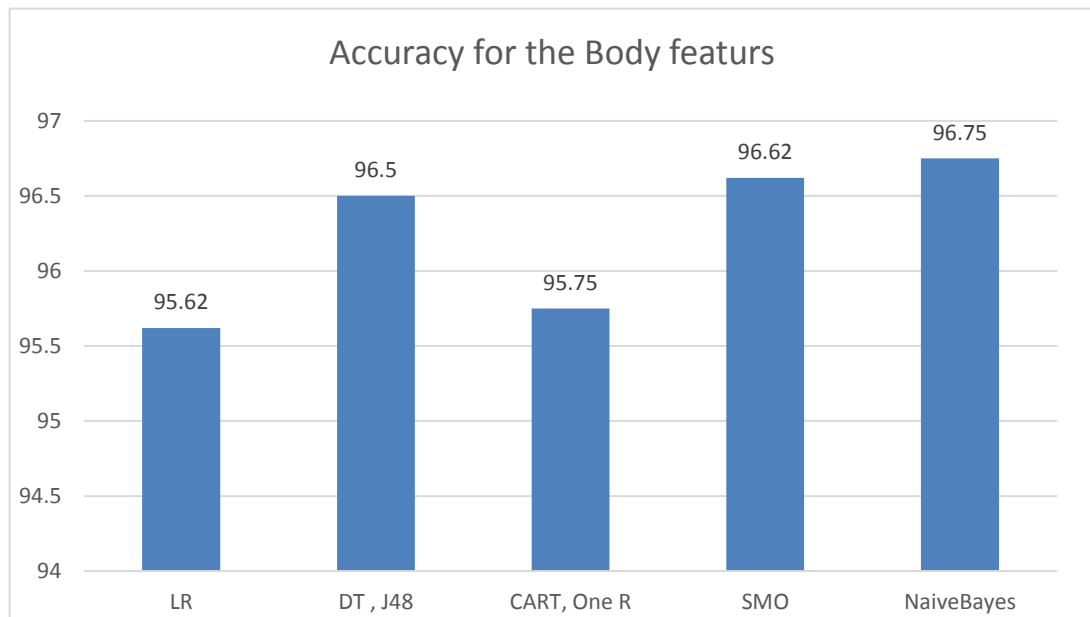


Figure (4.4) Accuracy of the Five Algorithms on Body Feature Only Test.

The results are close far from there obtained for the entire set. Thus, body feature is not a good representative of phishing detection task.

Then, the experiment was carried out on the URL feature only. The results are summarized in Table no. 4.4 and plotted in Figure 4.5.

The results show the DT classifier algorithm achieved the highest accuracy, precision and recall as of 95.65. While, the SMO classifier algorithm was the lowest result as 90.7.

Table (4.4): The Result of the Test on URL Feature Only

Algorithm	Accuracy	Precision	Recall
LR	93.62	.94	.93
DT , J48	95.65	.957	.957
CART, One R	93.7	.90	.98
SMO	90.7	.916	.90
NaiveBayes	91	.916	.91

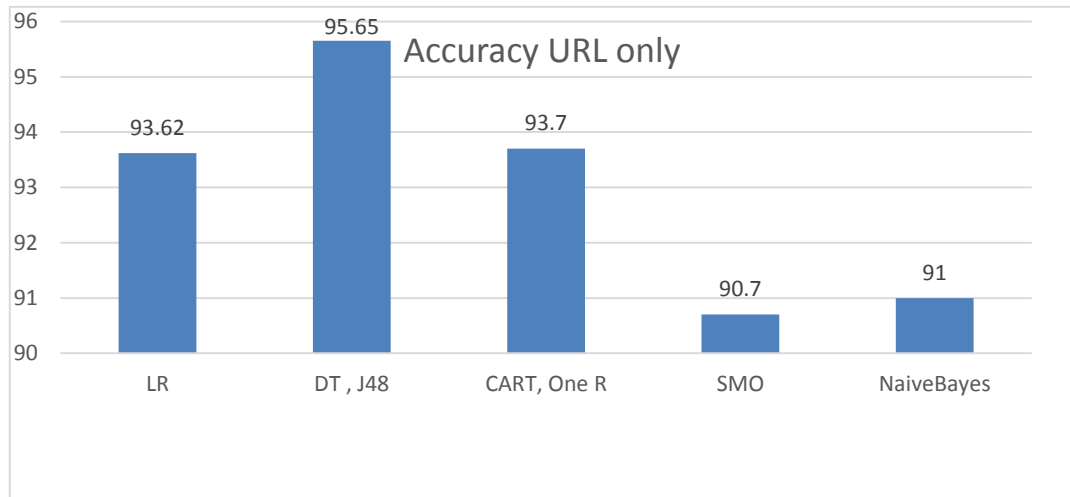


Figure (4.5) Accuracy of the Five Algorithms on URL Feature Only Test

The results are close far from there obtained for the entire set. Thus, URL feature is not a good representative of phishing detection task.

