Design and Development of Antispammer for SMS Spam Detection

Thesis submitted in partial fulfillment of the requirements for the award of degree of

Master of Technology

in

Computer Science and Applications

Submitted By

Sakshi Agarwal

(Roll No. 601303027)

Under the supervision of

Dr. Sanmeet Kaur

Assistant Professor

Ms. Sunita Garhwal

Assistant Professor

COMPUTER SCIENCE AND ENGINEERING DEPARTMENT

THAPAR UNIVERSITY

PATIALA – 147004

June 2015

Certificate

I hereby certify that the work which is being presented in the thesis entitled, "*Design and Development of Antispammer for SMS Spam Detection*", in partial fulfillment of the requirements for the award of degree of Master of Technology in *Computer Science and Applications* submitted in Computer Science and Engineering Department of Thapar University, Patiala, is an authentic record of my own work carried out under the supervision of Ms. Sunita Garhwal and Ms. Sanmeet Kaur and refers other researcher's work which are duly listed in the reference section.

The matter presented in the thesis has not been submitted for award of any other degree of this or any other University.

Sakshi Agarwal

(Sakshi Agarwal)

This is to certify that the above statement made by the candidate is correct and true to the best of my knowledge.

Sunita 18/6/15

Ms. Sunita Garhwal

Assistant Professor

CSED

Sanmeet Kaur

Ms. Sanmeet Kaur

Assistant Professor

CSED

Countersigned by

(Dr. Deepak Garg)

Head

Computer Science & Engineering Department

Thapar University

Patiala

(Dr. S. S. Bhatia)

Dean (Academic Affairs)

Thapar University

Patiala

Acknowledgement

I would like to express my sincere gratitude to my mentors and supervisors Dr. Sanmeet Kaur and Ms. Sunita Garhwal for their immense help, guidance, stimulating suggestion and full time encouragement. They always provided me a motivational and enthusiastic atmosphere to work with. It was a great pleasure to do dissertation under their supervision.

I am also thankful to Dr. Deepak Garg, Head, Computer Science & Engineering Department for his constant support and encouragement.

I would like to thank all the faculty members and staff of the department who were always there at the need of the hour and provided with all the help and facilities, which I required for the completion of this work.

I express thanks to my family for their love, support and enthusiastic encouragement without which I could not complete this dissertation. I also want to thank my friends for their moral support throughout the completion of this work, especially Varun Chauhan, without whose support I would not have been able to collect new data and Gurpreet Pal Singh, without whose help I would not have been able to connect with python in a short period.

Finally I thank the Almighty who gave me the strength to complete this work.

(Sakshi Agarwal)

601303027

Abstract

The growth of the mobile phone users has led to a dramatic increase in SMS spam messages. Though in most parts of the world, mobile messaging channel is currently regarded as “clean” and trusted, on the contrast recent reports clearly indicate that the volume of mobile phone spam is dramatically increasing year by year. The success of the mobile messaging channel has, unfortunately, made it a very attractive target for attack by spammers. It is an extremely growing problem, primarily due to the availability of very cheap bulk pre-pay SMS packages and the fact that SMS engenders higher response rates as it is a trusted and personal service. Here trust means almost all the messages received by the subscribers are opened and read at least once. Also because of the ease of use of Smartphones, numbers are easily dialled or links can be smoothly clicked, exposing the subscriber to more risk. To further exacerbate, the situation attackers are finding the traditional fixed email channel increasingly unprofitable and are focusing their activities on the SMS channel.

The growing volume of spam messages has increased the demand for accurate and efficient spam solutions. SMS spam filtering is a relatively new task which inherits many issues and solutions from email spam filtering. Many spam solutions have been proposed in the recent past. The one which we address in this thesis, treats spam detection as a simple two class document classification problem. The solution will consist of classification algorithm coupled with feature extractions. Classification along with appropriate features helped us improving the performance in terms of accuracy and has lesser computational time and storage requirements.

In this dissertation, we compared the performance achieved by several established machine learning techniques with our approaches. Additionally, we present details about a real and public SMS spam collection based on the perspective of Indian hams and spams. Further, it was analysed with various classifiers for the best results.

Table of Contents

List of Figures

List of Tables

List of Snapshots

List of Abbreviations

DoS	**D**enial **o**f **S**ervice
SMS	**S**hort **M**essage **S**ervice
MMS	**M**ultimedia **M**essaging **S**ervice
IP	**I**nternet **P**rotocol
RoI	**R**eturn **o**n **I**nvestment
MNO	**M**obile **N**etwork **O**perator
SMSC	**S**hort **M**essage **S**ervice **C**entre
SCCP	**S**ignalling **C**onnection **C**ontrol **P**art
VoIP	**V**oice **o**ver **I**nternet **P**rotocol
ISP	**I**nternet **S**ervice **P**rovider
TF	**T**erm **F**requency
TFIDF	**T**erm **F**requency-**I**nverse **D**ocument **F**requency
TP	**T**rue **P**ositive
FP	**F**alse **P**ositive
TN	**T**rue **N**egative
FN	**F**alse **N**egative
ACC	**Acc**uracy
SC	**S**pam **C**aught
BH	**B**locked **H**am
MCC	**M**athews **C**orrelation **C**oefficient
AuC	**A**rea **u**nder **C**urve
TPR	**T**rue **P**ositive **R**ate
MNB	**M**ultinomial **N**aive **B**ayes
SVM	**S**upport **V**ector **M**achine
Ada Boost	**Ada**ptive **Boost**ing
ROC	**R**eceiver **O**perating **C**haracteristic
CSV	**C**omma **S**eparated **V**alue
CV	**C**ross **V**alidation

Chapter 1

Introduction

SMS spams are dramatically increasing year by year due to the growth of mobile phone users around the world. Recent reports have clearly indicated the same. Mobile or SMS spam is a physical and thriving problem due to the fact that bulk pre-pay SMS packages are conveniently available these days and SMS is considered as a trusted and personal service. SMS spam filtering is a comparatively recent errand to deal such a problem. The volume of data traffic moving over the network is increasing exponentially and the devices which are connected to it are very much vulnerable. So there is a greater need to secure our system from this type of vulnerability, here network security play a very important role in this context.

1.1 Network Security

Network security is a process that keeps unauthorized parties away from gaining access on the network. Important information can prove to be an asset, which is protected from being attacked by the hackers. Hackers make use of loopholes and try to break in the security of the network due to which individual user or even an organization gets affected. The attacks mainly occur due to the failure in the implementation of the security policies resulting in corruption or loss of important data. Security policies give a brief description of what to be secured to support the business or mission. The following are the major goals of security [1]:

- Confidentiality: Information needs to be hidden from unauthorized access.
- Integrity: Information needs to be protected from unauthorized change.
- Availability: Information must be available to an authorized entity when it is needed.

1.2 Need of Security

The computer security Institute/FBI 2003 Computer Crime and Security survey indicated that the total annual losses reported in the 2003 were $201797,340 [2], thus proving that the information security is needed for carrying out the following tasks:

- To protect the important and sensitive information from unauthorized access.
- To protect information from editing, accidently or intentionally by unauthorized users.
- To protect that the information is delivered to proper place and the data do not lost.

1.3 Security Attacks

The three goals of security that are confidentiality, integrity and availability are at risk because of the continuous security attacks. The following section explains the security attacks that threaten the goals of security [3].

1.3.1 Attacks Threatening Confidentiality

- Snooping: It means interception of data by unauthorized entity. This may be done to intercept the confidential data.
- Traffic analysis: By looking at the type of traffic, the type of information exchanged between the sender and the receiver could be extracted.

1.3.2 Attacks Threatening Integrity

- Masquerading/ Spoofing: The attacker tries to impersonate somebody else to gather information from the user.
- Modification: The attacker can change the information after getting access to the information for his/her own benefit.
- Replaying: The attacker gets a copy of the message and later tries to replay it. The attacker replays the message for his/her own benefit.

- Repudiation: The sender of the message may later deny that he/she has sent the message and the receiver of the message may later deny that he/she has received the message.

1.3.3 Attacks Threatening Availability

- Denial of Service (DoS): Denial of service attack halts the services of the system. The attacker may overload the system with bogus requests causing the system to crash.

1.4 Security Services

Following are the security services that are related to services goals and attacks [4]:

1.4.1 Data Confidentiality

The protection of passing data and transmitted from passive attacks is Confidentiality. With regard to data transmission there can be several levels of protection. At macro level data confidentiality refers to the protection of all user data that is transmitted between two users over a period of time. At narrower level even the protection of a single message or even specific fields can be put under this service. Another important feature of confidentiality is the security of traffic from analysis. This includes that an attacker or a hacker should not be able to trace the source and destination, frequency, length or other characteristics of the traffic over a communications facility.

1.4.2 Data Integrity

A connection oriented integrity service has an important role to play in the streaming of messages. It ensures that the received messages are in their original form, implying the same form in which they were sent. It ensures that the message should not be replicated or altered while sending. This service also covers the destruction of data as well. Thus, both message modification and denial of service are covered under the connection-oriented integrity service.

A connectionless integrity service, on the other hand, deals with individual messages. It therefore, provides security against message modification only. The integrity services are related to active attacks. When a violation of integrity is detected, then such a violation is simply reported by the service that this violation and other portion of software needs a human intervention to recover from it.

1.4.3 Authentication

The authentication service concerns with assuring that a communication between two entities is authentic. When there is a single message, the authentication service assures the recipient that the message is from the same source that it claims to be from. Also, the service also makes sure that the connection is not interfered in such a way that a third party can pretend as one of the two legitimate parties for unauthorized transmission or reception purposes.

Two specific authentication services are:

- **Peer entity authentication:** It provides the evidence for the identity of a peer entity in an association. This service ensures that an entity is not involved in a masquerade or an unauthorized replay of a previous connection.
- **Data origin authentication:** It gives the evidence for the source of a data unit. Although protection against the duplication or modification of data units is not given. Applications such as electronic mail where there are no prior interactions between the communicating entities come under this type of service.

1.4.4 Access Control

Limiting and controlling the access to host systems and applications is called Access control. For achieving this, each entity that is trying to gain access has to be first authenticated in order to give access rights to the individual. In other words, selective restriction of access to a place or other resources is Access Control.

1.4.5 Non Repudiation

Non repudiation is a service that helps in preventing either sender or receiver from denying a transmitted message. This means when a message is sent, the receiver can

prove that it is sent by the alleged sender. In the same way, when a message is received, it can be proved by the sender that the message has been received by the alleged receiver. In this condition when the authenticity of a signature is being challenged, the third party is introduced which is trusted by both sender and the receiver for the checking of the signature. For the purpose of cryptographic text Digital signature is also used in this context.

1.5 Security Mechanisms

The following are the security mechanisms that provide security services [5].

- **Encipherment:** Encipherment means hiding or covering data. The two techniques used for encipherment are cryptography and steganography. This mechanism helps in providing data confidentiality.

- **Data integrity:** To check whether the data integrity is preserved or not, a check value can be used. The check value is sent by the sender along with the data. The receiver also creates its own check value and then the receiver can compare its created check value with the one received by the receiver to check the integrity of the message.

- **Digital Signature:** It refers to the data attached to a data unit which allows a recipient of the data unit to prove the source and integrity of the data unit and safeguard against forgery.

- **Authentication Exchange:** The messages are exchanged between entities to prove their identity to each other.

- **Traffic Padding:** Traffic padding is used to prevent the attacker from doing traffic analysis by inserting data that is not useful.

- **Routing Control:** The routes between the sender and the receiver are continuously changed so as to prevent the attacker from checking a particular route.

- **Notarization:** The communication between the sender and the receiver can be controlled by a third trusted party. This mechanism is used to prevent repudiation.

- **Access Control:** The passwords and PINs can be used as access control methods to allow user to access data.

1.6 Types of Mobile Messaging Attacks

The various types of mobile messaging attacks seen in networks today are [6]:

- **SMS Spam**: This is the most basic form of attack where unwanted messages are sent by the attackers to the end users for bulk marketing and Social Engineering Viral Hoaxes.
- **Premium rate fraud**: These are those uninvited messages which are sent to trick customers to call premium rate numbers or sign up for the subscription services that are charged to the bill for people.

  "CONGRATULATIONS! YOUR CELL NO.HAS WON 500.000 POUNDS IN THE ONGOING SONY ERICSSON MOBILE PROMO. FOR CLAIM CALL +447045754969".
- **Phishing (including SMShing)**: These are the uncalled messages which ask users to call at specific numbers by the help of which attackers become successful in obtaining classified information, which is then misused for fulfilling wrong goals.

  "BANK OF THE CASCADES: urgent account notification, verify unusual activity, call 1800-####."
- **Mobile Malware including Bots spreading via messaging**: Malware, an abbreviation for malicious software, penetrates a mobile device devoid of the approval of owner consent and owner does not even come to know about it. Actually it comprises of sending unsolicited links to the users and requesting them to download the executable that is dangerous and finally which leads to application abuses. Three of the most common forms of malware include:

i) *Virus:* It is a deleterious web application which reproduces itself. It can only infect a new host if it is distributed to the host through some means outside of the capability of the computer program.

ii) *Worm:* Self-propagating malicious computer program. It replicates itself and exercises several means for sending those copies to other nodes on the network. A worm is very harmful for any device as it extent and contaminate numerous hosts at a fast pace in a networked environment.

iii) *Trojan:* A computer program that doesn't replicate, but in its place authorizes hacker's un-authorized admittance to the affected host. Keystroke loggers are a severe form of a Trojan.

1.7 Market Trends Resulting in an Increase of SMS Attacks

Short Message Service (SMS) is one of the most flourishing cellular service engendering millions of dollars in perquisite for mobile operators yearly. Current estimations indicate that billions of SMS's are sent every day. According to the Portio Research [7], in 2010 the cost of total global cellular messaging market was around USD 179.2 billion which hiked to USD 200 billion in 2011 and has passed USD 253 billion till Sep, 2014. As the huge rise has occurred in SMS market, its profit has also increased in the direct proportion. Figure 1.1 shows the increase in profit in SMS comparing with the decrease in profit in Email.