Then, the experiment was carried out on the Header feature only; the results are summarized in Table 4.5 and plotted in Figure 4.6.

The results show the DT classifier algorithm achieved the highest accuracy, precision and recall as of 92.3. While, the CART classifier algorithm was the lowest result as 91.6.

Table (4.5) The Result of the Test on Header Feature Only

Algorithm	Accuracy	Precision	Recall
LR	92	0.92	0.92
DT , J48	92.3	0.92	0.92
CART, One R	91.6	0.92	0.91
SMO	92.1	0.92	0.92
NaiveBayes	92.2	0.92	0.92

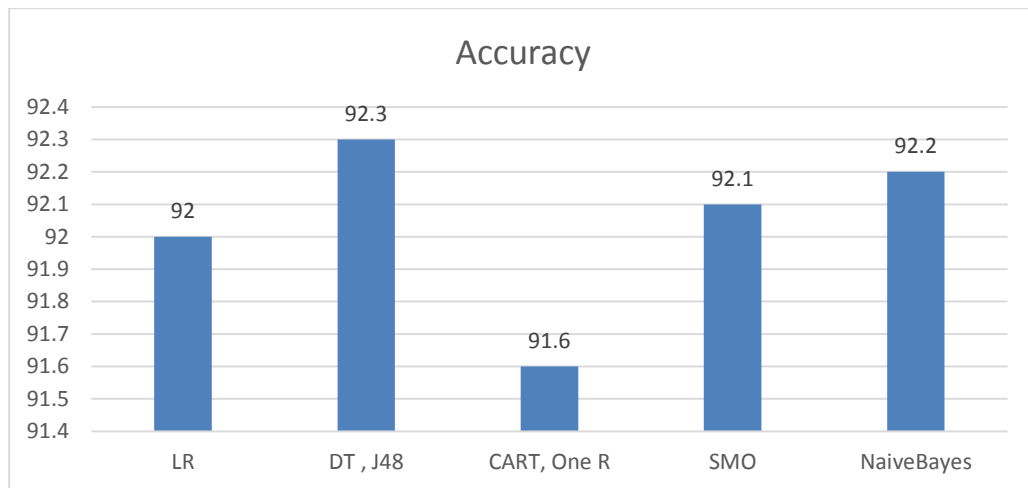


Figure (4.6) Accuracy of the Five Algorithms on Header Feature Only Test.

The results is close far from these obtained from the entire set. Thus, header feature is not a good representative for phishing detection task.

Then, the experiment was carried out on Java Script and external features. The results was summarized in Table 4.6 and plotted in Figure 4.7. The results show the LR classifier algorithm achieved the highest accuracy, precision and recall as of 96.4. While, the SMO classifier algorithm was the lowest result as 96.

Table (4.6) The Result of the Test on Java Script and External Features

Algorithm	Accuracy	Precision	Recall
LR	96.4	0.96	0.96
DT , J48	96.2	0.96	0.96
CART, One R	96.3	0.96	0.96
SMO	96	0.96	0.96
NaiveBayes	96.1	0.96	0.96

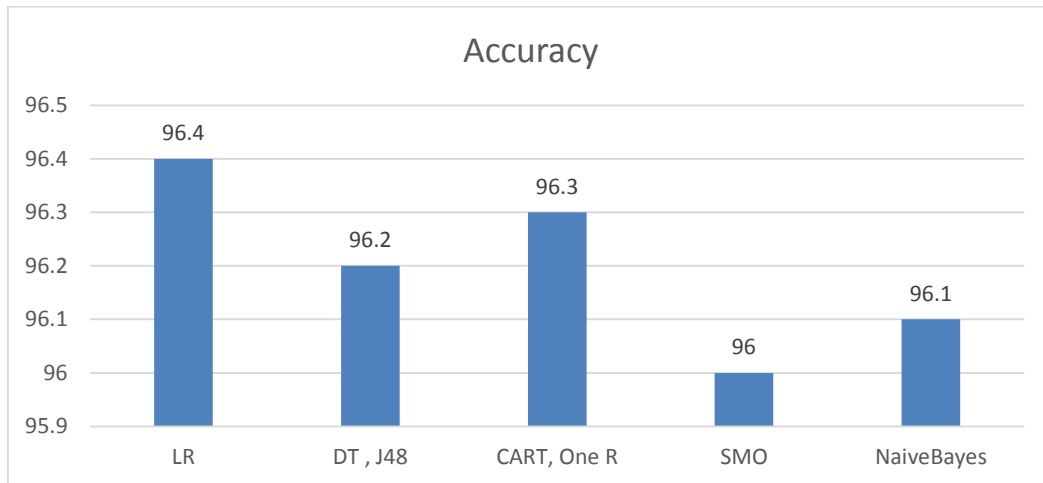


Figure (4.7) Accuracy of the Five Algorithms on Java Script and External Feature Only Test.

The results is close far from these obtained from the entire set. Thus, Java Script and external features is not a good representative for phishing detection task.

Then, the experiment was carried out on the entire features excluding the body feature. The results are summarized in Table 4.7 and plotted in Figure 4.8. The results show the DT classifier algorithm achieved the highest accuracy, precision and recall as of 98.2. While, the CART classifier algorithm was the lowest result as 96.3.

Table (4.7) The Result of the Test on All Features Excluding the Body Feature

Algorithm	Accuracy	Precision	Recall
LR	98.1	0.98	0.98
DT , J48	98.2	0.98	0.98
CART, One R	96.3	0.96	0.96
SMO	97.7	0.97	0.97
NaiveBayes	98.1	0.98	0.98

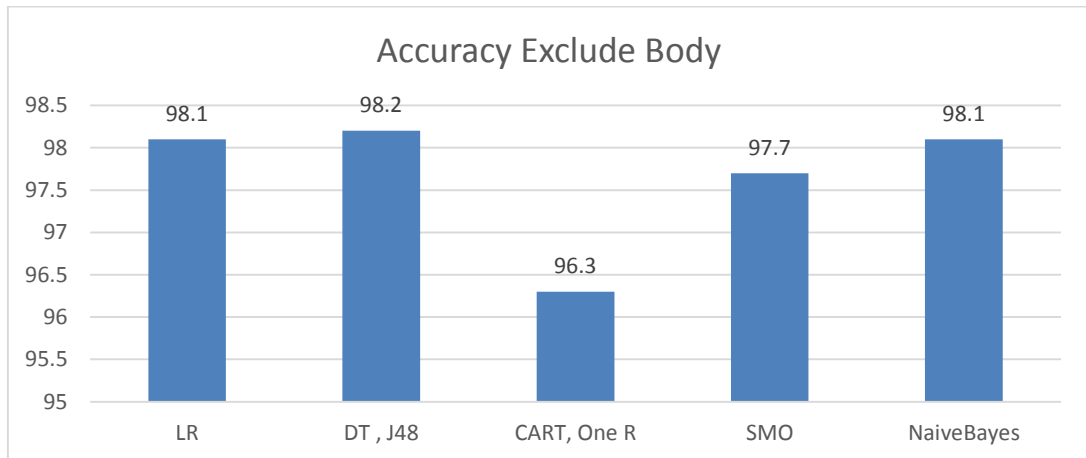


Figure (4.8) Accuracy of the Five Algorithms on all Feature Excluding Body Feature Test.

The results are close to those obtained from the entire set. Thus, the entire feature excluding the body feature is a good representative for phishing detection task.

Then, the experiment was carried out on the entire feature set excluding Java script and External features. The results are summarized in Table 4.8 and plotted in Figure 4.9. The results show the DT classifier algorithm achieved the highest accuracy, precision and recall as of 97.8. While, the CART classifier algorithm was the lowest result as 93.7.

Table (4.8) The Result of the Test on All Features Excluding Java Script and External Features

Algorithm	Accuracy	Precision	Recall
LR	97.6	0.97	0.97
DT, J48	97.8	0.97	0.97
CART, One R	93.7	0.94	0.94
SMO	97.5	0.97	0.97
NaiveBayes	96.2	0.96	0.96

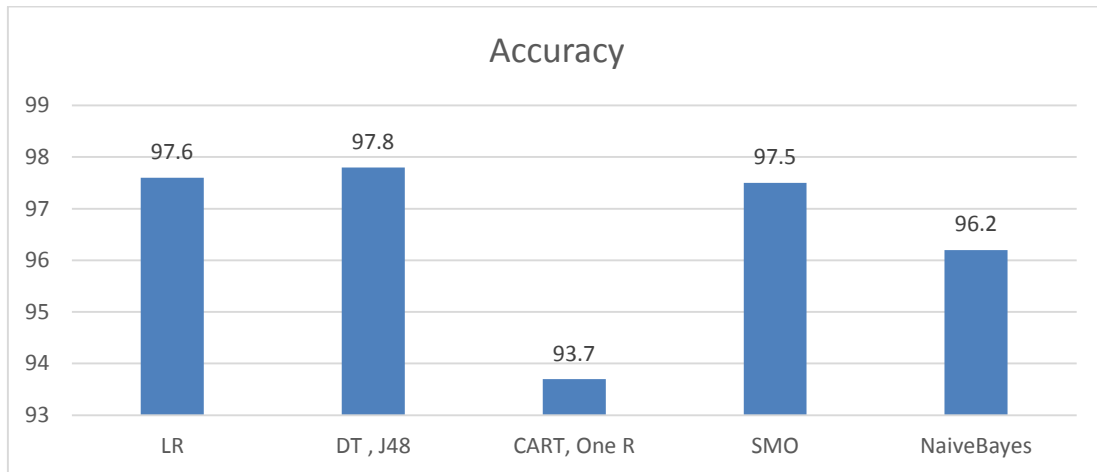


Figure (4.9) Accuracy of the Five Algorithms on All Feature Excluding Java Script And External Feature Test.