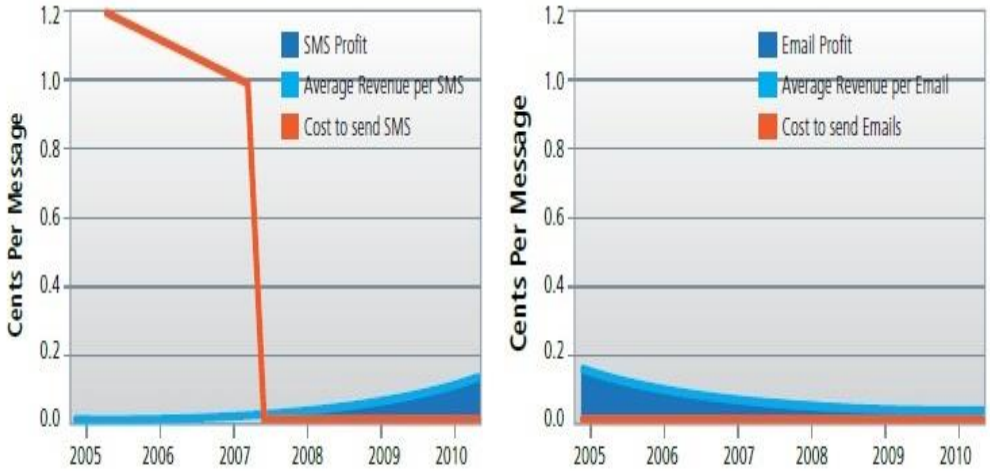


Figure 1.1: Profits comparison between SMS and email [6]

Spams are undesirable and unwelcomed messages which are sent electronically. These messages are sent by spammers for different ill wills of taking a hold over user's personal data or tricking them into the subscription of their premium tariff facilities. In the Email world, though spam is a properly handled obstacle but SMS spam experiences a yearly growth of more than 500%. It is an evolving setback especially in the Middle East and Asia. As an example of this is Chinese mobile subscribers received 200 billion spam messages in one week in 2008 [8].

According to the Cloudmark Report [9], the amount of mobile phone spam is not same in every region. For instance, in North America, much less than 1% of SMS messages were spam in 2010, while in parts of Asia up to 30% of messages were represented by spam. Moreover 200 billion spam messages have been received by people of China in just one week of 2008. Messaging attacks are primarily driven by a desire by the attacker to make money. There are five primary market drivers that have emerged over the last 3 years, leading to the hike in messages attacks [6]:

- **Driver #1: Networks are faster, open on the access side, open to the Internet and application portals**

  Mobile networks are increasingly under threat due to their evolution from closed, circuit switched networks accessible via voice only handsets, to open Internet Protocol (IP) based networks. The access side has opened up to mobile devices and 3G data cards.

- **Driver #2: Users are demanding more applications on their mobile phones**

  Devices are becoming increasingly powerful and have the capacity to run a wide range of user downloaded applications. Attackers are able to embed malware within these applications with relative ease.

- **Driver #3: The SMS channel is regarded as clean and secure**

  There is an unprecedented level of trust in SMS and subscribers are comfortable with using SMS for confidential information exchange, payment authorization and accessing financial and other critical applications on their mobile devices.

- **Driver #4: Move towards all-you-can-use unlimited messaging data plans**

  As discussed in the introduction unlimited messaging plans are making the economics and the ROI for sending mobile spam via SMS much more attractive and lucrative than ever before.

- **Driver #5: Consumer Demand for SMS over Traditional Email Vehicles**

  SMS has surpassed email as the number one form of communications between individuals around the world, with more than 75% of the global population being active users of SMS/MMS messaging technologies. SMS is popular because the messaging protocol is supported virtually on every phone, and is often viewed as a more efficient and less invasive form of communications compared to voice.

1.8 Consequences of Message Attacks

A consequence of this increase in SMS attacks is that mobile network operators (MNO's) are seeing their brand value and profitability eroded in a number of ways [10]:

- **Poor Customer Experience:** Mobile spam is regarded as a personal intrusion and negatively impacts the customer experience and damages the MNO's reputation amongst its subscribers. This can lead to subscriber churn to other MNO's or reduction in usage of SMS / data services, particularly if the only available option to prevent spam is to restrict messaging services.

- **Higher Infrastructure Cost:** As the volume of mobile spam and other malware increases, the MNO must add additional capacity to its network, particularly to messaging servers and SMSC infrastructure to cope with peaks caused by attacks.

- **Higher Operational Cost:** Mobile messaging abuse generates an increase in customer complaint calls to MNO help desks. These have to be investigated and refunds often have to be made. A major attack from outside the MNO's own network will often result in costly mitigation and refunding of inter-carrier charges.

- **Interconnect Issues:** When receiving large volumes of spam from another operator, some MNOs will “cut off” incoming SMS and MMS messages from the originating operator. This periodic cycle of blocking and then restoring interconnection causes operational headaches and costs for both operators.

- **Threat of Regulation:** Lack of an effective industry response to mobile messaging abuse invites regulators to impose regulatory requirements on MNOs, as seen recently in India where subscribers are limited to 100 SMS messages per day and unlimited text plans are banned.

1.9 Types of SMS Related Mobile Signaling Abuse

The perpetrators of SMS attacks have traditionally used mobile signaling abuse techniques to gain access to an MNO’s network. This is becoming less prevalent as more networks offer unlimited text plans but is still a problem in many regions. The following is a partial list of some of the popular messaging related signaling abuse observed in the industry [11].

- **SMS spoofing:** SMS spoofing is when the identity of the sender is taken over by a hacker. SMS messages are sent for free by the hacker whilst the victim is charged for sending this fraudulent traffic. This scenario can be accomplished using a mobile switching center emulator in a roaming scenario. The emulator sends the message to the victim’s home Short Message Service Centre (SMSC) whilst pretending the victim is roaming in a foreign network.

- **SMS faking:** It occurs at the time when the hacker succeeds in gaining an unauthorized access to the Mobile Network Operator’s network by faking the Signaling Connection Control Part (SCCP) calling and called party addresses. This enables the hacker to send free messages on the victim’s network whilst pretending the messages have come from another network. Some operators are protected against both SMS spoofing and faking with messaging security solutions integrating into their mobile SMSC/SMS router infrastructure.

- **SMS flooding:** SMS flooding occurs at the time when unwanted SMS's are sent to a user, which can cause a denial-of-service condition in both the core network and radio access networks. Mobile messaging security solutions protect against flooding type attacks using SMS sender rate-limiting algorithms, volume controls, user-reports, sender reputation and sender intelligence.

1.10 Types of Spam

Broadly Spam can be categorized into six major types; which are discussed as follows [12]:

- E-mail Spam
- Instant Messaging Spam
- Newsgroup Spam
- Mobile Phone Spam
- Internet Telephony Spam
- Spamdexing

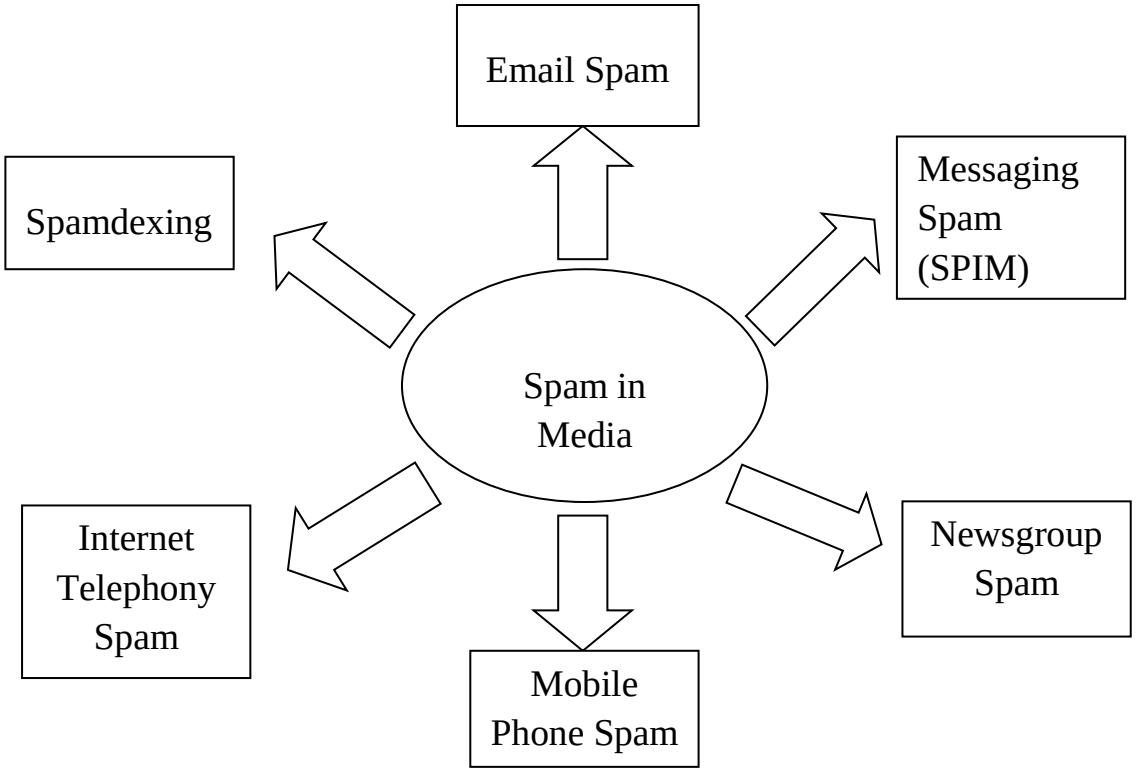


Figure 1.2: Types of Spam

1.10.1 E-mail Spam

Email spam is considered as the most prevalent and recognized form of spam. It targets the individual users through direct mails. Spammers firstly create a canon of e-mail users by scanning Usenet postings and stealing internet mail lists. Then they surf web for e-mail addresses. While the user is reading the e-mails, the cost meter is running and in this way E-mail spam costs money to the user of e-mail. The ISPs too incur the cost due to Email spams because when a bulk of spam mails are sent to the e-mail users a lot of wastage happens in the band width of the service providers and these costs are transmitted to users.

1.10.2 Instant Messaging Spam

Instant messaging systems such as Yahoo! Messenger, Windows Live Messenger, AIM, ICQ, XMPP , Tencent QQ and MySpace chat rooms are the common targets for spammers. Many IM systems offer a directory of individuals (users) including demographic information like age, and gender. Advertisers can retrieve this information easily by signing on to the system and send unsolicited messages which may include commercial scam-ware, viruses, as well as links to paid websites. The services like that of instant messaging are normally not blocked by firewalls, therefore it becomes especially useful and prone channel for spammers.

1.10.3 Newsgroup Spam

Newsgroup spam refers to posting of some random advertisement to the newsgroups. Spammers target those users who read news from these newsgroups. The advertisements are thus posted to many newsgroups at a time. A barrage of advertising and other irrelevant posts overwhelm the users and they robbed of the utility of the newsgroups through Newsgroup Spam.

1.10.4 Mobile Phone Spam

Mobile phone spam is directed towards text messaging service of cell phones. This can be especially irritating to customers not only because of the inconvenience they cause but

also because of the cost as they may be charged equal to a text message received in some networks. This kind of spams usually contains some promotional schemes and offers on various products and services. Many times service providers make use of this to trap the subscribers for activation of some paid service.

1.10.5 Internet Telephony Spam

This kind of spam refers to the pre-recorded, unnecessary, undesired bulk of telephone calls that are automatically processed on the internet which can be easily spread using the Voice over Internet Protocol (VoIP). These are similar to E-mail spam but are disturbing to the user as these are in the form of a phone call.

1.10.6 Spamdexing

Spamdexing consists of two words "Spam" and "Indexing". 'Spam' means flooding the Internet with many copies of a single message and Indexing means the systemic arrangement of data. So search engine spamming can be done by this practice of spamdexing.

1.11 The Spam Solutions

Many people have suggested different kinds of solutions to the spam problem. Some of them have been implemented with quite success while others which are mostly non technological solutions provide good attractive ideas with lots of hurdles to implement. In the following subsections a brief overview of these solutions has been discussed [13].

1.11.1 Non Technological solutions

To deal with the problem of spam some solutions were proposed based on the reaction of the receipts. Here we will mention few of them. The basic nature of these solutions is that they do not use any technological tools to address the problem rather they demand's the users and companies to take actions that will terrify people from sending spam. Another important feature of these solutions is that they are most proactive in nature. They can achieve high popularity in the organizations whose most of the available bandwidth is

wasted on downloading spam messages. If proper awareness and devotion is created on the side of message users then these suggested solutions can have very good results.

1.11.1.1 Recipient Revolt

This solution suggests that on reception of any spam the user will react with anger in messages and in physical world. This solution helped significantly to scare more legitimate companies to keep themselves away from using junk messages and forced the Internet service providers (ISPs) to change policies.

Some of the advantages from this solution are:

- Forcing ISPs to change policies.
- Legit companies will be afraid to spam resulting in the removal of message ids from their contacts.
- If it gains momentum then it will be having a nice positive feedback. The fewer spams the more effort can be spent on punishing them.

Some of the disadvantages of the solution are:

- Burden on ISPs for handling valid and invalid complaints.
- Authentication of complaints so that complaints are checked that they are against the right person.
- As spammers hide their identities, it will cause some people to block all messages from unknown persons. The result will be hurdles and limited range of communication.

1.11.1.2 Customer Revolt

Most of the spams contain advertisements of different sorts from companies. To deal with it this solution suggests that companies to which the users submit their data should be forced to disclose what they will do with that data and should stick to whatever they claim. There should be proper publishing of policies on the web pages, mentioning the purpose of data gathering. The disadvantages of this solution are:

- There may be false complaints.
- The burden of separating valid from invalid complaints.

1.11.1.3 Vigilante Attack

This solution suggests that spam addresses should be deal with anger and should be treated with messages bombs and denial of service attacks. Though it will make spammers to think before sending spam but sometimes an innocent might be a victim claiming that he is spammer. Some of the disadvantages of this solution are:

- Identification of spammer is very important for this kind of solution which is a hard job.
- The results of this solution might be nasty in some cases.

1.11.1.4 Contract-Law and limiting trial accounts

This solution requires an agreement between the user and the organization which provide the messaging facility. The user should sign a proper agreement before get the registration. Sufficient information should be gathered regarding the user to know his identity. The account should be on trial basis. After passing the trail successfully i.e. without being reported to have send spam, his account will get registered fully. If found violating the laws at any stage, his account will be abundant and should be punished. While this solution looks quite attractive but the big hurdle in its implementation is the disclosure of people's identity without their will to the organizations which might not be acceptable to many users.

1.11.2 Technical Solutions

The technical solutions are mostly reactive in nature i.e. once the spam is present at the user account then techniques are used to eliminate the spam. These solutions do not force spammers not to send spams, rather they work towards making the job of spammers hard. As more and more, the internet community learns about the problems of spams, the more proactive technical solutions we can expect. At the present moment, researchers have not

concentrated on the proactive solutions greatly. In the preceding subsections a brief over view of technical solutions are presented.

1.11.2.1 Domain filters

Programs are configured in such a manner that they only accept messages from specific domains. Messages whose domains are not mentioned will not be received. This way a lot of spam is blocked. The major disadvantages are:

- Spammers will start using the valid domains.
- Communication range is narrowed down.

1.11.2.2 Blacklisting

It filters out unknown addresses and maintains databases of known abusers thus eliminating spams from them. Servers are placed in distributed manner which constantly monitors the communication of users and try to figure out spammers and their sites. Though it can be help full in some cases but again innocent users might be caught as spammers. Some of the disadvantages are:

- The overhead of maintaining the database about the spammers.
- Constant updation of the databases and retrieval of information from the distributed database about the spammers.