The results is close from these obtained from the entire set. Thus, the entire feature set excluding the Java script and external features is a good representative for phishing detection task.

Then, the experiment was carried out on the entire feature set excluding Header feature. The results are summarized in Table 4.9 and plotted in Figure 4.10. The results show the DT classifier algorithm achieved the highest accuracy, precision and recall as of 98.2. While, the CART classifier algorithm was the lowest result as 96.3.

Table (4.9) The Result of the Test on All Features Excluding Header Feature

Algorithm	Accuracy	Precision	Recall
LR	98.1	0.98	0.98
DT , J48	98.2	0.98	0.98
CART, One R	96.3	0.96	0.96
SMO	97.8	0.97	0.97
NaiveBayes	98	0.98	0.98

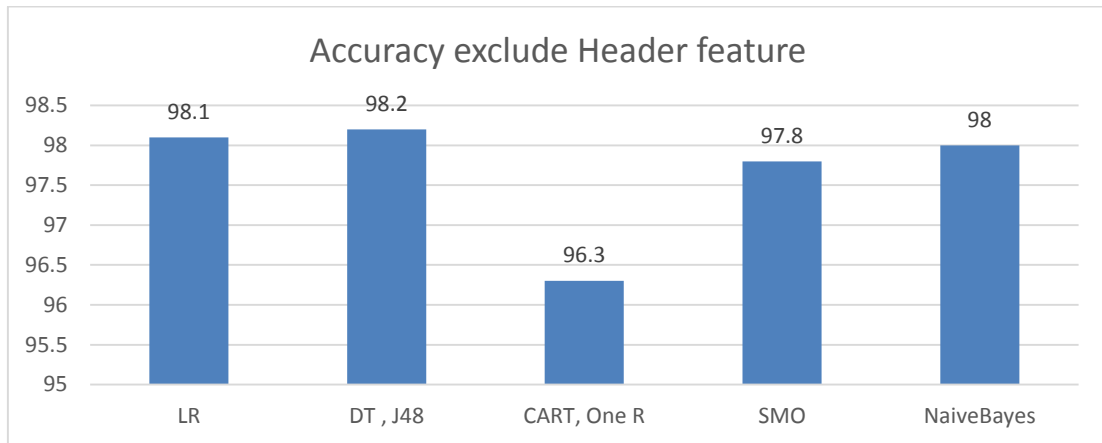


Figure (4.10) Accuracy of the Five Algorithms on All Feature Excluding Header Feature Test.

The results is close from these obtained from the entire set. Thus, the entire feature set excluding header feature is a good representative for phishing detection task.

Then, the experiment was carried out on the entire feature sets excluding URL feature. The results are summarized in Table 4.10 and plotted in Figure 4.11. The results show the NaiveBayes and SMO classifier algorithm achieved the highest accuracy, precision and recall as of 98.25. While, the CART classifier algorithm was the lowest result as 97.37.

Table (4.10) The Result of the Test on All Features Excluding URL Feature

Algorithm	Accuracy	Precision	Recall
LR	98.12	0.98	0.98
DT , J48	98	0.98	0.98
CART, One R	97.37	0.96	0.96
SMO	98.25	0.98	0.98
NaiveBayes	98.25	0.98	0.98

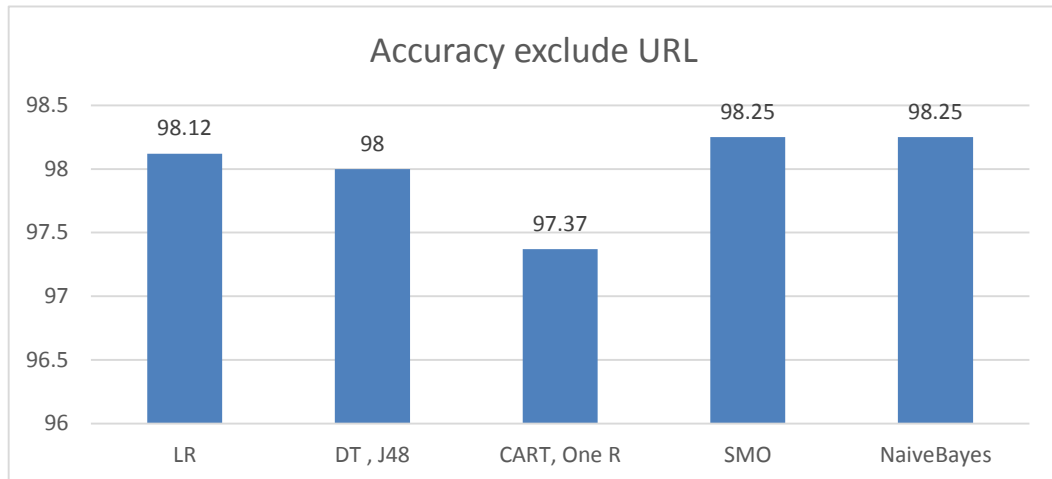


Figure (4.11) Accuracy of the Five Algorithms on All Feature Excluding URL Feature Test

The results is close from these obtained from the entire set. Thus, the entire feature set excluding the URL feature is a good representative for phishing detection task.