1.11.2.3 White list Filters

Programs are configured to learn all contacts of a user and allow messages from those contacts only. Messages from strangers are put into other folders thus eliminating the chances of spam to be present at user inbox folder. Disadvantages of the solution are:

- Configuration of the programs to learn about contacts of the users. .
- New parties message might be delayed as they are not directly visible to the user because of not being present at the inbox.
- Overall it will suffer from the limited range of communication and hurdles in communication.

1.11.2.4 Rules based

An expert examines Spam messages and then efforts are made to find word or phrasal relationships between message instances and it corresponding class. The relationships thus define are called rules. Many rules are combined in this way to make up the spam detecting solution. Certain weights will also be assigned to rules based on their utility towards the class definition. An unknown instance will be thus classified based on the absence or presence of certain predefined rules along with their weights in the message.

The disadvantage of this solution is the requirement of a human expert. Furthermore, rules might be outdated due to the spammer's knowledge about the solution. Thus changing the nature of the spams may lead to different relationship between textual message contents and its corresponding class. In such a challenging environment the needs of human expert will always be required to constantly update the system so that to cater for new kinds of spam.

1.12 Limitations in the Current Solutions and Proposed Alternative

All the solutions explained above have generally three major draw backs:

- Limited range of communication.
- Implementation hazards on users and companies sides.
- Expensive human resources requirements.

There exist an alternative to these solutions known as Automated Spam Filtering that will minimize the three drawbacks. The solution uses machine learning algorithms to learn from the previous data and then given an unknown instance it tries to predict its class from previously learned patterns. The benefit of this solution is that it will update its self and will learn automatically about new kinds of spam with minimum user input. The solution will treat the problem of spam detection as an instance of document classification problem. In the preceding section, a brief overview of the solutions is given.

Chapter 2

Literature Survey

SMS Spam Detection can be carried out using various methods and approaches. This chapter covers those methods and approaches. It also aims to review previous related work in spam detection and research areas in document classification.

2.1 Methods for SMS Spam Detection

There are 2 basic methods for detecting the SMS spams [14]:

- **Collaboration based method:** This method is based on the feedbacks from users and shared user experience. Collaborative filtering is a method of making automatic predictions (filtering) about the interests of a user by collecting preferences or taste information from many users (collaborating).
- **Content based method:** This method is focused on analyzing the textual content of messages. This is more popular due to the difficulty in getting access to the data about usage and user experience.

2.2 Approaches for Spam Detection

There are mainly two approaches for spam detection which are discussed below [15]:

2.2.1 Rule based Approach

Rule-based approach is used by creating rules to categorize the incoming messages. It is known as the direct approach. It does not require any training phase. Rules cover different threats, suspicious format and weak origin prone to sending spam means the sender is confirmed as open relay. While using this approach we have to be careful because rules generated by us can lead the incoming messages to misclassification. There is risk of false negative and false positive. Following steps are performed while we are using rule base approach for spam detection:

- **Pattern Analysis:** Pattern of spam and ham messages are analyzed from the set of a database of spam messages and ham messages. For pattern analysis, header and body of messages are searched for dubious keywords that are known to come in spam or ham messages. Analysis of spam messages also depends on the type of messages whether it is Adult, Product, Educational or Gambling Spam.

- **Pattern Selection:** Pattern is selected with the combination of several features. Selection of words/tokens with higher probabilities is done during this step. For the selection of spam like pattern, tokens with higher W/S (probability that the word is in spam messages) value are selected. Along with probability of words header of the message is also considered. Overall pattern selection includes selection of words, subject and the field of the message.

- **Score Assignment:** Score can be assigned using two ways: one it can be assigned by making calculations using score learning tool and in the other way, rule maker assigns score of its own choice. When we are assigning score with our choice, then we must be more careful. Wrong assignment of scores can lead to misclassification.

2.2.2 Learning Approach

This approach deals with training of Spam filter. A large set of ham messages and spam messages is used to train the spam filter. In training filter reads tokens from messages and adjust the values of tokens/words in the database according to their category whether they are from spam message or ham message. In order to reach maximum accuracy and generalization capabilities classifiers must extract only pertinent information from the training data. It is basically of two types [16]:

- **Supervised Machine Learning:** Supervised feature selection works like determining feature relevance after evaluating features' correlation with the class. In supervised learning the target concept is related to class affiliation. For this learning, different classifiers are available.

- **Unsupervised Machine Learning:** Unsupervised feature selection exploits data variance and separability to evaluate feature relevance when without labels. Here, the target concept is usually related to the innate structures of the data. Different clustering techniques are available for this learning.

2.3 Data Mining Approaches

Two common data mining techniques for finding hidden patterns in data are [17]:

- **Clustering analysis:** An automated process to group related records together is known as Clustering. On the basis of similar values for attributes related records are grouped together. In clustering analysis approach, it is not necessary that the end-user/analyst may specify beforehand how records are supposed to be related together. The objective of the analysis is, in fact, mostly to discover clusters or segments and then inspect the values and attributes that define the clusters.

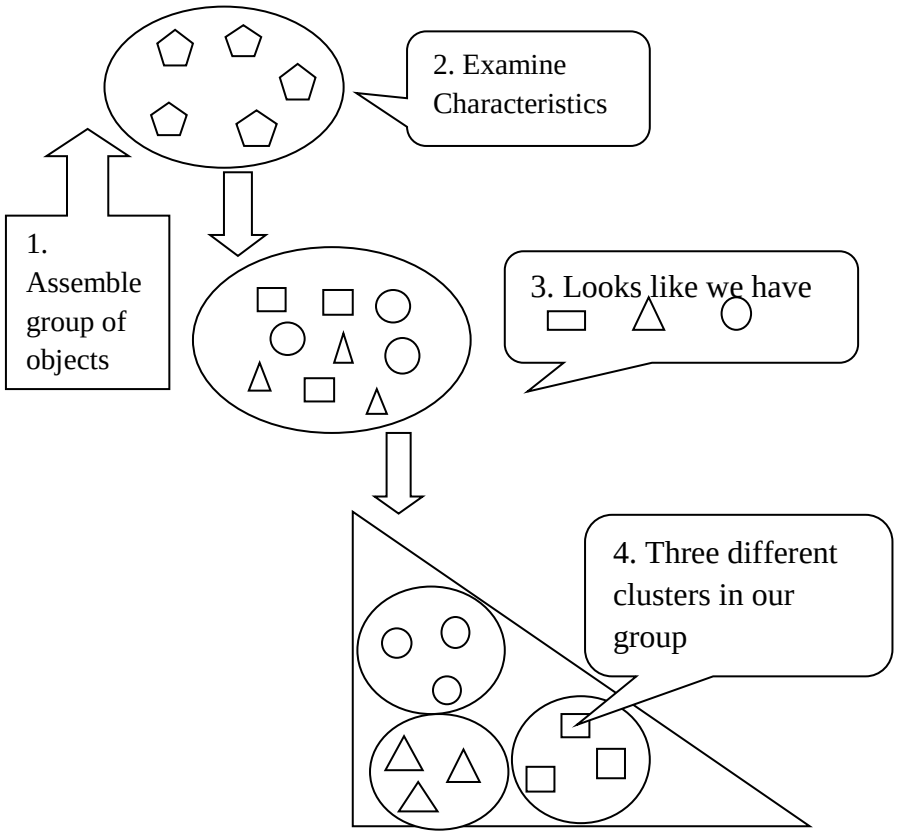


Figure 2.1: Clustering [18]

- **Classification analyses:** Classification is identical to clustering as it also classifies customer records into distinct segments called ‘classes’. However, unlike clustering a classification analysis requires that the end-user/analyst know ahead of time how classes are defined**.**

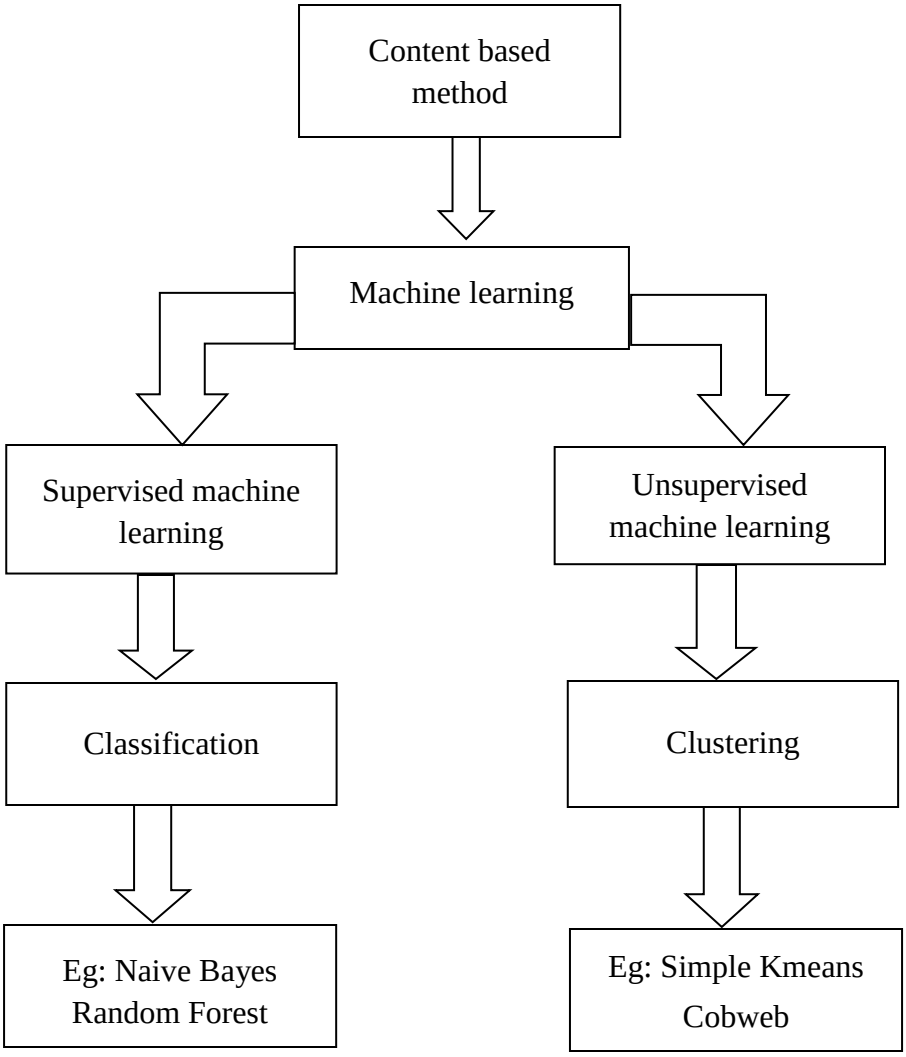


Figure 2.2: Learning Approach of Spam Detection

2.4 Spam as a Document Classification Problem

Automated spam filtering can be considered as a simple instance of document classification. In document classification problems, we have two sets of documents. The first document set has a predefined class and is known as the training set of documents.

This document set is used by the classifier to learn patterns in the data. The second documents set do not have the class labels with it and is used for the testing purpose. These documents set constitute all examples from the real world which will be given as input to the classification algorithm to classify later on. The problem of spam detection is necessarily the same with as that of document classification with two classes i.e. Spam and Legitimate. The job of our filtering process is to take messages as inputs and tries to learn about patterns that will represents different classes. Once the learning is done, then given an unknown instance of message it should be able to filter out spam with high accuracy. The Problem of document classification has been illustrated in figure 2.3.

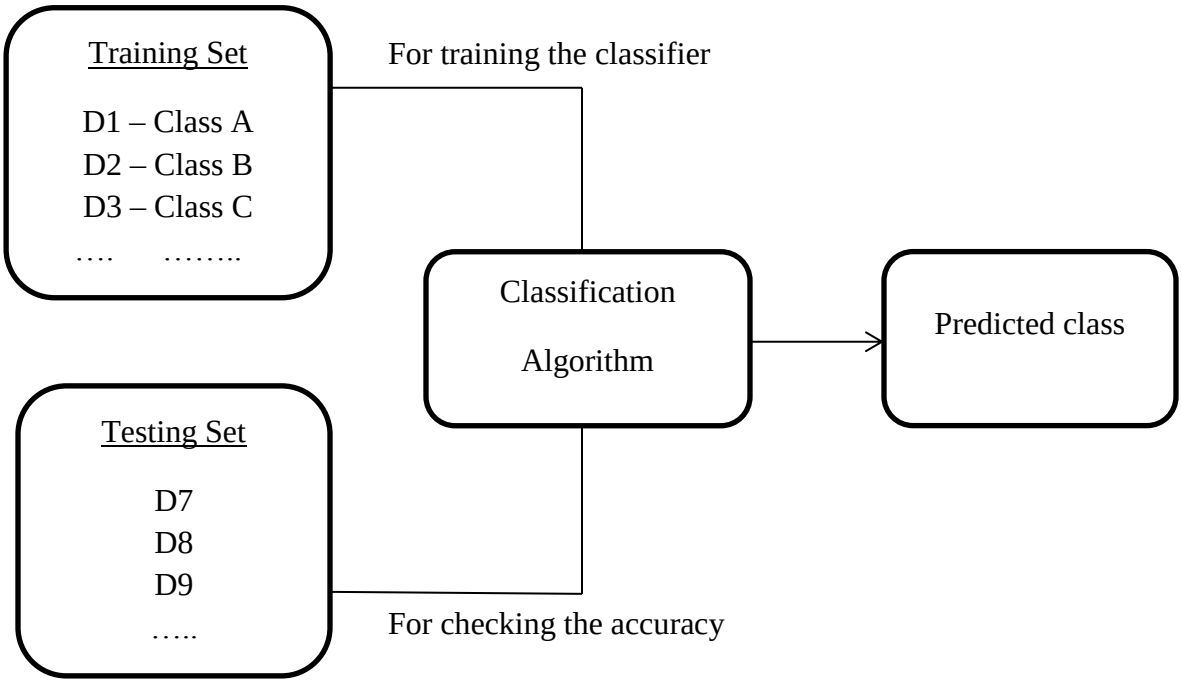


Figure 2.3: Problem of Document Classification

Document classification has a wide range of application and is fundamental task in information retrieval. As more and more textual information is available on the internet, its effective and fast retrieval is very important. Treating every web page as a document consisting of text will reduce the problem to document classification. Document classification is also used in organizing document for digital libraries. Other applications involve indexing, searching, web sites filtering, and hierarchical categorization of web pages.

2.5 Classification Algorithms

Systems that construct classifiers are one of the commonly used tools in data mining. Such systems take a collection of cases as input, each belonging to some class. Input is described by its values for a fixed set of attributes. Best output of a classifier can be judged by the accuracy of the prediction, in which the class of a new case is predicted. Few of the generally used classifiers are:

- **Naive Bayes:** Naive Bayes is one of the utmost proficient and helpful inductive learning approach for machine learning and data mining. While doing classification, its reasonable presentation is amazing. The reason behind it is the conditional independence hypothesis on which it is based upon seldom exists in physical domain applications [19].

- **Support Vector Machine:** SVMs attain significant enhancements over the presently finest performing approaches and perform stoutly over a diversity of various learning errands. Additionally they are entirely involuntary, eradicating the requirement for labor-intensive factor [20].

- **Decision Tree:** Decision tree classification algorithms uses decision tree as a projecting prototype that do mapping between the executions about an object to the deductions about that same object's target value. It is a prognostic modeling method that is used in statistics for machine learning [21].

- **Random Forest:** Random forests are an amalgamation of tree prognosticators in which every tree rest on the assessments of a haphazard vector sampled autonomously and with that same allotment for every single one tree of the forest. As the number of trees in the forest increases, generalization error for forests goes towards unification [22].

- **Adaboost:** Adaboost approach works good for the higher noise levels and it seldom over fits for low noise regime. Central to the understanding of this fact is the margin distribution. Adaboost can be considered as a restraint gradient depreciation in an error function with respect to the same margin [23].

2.6 Research Areas in Document Classification

A faithful representation of a document that is based on a sequence of words implies high dimensionality since the number of distinct words in Document can be very large even if a document is of moderate size. Dimensionality reduction methods are used to deal with this problem.

Detailed research in the field of document classification has revealed the following areas of concern [24]:

- **Statistical sparseness**

  High dimensional data are inherently sparse. Although the number of possible features (words) can be very large, each single document usually includes only a small fraction of them. Stemming algorithms can be used to reduce the sparseness of the data.

- **Domain knowledge**

  Since documents are given in the natural language, it appears that linguistic studies should help in discovering their inner structure, and therefore in understanding their meaning. Help of domain experts are usually used to sort out this problem.

- **Multi-Labeling**

  In the multi-labeled version of document classification, a document can belong to multiple classes simultaneously. In our case of spam detection we can say a SMS message which is controversial and is considered as spam and legitimate at the same time. In case where each document has only a single label we say that the categorization is uni-labeled document classification problem

2.7 Related Work

Gomez Hidalgo *et al.* (2006) had done a milestone work to detect mobile phone spam and assessed several Bayesian based classifiers [25]. In this work, the first two well-known

SMS spam datasets, namely, the Spanish and English test databases were proposed by the authors. A number of message portrayal methods and machine learning approaches were tested by the authors on those two datasets. They came up with the conclusion that Bayesian filtering techniques can be adequately employed to classify SMS spam.

Cormack *et al*. (2007) evaluated that even content-based spam filtering cab be used for short text messages which occurs in three diverse perspectives: SMS, blog comments, and email summary information [26]. Authors concluded that SMS are restricted to have insufficient words for the proper support of words or word bigram based spam classifiers. Thus the filter's efficiency was improved by expanding the set of features to include orthogonal sparse word bigrams and also to include character bigrams and trigrams. They implemented DMC, a compression model based classifier, which does not rely on explicit featurization and performed well on short messages and message fragments.

Nuruzzaman *et al.* (2012) looked over the efficiency of sieving message spam on independent cellular phones using Text Classification approaches [27]. On an independent mobile various processing were done related to training, filtering, and updating. Their established outcomes display that the projected model was successful in distilling messages hams and spam with reasonable efficiency, less storage consumption, and appropriate processing time without taking the help from a machine.

Coskun and Giura (2012) gave a network-based online detection technique for the identification of SMS spams campaign by taking the calculation of number of messages which were sent in single network over a small period of time and carry similar sort of data [28]. The approach given by them involved Bloom filters to keep a tentative count of message content occurrences.

Sarah Jane Delany *et al.* (2012) have worked on a clustering experiment on a SMS corpus [29]. To access the behavior of SMS spam, they compiled 1353 spam messages and tried to use it as the dataset which comprehended of no duplicity. They applied k-way spectral clustering with orthogonal initialization. By applying spectral clustering on their own compiled dataset few clusters were produced which were ten in count with their linked top 8 terms and a presumed annotation.

Tiago A. Almeida *et al.* (2013) showcased the particulars of a new authentic, open and non-encoded SMS spam compilation which constitutes of maximum number of messages [30]. It is composed by 4,827 mobile ham messages and 747 mobile spams. Furthermore, the authors performed several established machine learning algorithms on their dataset and they came to the conclusion that according to them SVM is a better approach for advance evaluation as it achieved good accuracy.

Houshmand Shirani-Mehr (2014) applied different machine learning algorithms to SMS spam classification problem, compare their performance to gain insight and further explore the problem, and design an application based on one of these algorithms that can filter SMS spams with high accuracy [31]. They used a database of 5574 text messages.

Chapter 3

Problem Statement

This chapter includes the problem statement and objectives of the thesis.

3.1 Problem Statement

Today network security is more complex compared to earlier. Frequent users of text messaging began to see an increase in the number of unsolicited commercial advertisements being sent to their mobile phones through text messaging. This can be particularly annoying for the recipient because unlike in email some recipients may be charged a fee for every message received. SMS Spam is an emergent problem for the mobile phone users. The recent increases in the spam rate had caused a great concern among the community. There are many techniques to deal with this spam problem using different kinds of Spam filters. Basically all these filters classify the messages in to the category of Spam and Ham (non-Spam). Most of the classifiers decide fate of an incoming message on the basis of some words in data part and categorize it. There are two parts, known as test data and training data, that work as the database for the Spam classifier to classify the messages.

The problem of spam has been addressed as a simple two class document classification problem where the main goal is to filter out or separate spam from non-spam. Since document classification tasks are driven by huge unproductive data, so selecting most discriminating features for improving accuracy is one of the main objectives and this thesis work concentrates on this task. Classification in the reduced dimensions will not only save us time but also will have lesser memory requirements.

The primary aim of this work is to concentrate on different classification techniques and to compare their performances on the domain of spam messages detection. A number of pre-classified messages were processed with the techniques to see which one is most successful and under which set of features. Secondary aim of the thesis is to work towards finding the best couple of featured reduction technique and classification

algorithm and to increase the earlier performance. Though it is a hard job as hundreds and thousands of such couples exits but this work can be considered as a step towards that goal. Apart from these aims we have also tried to alter the previously available global messages dataset according to the Indian messages and analysed the results for the Indian dataset as earlier there was no dataset containing Indian messages.

3.2 Thesis Objectives

Since SMS spam filtering is about finding the best feature to implement according to the classifier, there is a possibility to find such a couple which gives better accuracy than the existing results. Whole work of this thesis has been done to achieve the following objectives:

1. Preparing a dataset for Indian SMS messaging market.
2. Finding the best couple of featured reduction technique and classification algorithm and to increase the accuracy of existing system.
3. Design and development of SMS Spam Detection System.
4. Verify and validate proposed system.

Chapter 4

Implementation

This chapter comprise automatic SMS spam detection system, datasets used for the process and its methodology. For relevant information retrieval, the textual data of datasets was passed through pre-processing steps. The pre-processed data is then represented in numeric or vector form using suitable representation. Thus the data obtained after pre-processing is reduced. The reduced data is then passed onto classifier to learn the patterns in the data and filter them as spam or ham.

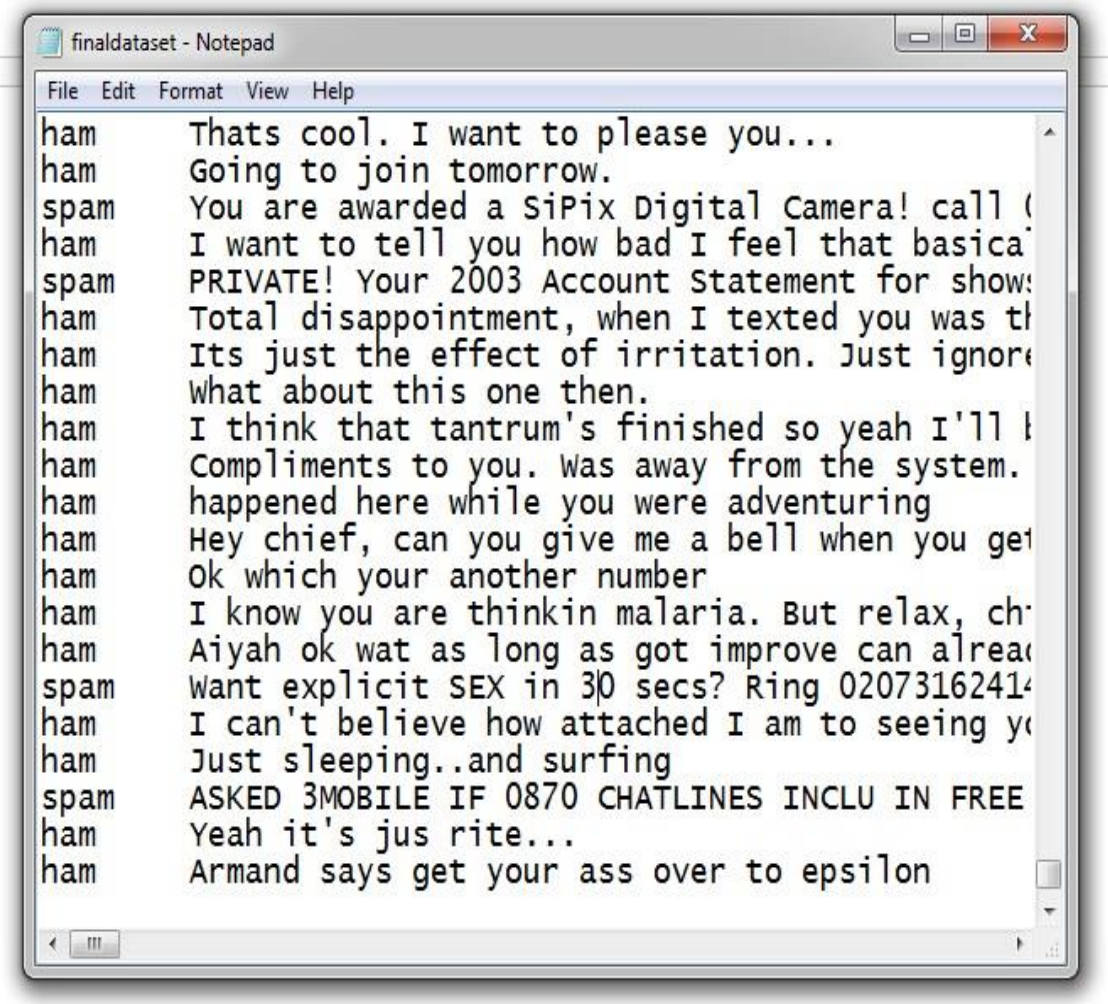
4.1 Datasets Used

In our experiments we have incorporated two datasets, one which was given by T. A. Almeida *et al.* which is available online for research sources and is used universally. The second one is an altered version of the same dataset, in which we have included 1187 Indian messages. As there was no dataset available with the Indian messages, so we tried to make a contribution in this field.

4.1.1 Dataset I: SMS Spam Collection Data Set by T. A. Almeida *et al.*

Since cellular messages repeatedly have a number of acronyms, it affects the efficiency of the filters. So a big and good message dataset is used in this process. This dataset is filled with spam messages mined from GrumbleText and WhoCallsMe websites and ham messages from the SMS Spam Collection. T. A. Almeida *et al.* [10] added a collection of 425 SMS spam messages physically obtained from the GrumbleText Web site. They have also included a subset of 3,375 legitimate messages arbitrarily selected from the NUS SMS Corpus which is a dataset of about 10,000 legitimate messages collected from volunteers, mostly Singaporeans and students attending the University, who were made aware that their contributions were going to be made publicly available. They have inserted 450 SMS ham messages collected from Caroline Tag's PhD Thesis. Finally, they unified the SMS Spam Corpus v.0.1 Big. It consists of 1,002 SMS legitimate messages and 322 spam messages. Hence the dataset we incorporated for the experiment is

composed by 4,827 ham messages and 747 mobile spams and its snapshot is shown below in snapshot 4.1.



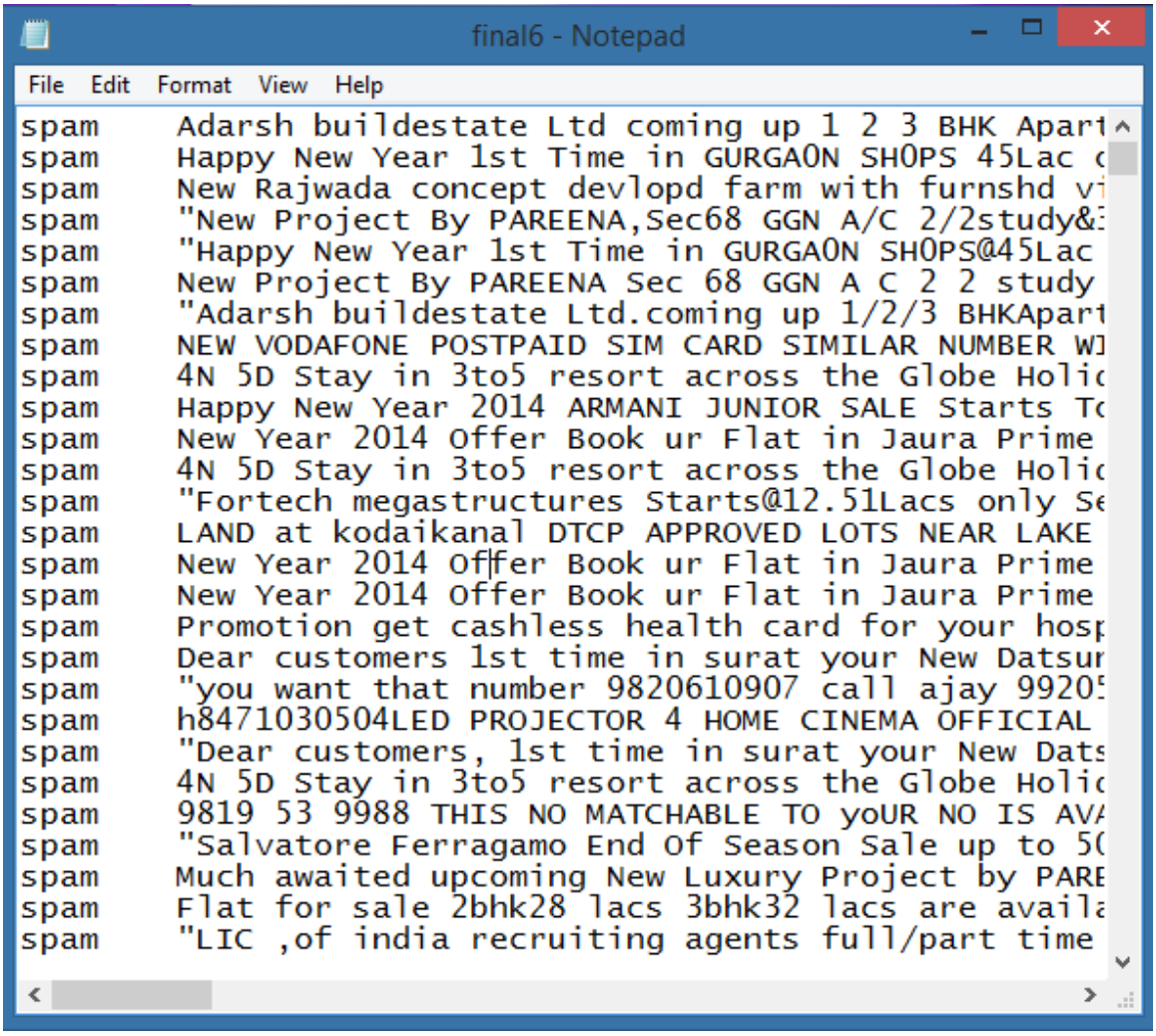
Snapshot 4.1: Dataset I

Table 4.1 Data division of Dataset I

Messages	Amount	Percentage
Hams	4,827	86.6
Spams	747	13.4
Total	**5,574**	**100**

4.1.2 Dataset II: SMS Spam Collection Dataset including Indian content

The dataset provided by T. A. Almeida *et al.* is already composed of 4,827 ham messages and 747 mobile spams, here we tried to convert the same dataset for Indian market by adding Indian spams and hams. We added 439 legitimate messages and 748 spams from Indian perspective. Legitimate messages are collected from from volunteers, students attending the Thapar University, who were made aware that their contributions were going to be made publicly available. The spam messages which we have used are provided us by a Mobile Network Operator company.