Then, the experiment was carried out on automated feature selection. The results are summarized in Table 4.11 and plotted in Figure 4.12. The system automatically generated three groups using Automatic selected features through a Classifier subset evaluator, consistency subset evaaluator with a genetic method search, each one contained different features as mention in section 3.3. The result showed high level of accuracy for group number 4 as of 98.6.

Table (4.11) Accuracy for Automated Generated Groups

Algorithm	Group 1	Group 2	Group 3
LR	97.37	75.25	98
DT, J48	97.5	75.25	98
CART, One R	96.5	75.12	97
SMO	97.37	87	98.25
NaiveBayes	95.87	65.37	98.25
Maximum	97.5	87	98.25

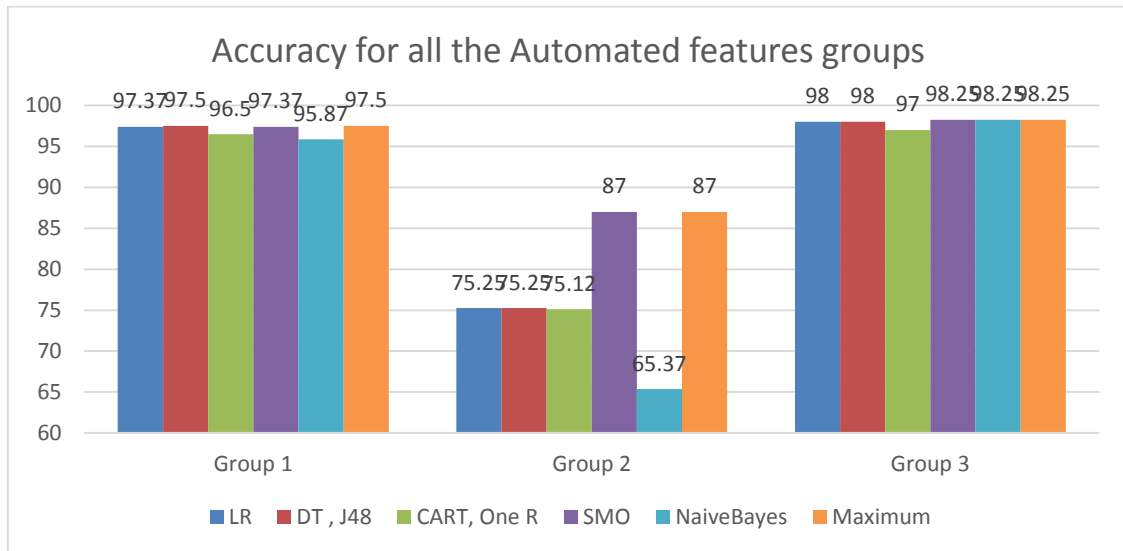


Figure (4.12) Accuracy for all Automated Features

The result showed almost the same average accuracy between the manual feature selection and the automated feature selection with difference 0.06 % as shown in table 4.12. Moreover, the Decision Tree (DT, J48) classifier algorithm has the highest average accuracy in both manual and automated scenarios as shown in Figure 4.13.

Table (4.12) Accuracy for both Manual and Automated Feature Selection

	Manual Selected Group									Automated		
Algorithm	All	Body	URL	Header	Java	All - Body	All - URL	All - Header	All - Java	G1	G2	G3
LR	97.75	95.62	93.6	92	96.4	98.1	98.12	98.1	97.6	97.37	75.3	98
DT , J48	98	96.5	95.7	92.3	96.2	98.2	98	98.2	97.8	97.5	75.3	98
CART, One R	97	95.75	93.7	91.6	96.3	96.3	97.37	96.3	93.7	96.5	75.1	97
SMO	98	96.62	90.7	92.1	96	97.7	98.25	97.8	97.5	97.37	87	98.25
NaiveBayes	97.37	96.75	91	92.2	96.1	98.1	98.25	98	96.2	95.87	65.4	98.25
Maximum	98	96.75	95.7	92.3	96.4	98.2	98.25	98.2	97.8	97.5	87	98.25