Snapshot 4.2: Dataset II

Table 4.2 Data division for Dataset II

Messages	Amount
Hams	4,827
Indian hams	439
Spams	747
Indian Spams	748
Total	**6761**

So earlier there were 5,574 total messages in the dataset but now after adding 1,187 Indian messages, dataset get increased to 6,761 in count.

4.2 Methodology

This section describes the general design of workflow of the experiment. In this experiment machine learning tool is used for the analysis and classification of the dataset. At the first level data is gathered from different sources to create a good dataset of ham and spam in text format and give that data as the input in model. At the second level of the experiment we converted the data set which is earlier in the text format to CSV (Comma Separated Value). Then preprocessing is done for a better quality input either by removal of unrequired words or by performing stemming on them. Then the pre-processed data is transformed into a machine readable form or non-contextual form by converting to vector or by doing discretization. The labeled data is opened and the attributes are listed. The attributes that are used for the analysis purpose are text and class in this dataset. After that, a classifier is applied to the data set we have used. Thus, the data is trained using the dataset. Testing is performed on the testing data to get the final results. Finally at the last step of the experiment, Confusion Matrix and Receiver Operating Characteristic (ROC) are obtained from the dataset and the results of the applied classifier are analyzed and discussed.

The spam detection algorithm used in our research work is shown below.

4.2.1 Spam Detection Algorithm

1. Arrange the data into set of training and testing examples.
2. N = Number of messages instances in the dataset.
3. M = Number of testing examples.
4. For I = 1 to N
    i. Pick up message instance I.
    ii. Remove the list of stop words from message I.
    iii. Perform Stemming on message instance I.
    iv. Convert the text message into vector form.
    v. Train the system using classification technique.
6. For J = 1 to M
    i. Pick up the testing example number J.
    ii. Perform pre-processing.
    iii. Convert the text message into vector form.
    iv. Use the trained system to classify the example.
    v. Store the results of classification i.e. its accuracy and other evaluation measures.
7. Take an average of the evaluation measures to reflect the performance over the entire set of testing examples.

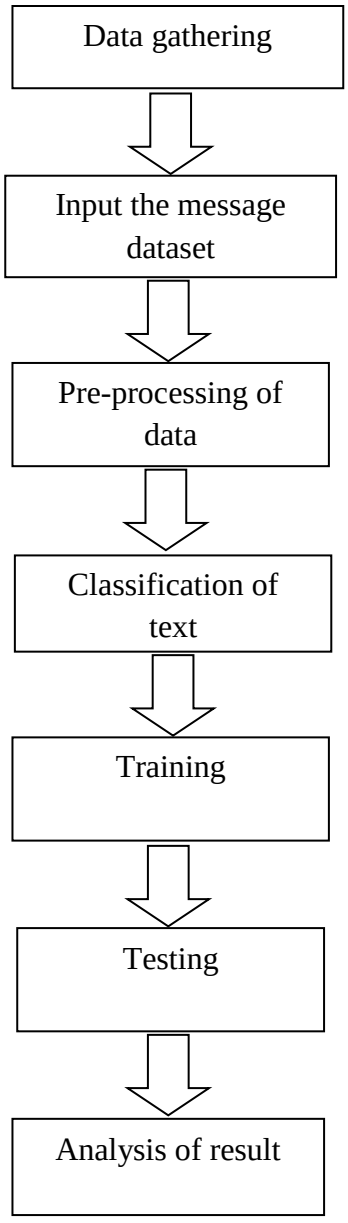


Figure 4.1 Process of Spam Filtering

4.2.2 Pre-processing

In text retrieval tasks the pre-processing of the textual information is very critical and important. Main objective of text pre-processing is to remove data which do not give useful information regarding the class of the document. Furthermore, we also want to remove data that is redundant. Most widely pre-processing steps in the textual retrieval tasks are removing of stop words and performing stemming to reduce the vocabulary. In addition to these two steps, we have also removed the words that have length lesser than

or equal to two in certain cases. Next we are going to describe the pre-processing steps in detail.

4.2.2.1 Removal of Words Lesser in Length

Investigation of English vocabulary shows that almost all such words whose length are lesser than or equal to two contains no useful information regarding class of the document. Examples includes a, is, an, of, to, as, on etc. though there are words which have length of three and are useless like the, for, was, etc but removing all such words will cost us losing some words that are very useful in our domain, like sex, see, sir, fre (often fre is used instead of free to deceive the automatic learning filter).

All messages of the data set were passed through a filter which removed the words that have length lesser than or equal to two. This removed bundle of words from the corpus that were useless and reduced the size of the corpus to great extent.

4.2.2.2 Removal of Stop Words

In information textual retrieval there are words that do not carry any useful information and hence are ignored during indexing and searching. Stop terms definition in context of internet search engines is ‘words that is so common on the Internet that search engines ignore them. E.g. homepage, home page, www, Web, Web page, the, of, that, is and, to, etc. In terms of database searches it is defined as ‘words that databases will not search for’. In general and for document classification tasks, we consider them as words intended to provide structure to the language rather than the content and mostly include pronouns, prepositions and conjunctions [13]. There are various lists of stop words available on the internet but we have tried to use a list of maximum stop words for better results. In our set of experiments, the list of stop words contains 174 words, few of them are shown below in figure 4.2.

then, there, that, which, the, those, now, when, which, was, were, been, had, have, has, will, subject, here, they, them, may, can, for, such, and, are, but, not, with, your.

Figure 4.2: Stop Word List

4.2.2.3 Stemming

The second main pre-processing tasks applied in textual information retrieval tasks is the stemming. It can be defined as 'an algorithm developed to reduce a search query to its stem or root form, in other words, variations of particular words such as past tense and plural and singular usage are taken into account when performing a search. For example, applies, applying & applied matches apply'. In the context of searching it can be defined as "expansion of searches to include plural forms and other word variations". In the context of document classification we can define it to be a process of representing words and its variants with its root.

4.2.3 Representation of Data

The Next main task was the representation of data. The data representation step is needed because it is very hard to do computations with the textual data. The representation should be such that it should reveal the actual statistics of the textual data. Data representation should be in a manner so that the actual statistics of the textual data is converted to proper numbers. Furthermore it should facilitate the classification tasks and should be simple enough to implement.

The representation schemes considered in thesis were based on the words statistic. There are some representations schemes that work with the handmade phrasal statistics also [32]. We used the words statistics due to its simplicity and secondly as the instances of messages changes then using the predefined phrases would not have that much of the effect on accuracy. It should be noted that the words statistics completely ignores the context in which the word is used. It rather looks for just the occurrences of words in the message instances and forms those statistics as the basis for prediction. Considering the context of words require many natural language processing tasks to be performed and will increase the complexity of solution. Furthermore non contextual words statistics have been used over the years on document classification tasks with acceptable performances. That's why we also used the non-contextual representation of words. Next we will describe different representation schemes that have been used in the textual processing tasks.

4.2.3.1 Term frequency

Term frequency is also widely known as bag of words weighting and vector space model. It is also relatively simple weighting which counts the number of occurrences of term in a message. Mathematically it can represented as

$$Term\ Frequency\_W_{ij} = tf_{ij}$$

4.2.3.2 Term Frequency with Lengths Normalized

In order to cope with documents of different lengths a variant of term frequency is introduced. Here every weight of a term will be divided by the total number of terms frequencies in the message instance. Mathematically it can be represented as

$$TF\_normalized\_W_{ij} = \frac{tf_{ij}}{\sum_{k=1}^{M} tf_{kj}}$$

4.2.3.3 Term Frequency inverse document frequency

This is the most widely used weighting scheme. Term frequency and Boolean weighting do not take the global statistics of the term into account. As already established that those terms whose presence is in lesser number of messages can discriminate well between the classes, TFIDF representation takes this property coupled with term frequency to define a new weighting which can be expressed mathematically as

$$TFIDF\_W_{ij} = tf_{ij} \times \log(N/n_i)$$

4.2.3.4 Term Frequency inverse document frequency with lengths Normalized

To account for the documents of different lengths the weights obtained from the TFIDF are normalized. We performed few experiments with this weighting and got good results.

$$TFIDF\_Normalized\_W_{ij} = \frac{tf_{ij} \times log(N/n_i)}{\sum_{k=1}^{M} tf_{kj} \times log(N/n_k)}$$

4.2.4 Classification

The SMS messages instances after applying features will be provided as inputs to the classification algorithm. A classification algorithm can be defined as "A predictive model that attempts to describe one column (the label) in terms of others (the attributes). A classifier is constructed from data where the label is known and may be later applied to predict label values for new data where the label is unknown". In Simple terms, classification is a task of learning data patterns that are present in the data from the previous known instances and associating those data patterns with the classes. Later on when given an unknown instance it will search for data patterns and thus will predict the class based on the absence or presence of data patterns.

4.3 Technology Used

We have used Data Mining as the backbone technology for developing the SMS message spam detector. It is used to generate new information from the large pre-existing database. Figure 4.3 shows the architecture of data mining which is being used in our process.

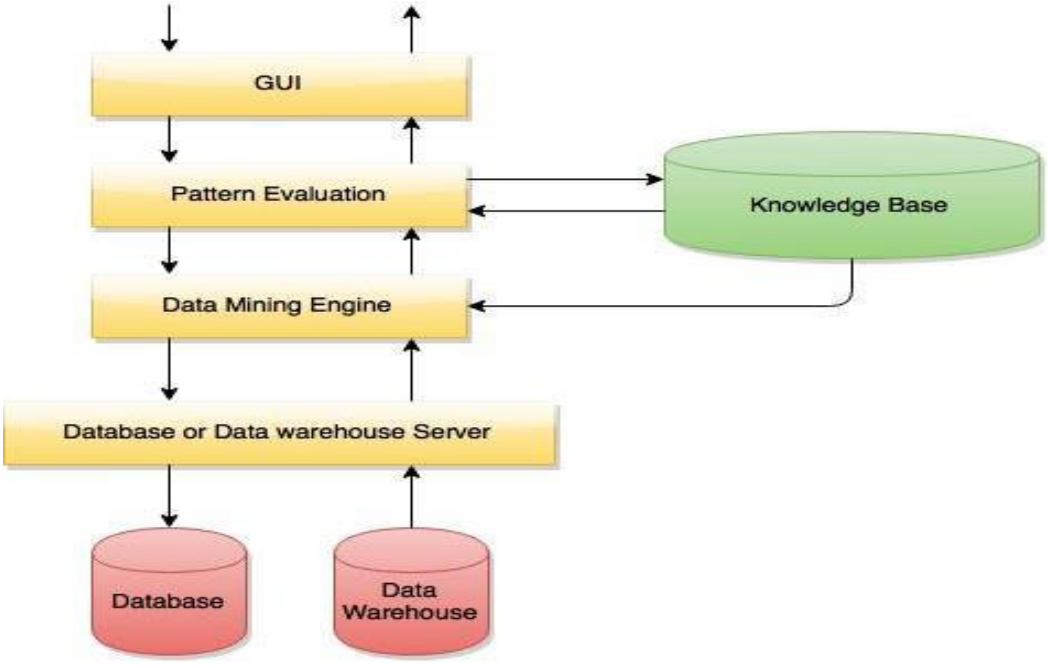


Figure 4.3: Architecture of Data Mining System

4.4 Proposed System

We have performed classification using *scikit-learn 0.16.0* library of Python. For creating an interface for users, we have used *tkinter* library of Python.

4.4.1 Architecture of the System

In figure 4.4, processing is carried out for two phases- training and testing. Raw SMS dataset is passed to training phase for creating the knowledge sources. Live messages are passed to testing phase for getting detected as ham or spam, using our proposed GUI (Graphical User Interface).