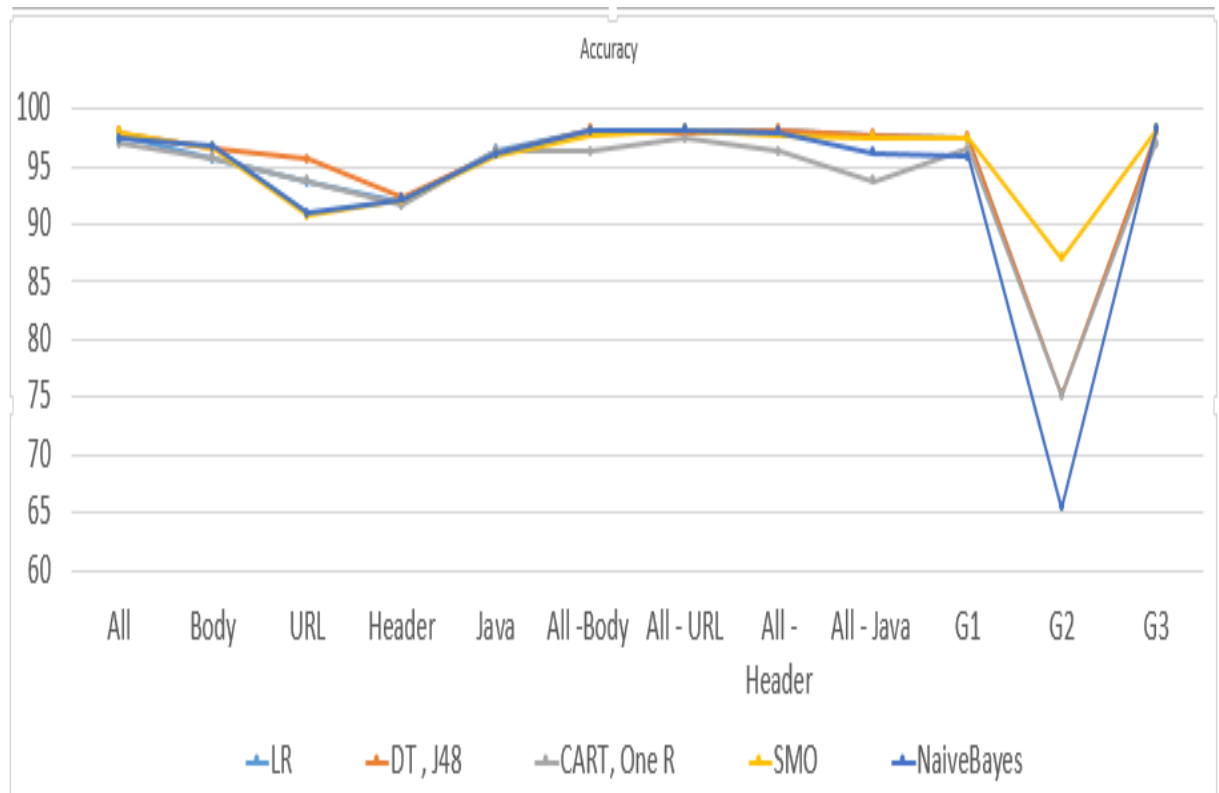


Figure (4.13) Accuracy for Five Classifier Algorithms in Both Scenarios

Then, the experiment was carried out on the Multi classifier integration. The multi classifier integrated between the LR, DT as the first two algorithms and SMO is the 3rd one as shown in Figure 4.14 . the impact of rescheduling the classifiers gives the same result of accuracy, precision and recall.

The results are summarized in Table 4.13. The results show the Multi classifier integration with clustering enhance the result of accuracy 98.37 as well as precision and recall.

AT	AU	AV	AW	AX	AY	AZ
url word login link	Url word Update link	Kmean Cluster	J48 result	LR result	SMO	Class
0	0	0	ham	ham	ham	ham
0	0	2	phish	phish	phish	phish
0	0	3	phish	phish	phish	ham
0	0	3	phish	ham	ham	ham
0	0	3	phish	ham	ham	ham
0	0	3	phish	ham	phish	ham
0	0	0	ham	ham	ham	phish

Figure (4.14) Sample of the Dataset with the three selected classifiers for the integrated system.

Table (4.13) The Result of test Multi-Classifier Integration

Multi-classifier integration	Accuracy	Precision	Recall
Without clustering	98.25	0.983	0.983
With clustering	98.37	0.981	0.988

Chapter Five

5.1 Conclusion

Phishing emails have become common problem in recent years. Phishing is a type of attack in which victims sent emails into which users have to provide critical information and then it directly sent to the phisher. So detection of that type of email is necessary. There are many techniques for detecting phishing email but there is some limitation like accuracy is low, content can be same as legitimate email so cannot be detected, detection rate is not high.

In this research, the accuracy of phishing email detection were evaluated based on manual feature selection and automated feature selection on five classifier algorithms. Finally, comparison between the two scenarios was conducted.