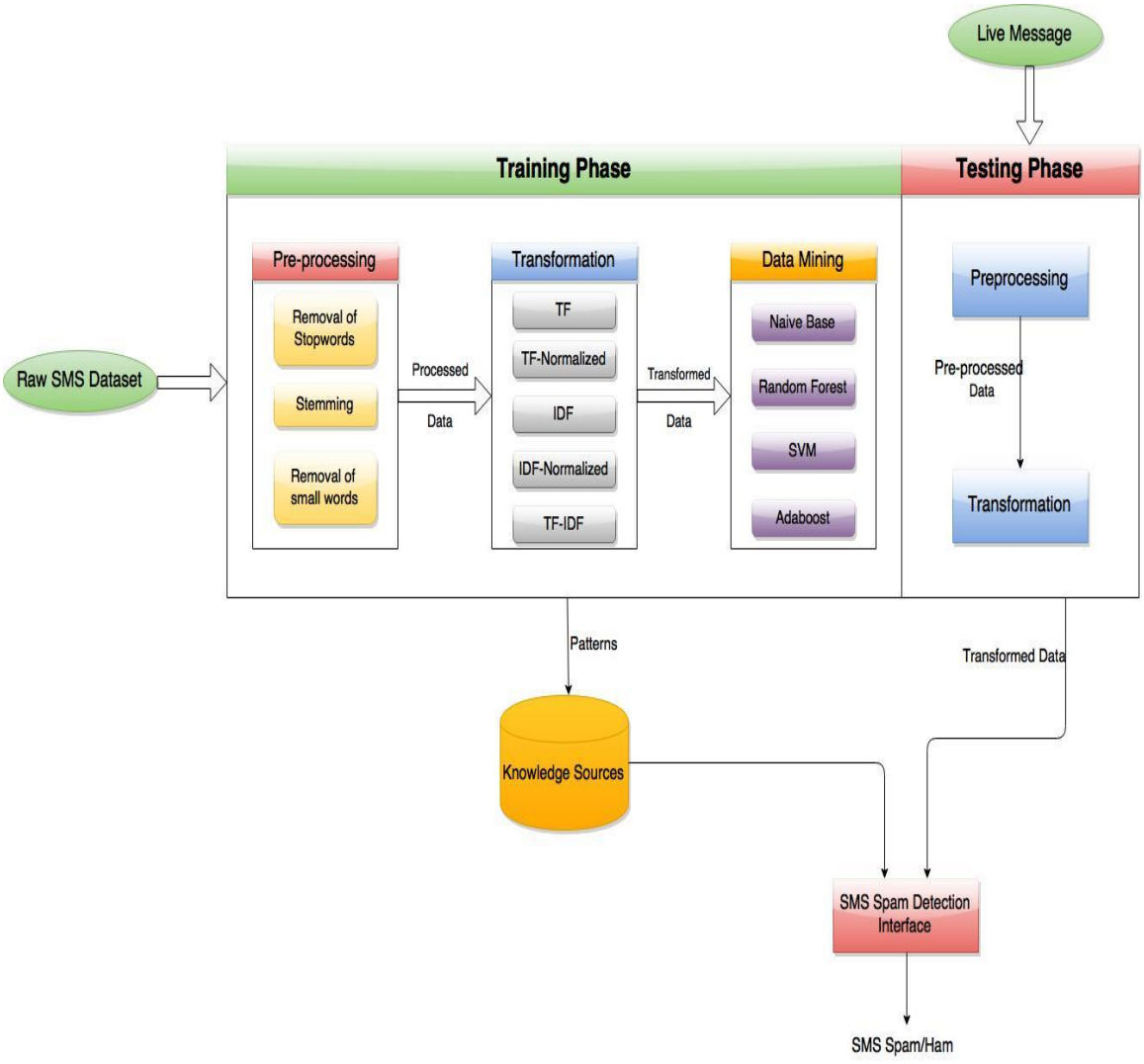


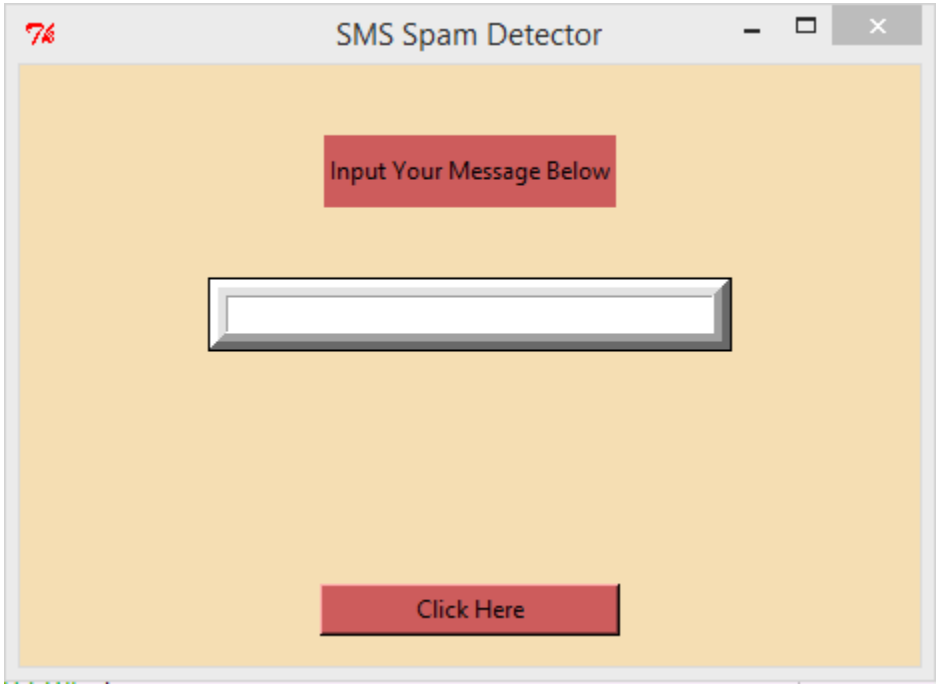
Figure 4.4: Architecture of Proposed System

4.4.2 Workflow of the System

The GUI is developed in Python and it satisfies the following characteristics of being a successful GUI:

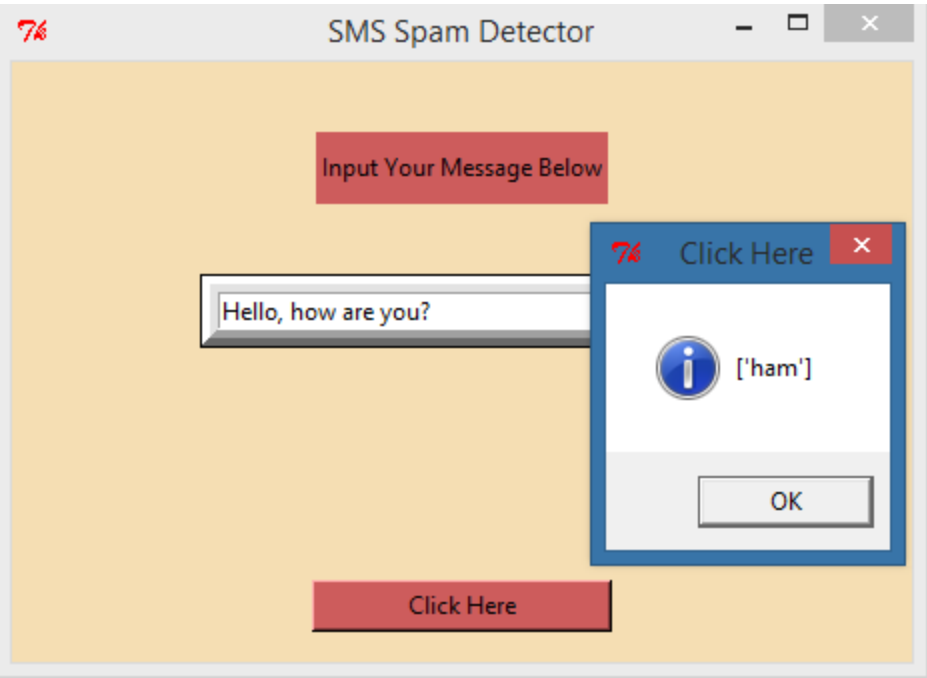
- Clear
- Concise
- Responsive
- Consistent
- Attractive
- Efficient

The System is named as SMS Spam Detector. In this system, message is given as input and a message box pops up as the output indicating, whether the input message is spam or ham.

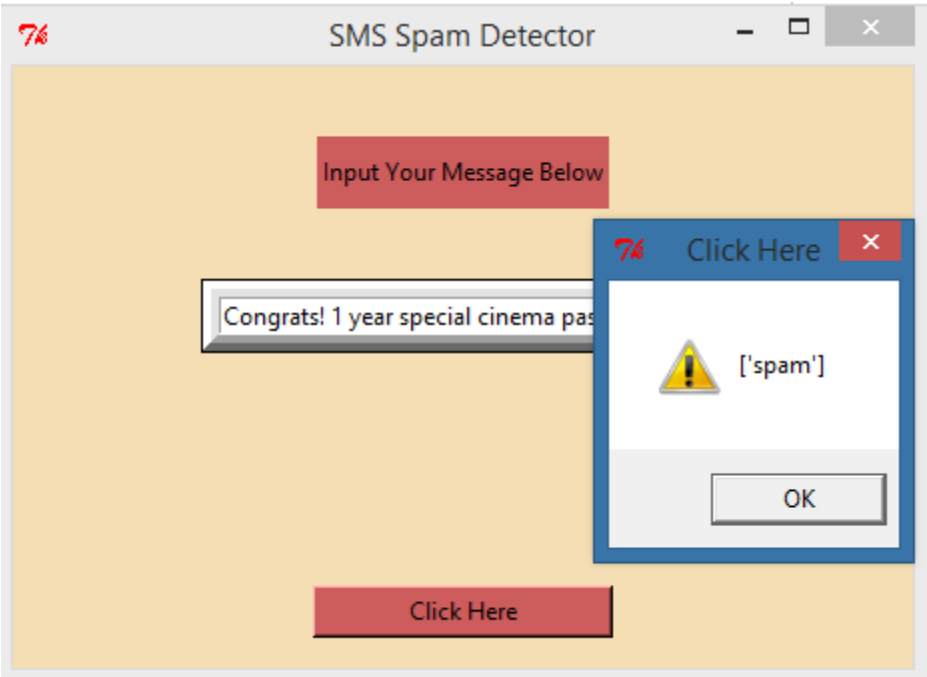


Snapshot 4.3: GUI for SMS Spam Detection

Snapshot 4.4 is displaying the pop up message for a ham SMS message and snapshot 4.5 is displaying the pop up message for a spam SMS message.



Snapshot 4.4: SMS Spam Detector for Ham message



Snapshot 4.5: SMS Spam Detector for Spam message

Chapter 5

Performance Evaluation

In this chapter, different classifiers are applied on both the datasets: the large corpus SMS Spam Collection Dataset created by T. A. Almeida et al. and SMS Spam Collection Dataset including Indian content. By applying different classifiers the best and the worst classifier can be judged, based upon the different evaluation metrics. From the statistics while applying single classifier at a time on both the datasets, Support Vector machine classifier gave the best accuracy among different applied classifiers. Multinomial Naive Bayes gave the second best accuracy, additionally it had the least time consumed while performance.

5.1 Evaluation Metrics

The metrics measure the percentage of Spam detected by the system and how many misclassifications it makes. Few of the evaluation metrics are:

- **True Positive (TP):** When positive instances are correctly classified it is reported by a number called True positive.
- **False Positive (FP):** When positive instances are incorrectly classified it is reported by this number called False positive.
- **False Negative (FN):** When negative instances are incorrectly classified it is reported by this number.
- **True Negative (TN):** When negative instances are correctly classified it is reported by this number.
- **Correctly Classified Instances:** It is the summation of True Negative (TN) and True Positive (TP).
- **Incorrectly Classified Instances:** It is the summation of False Positive (FP) and False Negative (FN).

The above mentioned basic metrics were used for further calculations of various metrics. For evaluating the performance of spam detection system, we measured accuracy (ACC),

Spam Caught (SC), Blocked Hams (BH), Matthews Correlation Coefficient (MCC), F-measure, precision, recall, Area Under the ROC curve (AuC), True Positive Rate (TPR) and time consumed by the classifier.

- **Accuracy (ACC):** Accuracy can be defined as the proportion of correctly classified classes namely True Positive and True Negative over the total number of classifications as depicted by formula below:

$$Accuracy = \frac{(TN + FP)}{(TN + TP + FN + FP)} \times 100$$

- **Precision (P):** Precision is the fraction of the messages retrieved that are relevant for the user.

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

- **Recall (R):** Recall is the fraction of the successfully retrieved messages that are relevant to the query.

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

- **Matthews Correlation Coefficient (MCC):** MCC is used to determine the quality of classification for two classes even when size of the classes varies. It ranges between -1 and +1 where +1 representing the finest performance.

$$MCC = \frac{(TP\ \times\ TN)\ -\ (FN\ \times\ FP)}{\sqrt{(TP\ +\ FP)\ \times\ (TP\ +\ FN)\ \times\ (TN\ +\ FP)\ \times\ (TN\ +\ FN)}}$$

- **F-Measure:** It serves as the harmonic mean of precision and recall.

$$F - measure = 2\ \times \frac{P\ \times R}{P + R}$$

- **Spam Caught (SC):** It is the ratio of number of caught spams to the total number of spams exists in the dataset.

$$SC = \frac{False\ negative\ data}{Total\ number\ of\ spams}$$

- **Blocked hams (BH):** It is the ratio of number of hams that are detected as spams to the total number of hams exists in the dataset.

$$BH = \frac{False\ positive\ data}{Total\ number\ of\ hams}$$

- **Area under Curve (AuC):** A Receiver Operating Characteristic (ROC) curve is a graphical plot which is used to represent the performance of a binary classifier system. It is plotted using the values of True Positive Rate on the vertical axis and False Positive Rate on the horizontal axis of the curve. The area covered under this curve is known as Area under Curve (AuC).

5.2 Experiment I: Classification performed on Dataset I

Following classifiers were applied on the above mentioned dataset and results were observed as follows:

5.2.1 Multinomial Naive Bayes

Multinomial Naive Bayes classifier was applied on the dataset after various feature selection and extraction. Count vectorizer, Term Frequency (TF) and Inverse Document Frequency (IDF) are the basic features for this dataset along with the various features applied. To divide the dataset in an appropriate ratio for training and testing plays a significant role in the process. Sometimes Cross Validation (CV) may yield a better result, according to the applied classifier. After applying various features different results were obtained and they are tabulated in table 5.1. Naive Bayes due to its speed as well as efficiency turns out to be the most effective classifier for practical implementation.

Table 5.1: Dataset I: Results of MNB

Features	Accuracy	Time-seconds
Count vectorizer + 70 :30	0.98684997	0.546999931
Count vectorizer + 70 :30 + Stopwords	0.984459056	0.765999794
Count vectorizer + 75 :25	0.985652798	0.582999945
Count vectorizer + 75 :25 + Stopwords	0.984218077	0.57799983
Count vectorizer + 80 :20	0.984753363	0.906000137
Count vectorizer + 80 :20 + Stopwords	0.986547085	0.832000017
Count vectorizer + crossvalidation (3 folds)	0.979727305	5.188999891
Count vectorizer + crossvalidation (3 folds) + Stopwords	0.977574453	4.837000132
Count vectorizer + crossvalidation (5 folds)	0.98026486	8.052999973
Count vectorizer + crossvalidation (5 folds) + Stopwords	0.979547049	7.921999931
Count vectorizer + crossvalidation (10 folds)	0.980805033	16.18799996
Count vectorizer + crossvalidation (10 folds) + Stopwords	0.979370052	13.96399999
TFIDF + 70 :30	0.954572624	0.29700017
TFIDF + 70 :30 + Stopwords	0.968320383	0.296999931
TFIDF + 75 :25	0.956958393	0.312999964
TFIDF + 75 :25 + Stopwords	0.972022956	0.356999874
TFIDF + 80 :20	0.961434978	0.351999998
TFIDF + 80 :20 + Stopwords	0.969506726	0.368999958
TFIDF + crossvalidation (3 folds)	0.954969501	3.070999861
TFIDF + crossvalidation (3 folds) + Stopwords	0.964657338	3.046999931
TFIDF + crossvalidation (5 folds)	0.958915851	4.904000044
TFIDF + crossvalidation (5 folds) + Stopwords	0.970575858	4.809000015
TFIDF + crossvalidation (10 folds)	0.963224259	9.320999861
TFIDF + crossvalidation (10 folds) + Stopwords	0.974705397	10.31400013

Results above accuracy of 97% are graphically shown in Figure 5.1 for a better observation. Our best result with Naive Bayes turned out when it was implemented with multinomial model and Laplace smoothing. It shows the 98.68% of ACC, 93.85% of SC and 0.55 % of BH. Area under curve (AuC) for the same was covered 96.65% is shown in Figure 5.2.