For manual feature selection, 47 email features were selected and grouped in four groups (body features, Header features, URL features and Java script features with external features) according to the email structure. The results showed that the body group obtained the highest accuracy as of 96.75 in detecting phishing email.

On the other hand, the accuracy was tested for all features together excluding one of the four groups each time. The result showed, the highest accuracy 98.25 was obtained when we excluded the URL features group from the all features.

For the automated selection, the accuracy was tested on three groups, which were automatically generated by the system using automatic selected features. The result showed that there are a difference in the accuracy among the three groups. The highest group was group number three as it achieved accuracy equal to 98.25, which is equal to the result of manual feature selection despite only 33 features were used in the Group no.

3 of the automated features selection comparing to 30 features were used in the manual feature selection group that achieved the highest accuracy.

The Decision Tree (DT, J48) classifier algorithm proved its efficiency in phishing email detection in manual feature selection. While, SMO proved its efficiency in phishing email detection in automated scenarios regardless if the selected features are small or big.

Finally, the Multi classifier integration results show that the clustering Emails before the classification enhance the result of accuracy 98.37 as well as precision and recall.

5.2 Future Work

Feature selection techniques need more improvement to cope with the continuous development of new techniques by the phishers over time. Therefore, we recommend developing a new automated tool in order to extract new features from new raw emails to improve the accuracy of detecting phishing email and to cope with the expanding phisher techniques.

References

- Abu-Nimeh, S., Nappa, D., Wang, X., & Nair, S. (2007, October). A comparison of machine learning techniques for phishing detection. In Proceedings of the anti-phishing working groups 2nd annual eCrime researchers summit (pp. 60-69). ACM.
- Abu-Nimeh, S., Nappa, D., Wang, X., & Nair, S. (2009, June). Distributed phishing detection by applying variable selection using Bayesian additive regression trees. In Communications, 2009. ICC'09. IEEE International Conference on (pp. 1-5). IEEE.
- Adida, B., Chau, D., Hohenberger, S., & Rivest, R. L. (2006). Lightweight email signatures. In Security and Cryptography for Networks (pp. 288-302). Springer Berlin Heidelberg.
- Akinyelu, A. A., & Adewumi, A. O. (2014). Classification of phishing email using random forest machine learning technique. Journal of Applied Mathematics.
- Alguliev, R. M., Aliguliyev, R. M., & Nazirova, S. A. (2011). Classification of textual e-mail spam using data mining techniques. Applied Computational Intelligence and Soft Computing, 10.
- Al-Momani, A., Gupta, B. B., Atawneh, S., Meulenberg, A., & Al-Momani, E. (2013). A survey of phishing email filtering techniques. Communications Surveys & Tutorials, IEEE, 15 (4), 2070-2090.
- Al-Momani, A., Gupta, B. B., Wan, T. C., Altaher, A., & Manickam, S. (2013). Phishing dynamic evolving neural fuzzy framework for online detection zero-day phishing email.

- Al-Momani, A., Wan, T. C., Altaher, A., Manasrah, A., Al-Momani, E., Anbar, M., Ramadass, S. (2012). Evolving fuzzy neural network for phishing emails detection, *Journal of Computer Science*, 8, 1099.
- Almomani, A., Wan, T. C., Manasrah, A., Altaher, A., Baklizi, M., & Ramadass, S. (2013). An enhanced online phishing e-mail detection framework based on evolving connectionist system. *International Journal of Innovative Computing, Information and Control (IJICIC)*, 9(3), 169-175.
- Altaher, A., Al-Momani, A., Wan, T. C., Manasrah, A., Al-Momani, E., Anbar, M., ... & Ramadass, S. (2012). Evolving fuzzy neural network for phishing emails detection. *Journal of Computer Science*, (7), 1099.
- Apacheorg. (2016). Apacheorg. Retrieved 18 November, 2016, from <http://spamassassin.apache.org/publiccorpus/>
- Azad, B. Identifying Phishing Attacks.
- Basnet, R. B., & Sung, A. H. (2010). Classifying phishing emails using confidence-weighted linear classifiers. In *International Conference on Information Security and Artificial Intelligence (ISAI)* (pp. 108-112).
- Bergholz, A., Chang, J. H., Paass, G., Reichartz, F., & Strobel, S. (2008, August). Improved Phishing Detection using Model-Based Features. In *CEAS*.
- Cao, Y., Han, W., & Le, Y. (2008). Anti-phishing based on automated individual white-list. In *Proceedings of the 4th ACM workshop on Digital identity management* (pp. 51-60).
- Chandrasekaran, M., Narayanan, K., & Upadhyaya, S. (2006, June). Phishing email detection based on structural properties. In *NYS Cyber Security Conference* (pp. 1-7).