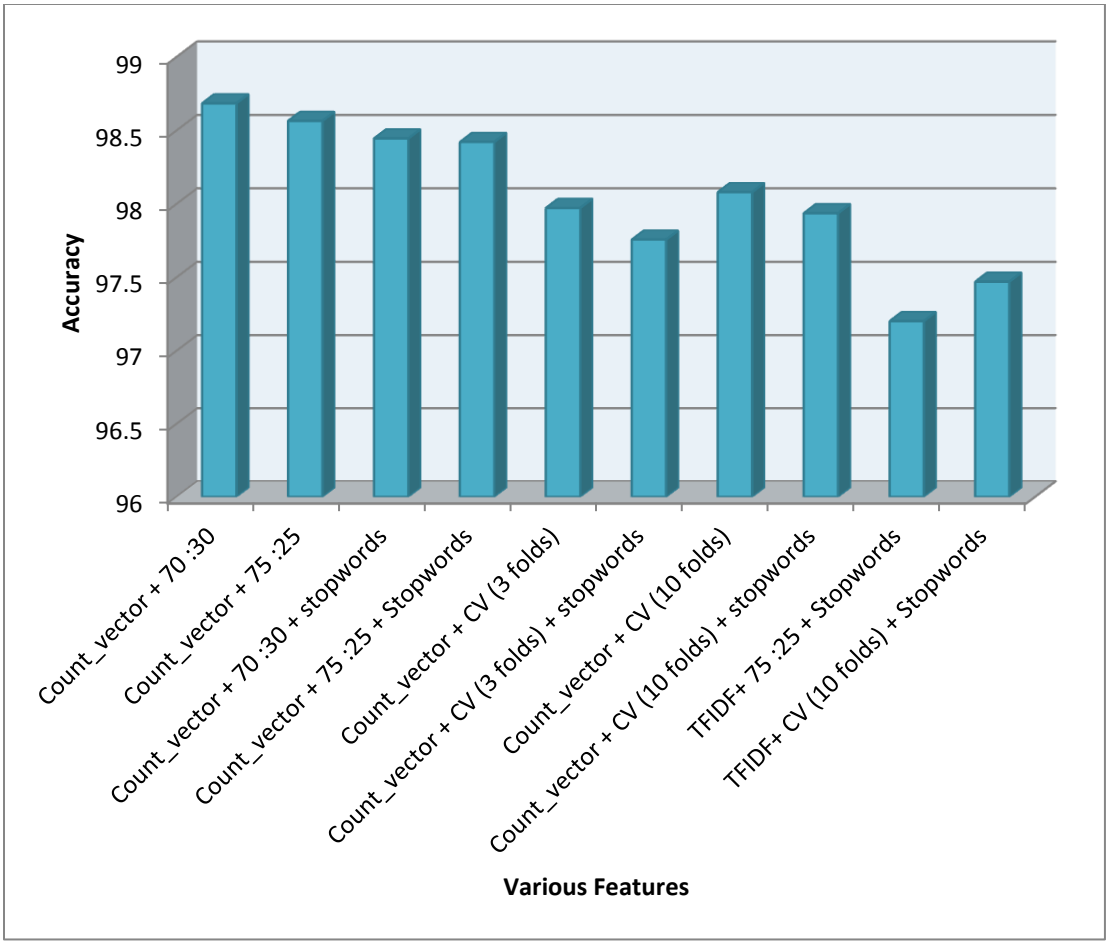


Figure 5.1: Dataset I: Comparison of various features with MNB

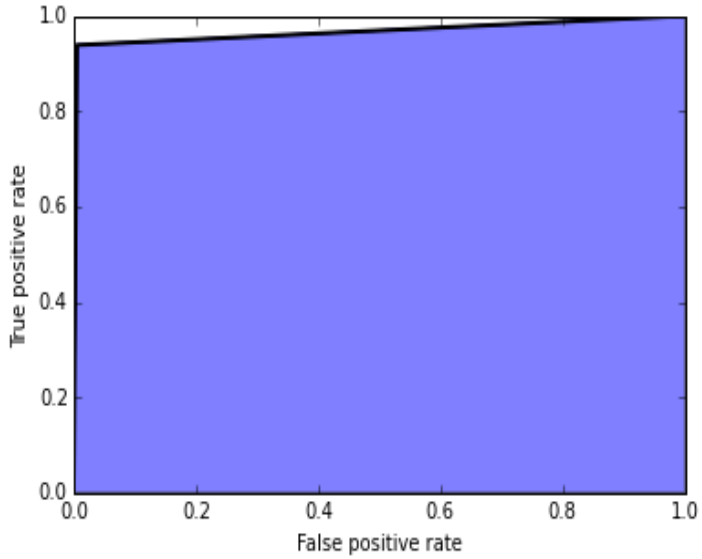


Figure 5.2: Area under curve for best result of MNB - 0.96

5.2.2 Random Forest

Random Forest is an averaging ensembling technique of classification which ensembles decision trees built from the training set. Different results of Random forest classifier were obtained for various feature extractions and they are tabulated in table 5.2.

Table 5.2: Dataset I: Results of Random Forest

Features	Accuracy	Time – seconds
Count vectorizer + 70 :30	0.966527197	6.72300005
Count vectorizer + 70 :30 + Stopwords	0.962343096	8.30099988
Count vectorizer + 75 :25	0.964849354	6.95600009
Count vectorizer + 75 :25 + Stopwords	0.968436155	9.748000145
Count vectorizer + 80 :20	0.972197309	7.201999903
Count vectorizer + 80 :20 + Stopwords	0.964125561	10.98099995
Count vectorizer + crossvalidation (3 folds)	0.967348403	34.171
Count vectorizer + crossvalidation (3 folds) + Stopwords	0.967527808	47.51300001
Count vectorizer + crossvalidation (5 folds)	0.968960059	57.3440001
Count vectorizer + crossvalidation (5 folds) + Stopwords	0.968242891	84.47200012
Count vectorizer + crossvalidation (10 folds)	0.970756298	121.852
Count vectorizer + crossvalidation (10 folds) + Stopwords	0.973804827	161.6099999
TFIDF + 70 :30	0.970711297	5.860000134
TFIDF + 70 :30 + Stopwords	0.962940825	14.68400002
TFIDF + 75 :25	0.969870875	6.437000036
TFIDF + 75 :25 + Stopwords	0.964849354	16.07399988
TFIDF + 80 :20	0.974887892	7.133999825
TFIDF + 80 :20 + Stopwords	0.965022422	18.76799989
TFIDF + crossvalidation (3 folds)	0.970218873	32.31399989
TFIDF + crossvalidation (3 folds) + Stopwords	0.969501256	45.52999997
TFIDF + crossvalidation (5 folds)	0.97129399	57.19499993
TFIDF + crossvalidation (5 folds) + Stopwords	0.970215182	77.9849999
TFIDF + crossvalidation (10 folds)	0.971832856	118.7189999
TFIDF + crossvalidation (10 folds) + Stopwords	0.972727624	155.1690001

Few best results are graphically shown in Figure 5.3 for a better observation. Our best result with Random forest shows the 97.5% of ACC, 79.55% of SC and 0.1 % of BH. AuC for the same was covered 88.25% is shown in Figure 5.4.

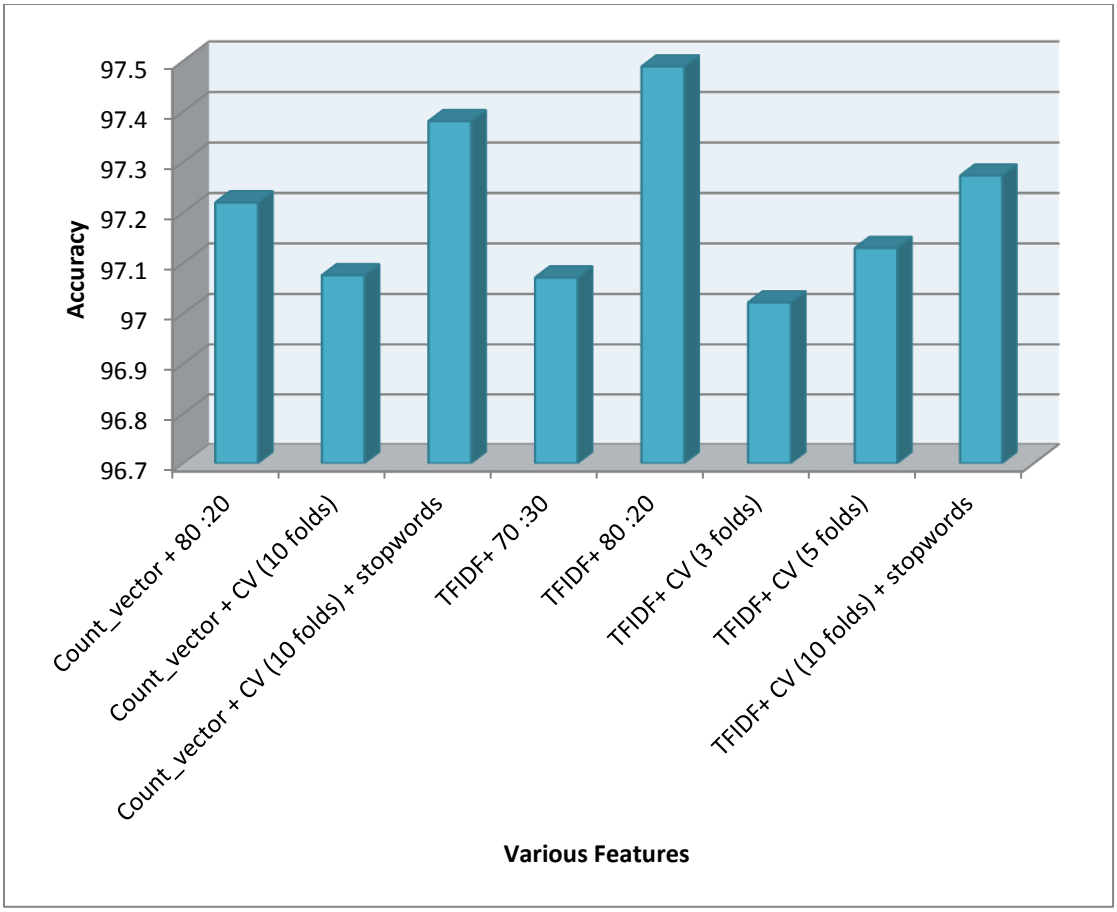


Figure 5.3: Dataset I: Comparison of various features with Random Forest

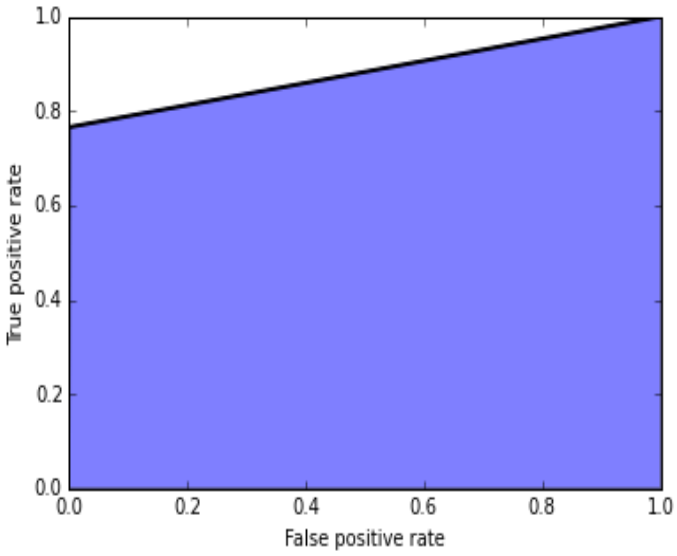


Figure 5.4: Area under curve for best result of Random Forest – 0.88

5.2.3 Support Vector Machines

In this section, Support Vector classifier was applied on the dataset after various feature extractions. SVM can be applied with different kernels such as linear, degree-2 polynomial, degree-3 polynomial, sigmoid etc. As the degree of kernel keeps increasing, the complexity of the approach and the time consumed keeps increasing. After performing various experiments, "Linear" kernel gave the best result for SVM. In terms of accuracy, SVM had the best performance. Hence for linear kernel SVM different results were obtained for various feature extractions and they are tabulated in table 5.3.

Table 5.3: Dataset I: Results of SVM

Features	Accuracy	Time – seconds
Count vectorizer + 70 :30	0.98266587	67.73200011
Count vectorizer + 70 :30 + Stopwords	0.980872684	68.02199984
Count vectorizer + 75 :25	0.984935438	85.11599994
Count vectorizer + 75 :25 + Stopwords	0.982065997	81.62400007
Count vectorizer + 80 :20	0.985650224	92.77699995
Count vectorizer + 80 :20 + Stopwords	0.984753363	88.86199999
Count vectorizer + crossvalidation (3 folds)	0.982956584	473.336
Count vectorizer + crossvalidation (3 folds) + sw	0.98116254	509.6029999
Count vectorizer + crossvalidation (5 folds)	0.98385295	849.2450001
Count vectorizer + crossvalidation (5 folds) + Stopwords	0.982059227	883.3529999
Count vectorizer + crossvalidation (10 folds)	0.984750909	1480.901
Count vectorizer + crossvalidation (10 folds) + Stopwords	0.982651709	1509.111
TFIDF + 70 :30	0.98684997	142.552
TFIDF + 70 :30 + Stopwords	0.982068141	131.323
TFIDF + 75 :25	0.988522238	120.9959998
TFIDF + 75 :25 + Stopwords	0.983500717	149.7849998
TFIDF + 80 :20	0.986547085	182.697999954
TFIDF + 80 :20 + Stopwords	0.984753363	168.408
TFIDF + crossvalidation (3 folds)	0.982597775	815.1530001
TFIDF + crossvalidation (3 folds) + Stopwords	0.978112666	671.155
TFIDF + crossvalidation (5 folds)	0.981520305	1387.453
TFIDF + crossvalidation (5 folds) + Stopwords	0.977751395	1230.168
TFIDF + crossvalidation (10 folds)	0.983674351	2261.908
TFIDF + crossvalidation (10 folds) + Stopwords	0.980621954	2640.273

Few best results are graphically compared in Figure 5.5 for a better observation. Our best result with SVM shows the 98.85% of ACC, 92.85% of SC and 0.25 % of BH. AuC for the same was covered 96% is shown in Figure 5.6.