- Chhabra, S. (2005). Fighting spam, phishing and email fraud (Doctoral dissertation, University of California Riverside).
- Gansterer, W. N., & Pölz, D. (2009). E-mail classification for phishing defense. In *Advances in Information Retrieval* (pp. 449-460). Springer Berlin Heidelberg.
- Jain, A., & Richariya, V. (2011). Implementing a web browser with phishing detection techniques. arXiv preprint arXiv:1110.0360.
- Jameel, Noor Ghazi M., and Loay E. George. Detection of phishing emails using feed forward neural network. *International Journal of Computer Applications* 77 2013.
- Kathirvalavakumar, T., Kavitha, K., & Palaniappan, R. (2015). Efficient Harmful Email Identification Using Neural Network, *British Journal of Mathematics & Computer Science*, (1), 58.
- Khonji, A. M., & Iraqi, Y. (2011). A Brief Description of 47 Phishing Classification Features.
- Khonji, M., Iraqi, Y., & Jones, A. (2013). Enhancing phishing E-Mail classifiers: a lexical URL analysis approach. *International Journal for Information Security Research (IJISR)*, 2(1/2).
- Kumar, R. K., Poonkuzhali, G., & Sudhakar, P. (2012, March). Comparative study on email spam classifier using data mining techniques. In *Proceedings of the International MultiConference of Engineers and Computer Scientist* (Vol. 1, pp. 14-16).
- Kumaraguru, P., Sheng, S., Acquisti, A., Cranor, L. F., & Hong, J. (2010). Teaching Johnny not to fall for phish. *ACM Transactions on Internet Technology (TOIT)*, 10(2), 7.

- Kumaraguru, P., Rhee, Y., Acquisti, A., Cranor, L. F., Hong, J., & Nunge, E. (2007, April). Protecting people from phishing: the design and evaluation of an embedded training email system. In Proceedings of the SIGCHI conference on Human factors in computing systems (pp. 905-914). ACM.
- Lungu, I., & Tabusca, A. (2010). Optimizing anti-phishing solutions based on user awareness, education and the use of the latest web security solutions. *Informatica Economica*, 14(2), 27.
- Ma, L., Yearwood, J., & Watters, P. (2009, September). Establishing phishing provenance using orthographic features. In eCrime Researchers Summit, 2009. eCRIME'09. (pp. 1-10). IEEE.
- Manning, R., & Aaron, G. (2015). Phishing Activity Trends Report. Anti-Phishing Work Group, Tech. Rep. 1st -3rd Quarter.
- Microsoft Consumer Safety Index reveals impact of poor online safety behaviors in Singapore. (2014, February 11). Retrieved March 11, 2016, from <https://news.microsoft.com/en-sg/2014/02/11/microsoft-consumer-safety-index-reveals-impact-of-poor-online-safety-behaviours-in-singapore/>
- Monkeyorg. (2016). Monkeyorg. Retrieved 18 November, 2016, from <http://monkey.org>
- Nizamani, S., Memon, N., Glasdam, M., & Nguyen, D. D. (2014). Detection of fraudulent emails by employing advanced feature abundance. *Egyptian Informatics Journal*, 15(3), 169-174.
- Paaß, G., & Bergholz, A. (2009). Project Exhibition: AntiPhish-Machine Learning for Phishing Detection.

- Pandey, M., & Ravi, V. (2012, December). Detecting phishing e-mails using text and data mining. In *Computational Intelligence & Computing Research (ICCIC), 2012 IEEE International Conference on* (pp. 1-6). IEEE.
- Parmar, B. (2012). Protecting against spear-phishing. *Computer Fraud & Security*, 8-11.
- Ramanathan, V., & Wechsler, H. (2012). phishGILLNET—phishing detection methodology using probabilistic latent semantic analysis, AdaBoost, and co-training. *EURASIP Journal on Information Security*, 1-22.
- Rathi, M., & Pareek, V. (2013). Spam Mail Detection through Data Mining-A Comparative Performance Analysis. *International Journal of Modern Education and Computer Science*, (12), 31.
- Safavian, S. R., & Landgrebe, D. (1990). A survey of decision tree classifier methodology.
- Steinberg, D., & Colla, P. (2009). CART: classification and regression trees. *The top ten algorithms in data mining*, 9, 179.
- Toolan, F., & Carthy, J. (2009, September). Phishing detection using classifier ensembles. In *eCrime Researchers Summit, 2009. eCRIME'09.*(pp. 1-9). IEEE.
- Weka (2016). Machine Learning Group at the University of Waikato
<http://www.cs.waikato.ac.nz/ml/weka/>
- Wu, Y., Zhao, Z., Qiu, Y., & Bao, F. (2010, May). Blocking foxy phishing emails with historical information. In *Communications (ICC), 2010 IEEE International Conference on* (pp. 1-5). IEEE.
- Zhang, N., & Yuan, Y. (2013). Phishing detection using neural network. Department of Computer Science, Department of Statistics, Stanford University. Web, 29.