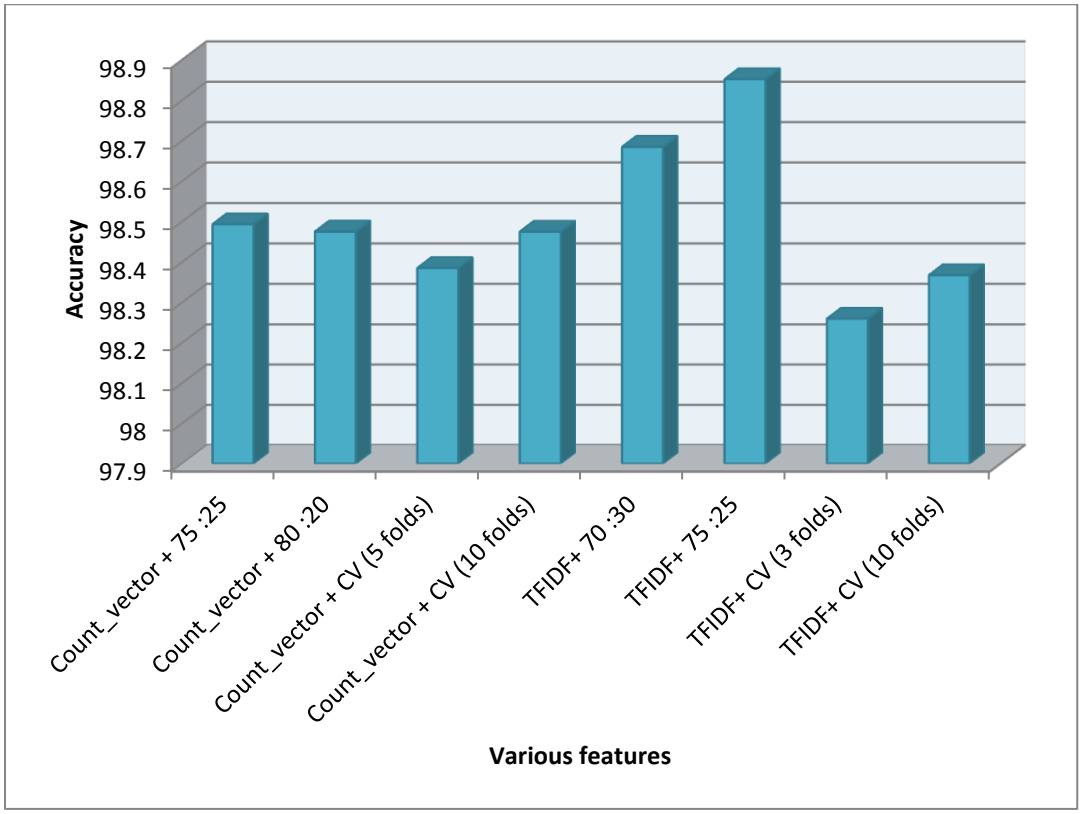


Figure 5.5: Dataset I: Comparison of various features with SVM

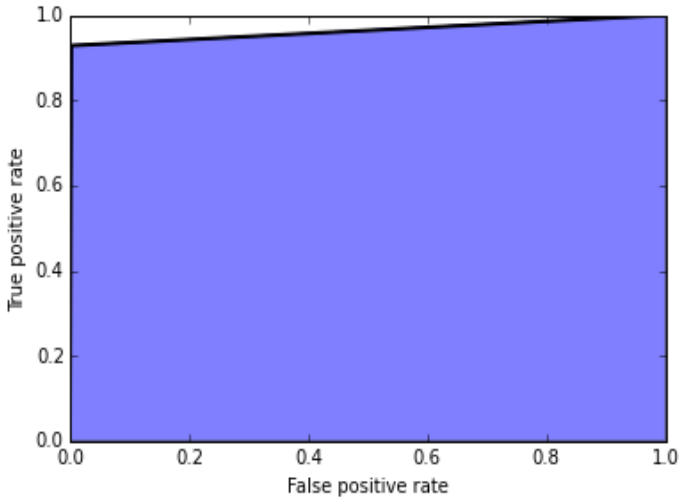


Figure 5.6: Area under curve for best result of SVM – 0.96

5.2.4 Adaboost

Adaboost is a boosting ensembling technique of classification which sequentially keeps building classifier for better results of previously misclassified instances. Adaboost works on the basis of applying different weights at every iteration, only initially weight is uniformly distributed. It was applied on the dataset after various feature extraction. Different results of Adaboost classifier were obtained for various feature extractions and they are tabulated in table 5.4.

Table 5.4: Dataset I: Results of Adaboost

Features	Accuracy	Time – seconds
Count vectorizer + 70 :30	0.976090855	287.1210001
Count vectorizer + 70 :30 + Stopwords	0.96951584	290.4230001
Count vectorizer + 75 :25	0.976327116	317.5519998
Count vectorizer + 75 :25 + Stopwords	0.9632287	340.0769999
Count vectorizer + 80 :20	0.981165919	361.3989999
Count vectorizer + 80 :20 + Stopwords	0.9632287	378.9190001
Count vectorizer + crossvalidation (3 folds)	0.979368497	1018.123
Count vectorizer + crossvalidation (3 folds) + Stopwords	0.973448152	974.1400001
Count vectorizer + crossvalidation (5 folds)	0.977933824	2248.508
Count vectorizer + crossvalidation (5 folds) + Stopwords	0.971293347	2054.662
Count vectorizer + crossvalidation (10 folds)	0.977394214	4897.22
Count vectorizer + crossvalidation (10 folds) + Stopwords	0.974341172	4861.292
TFIDF + 70 :30	0.972504483	396.5279999
TFIDF + 70 :30 + Stopwords	0.962050271	451.99
TFIDF + 75 :25	0.976327116	441.5090001
TFIDF + 75 :25 + Stopwords	0.952690716	590.776
TFIDF + 80 :20	0.973991031	513.7979999
TFIDF + 80 :20 + Stopwords	0.959349593	811.78
TFIDF + crossvalidation (3 folds)	0.977753857	1752.825
TFIDF + crossvalidation (3 folds) + Stopwords	0.969860065	1586.796
TFIDF + crossvalidation (5 folds)	0.982235542	2950.216
TFIDF + crossvalidation (5 folds) + Stopwords	0.973087714	2523.849
TFIDF + crossvalidation (10 folds)	0.980801166	5235.133
TFIDF + crossvalidation (10 folds) + Stopwords	0.974878807	4991.959

Few best results are shown graphically in figure 5.7 for the better observation. Our best result with Adaboost shows the 98.11% of ACC, 88.96% of SC and 0.51 % of BH. Area under curve for the same was covered 94% is shown in Figure 5.8.

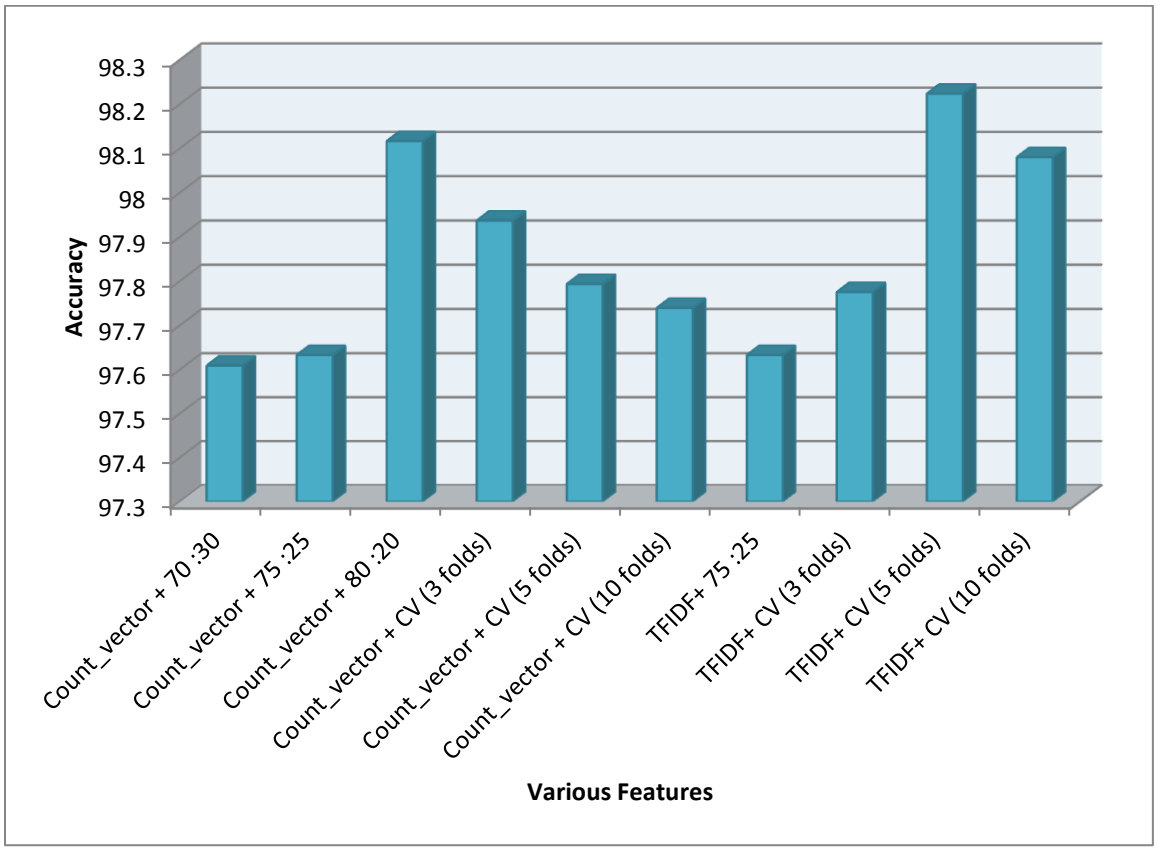


Figure 5.7: Dataset I: Comparison of various features with Adaboost

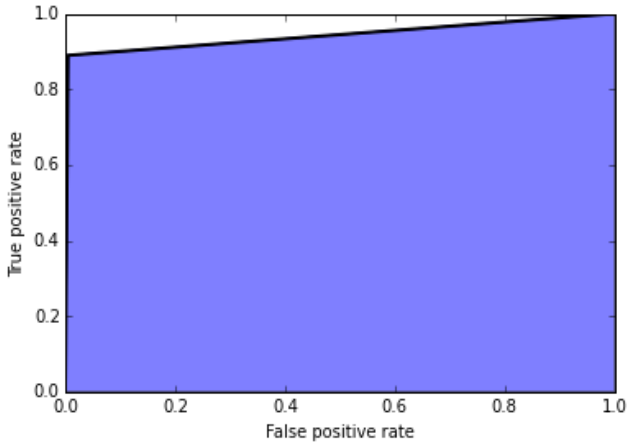


Figure 5.8: Area under curve for Adaboost– 0.94

5.3 Experiment II: Classification performed on Dataset II

Following classifiers were applied on the above mentioned dataset and results were observed as follows:

5.3.1 Multinomial Naive Bayes

It was applied on the dataset after various feature extraction. Different results of Multinomial Naive Bayes classifier were obtained for various feature extractions and they are tabulated in table 5.5.

Table 5.5: Dataset II: Results of MNB

Features	Accuracy	Time – seconds
Cross vectorizer + 70 :30	0.975357319	1.588000059
Cross vectorizer + 70 :30 + Stopwords	0.97683588	1.217000008
Cross vectorizer + 75 :25	0.975753992	1.522000074
Cross vectorizer + 75 :25 + Stopwords	0.975753992	1.342999935
Cross vectorizer + 80 :20	0.978566149	1.598999977
Cross vectorizer + 80 :20 + Stopwords	0.977827051	1.537999868
TFIDF + 70 :30	0.944307541	0.657999992
TFIDF + 70 :30 + Stopwords	0.956628881	0.667999983
TFIDF + 75 :25	0.947368421	0.710000038
TFIDF + 75 :25 + Stopwords	0.955647546	0.674000025
TFIDF + 80 :20	0.949741316	0.707999945
TFIDF + 80 :20 + Stopwords	0.959349593	0.700999975

Few best results are graphically compared in Figure 5.9 for a better observation. Our best result with MNB shows the 97.85% of ACC, 92.09% of SC and 0 .81 % of BH.

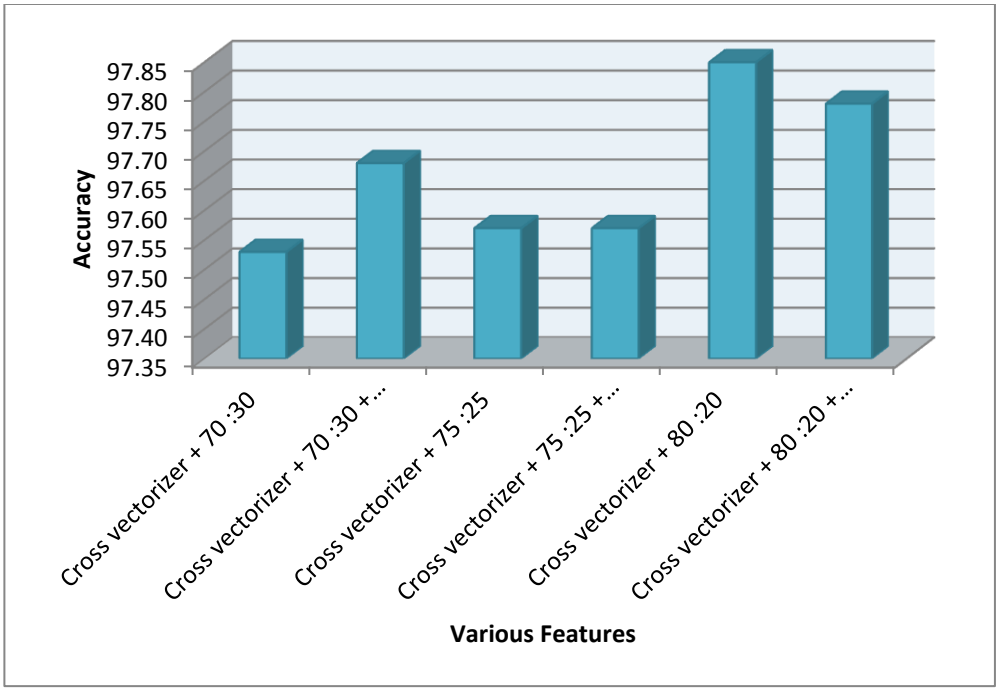


Figure 5.9: Dataset II: Comparison of various features with MNB

5.3.2 Random Forest

It was applied on the dataset after various feature extraction. Different results of Random Forest classifier were obtained for various feature extractions and they are tabulated in table 5.6.

Table 5.6: Dataset II: Results of Random Forest

Features	Accuracy	Time – seconds
Cross vectorizer + 70 :30	0.956136028	21.3440001
Cross vectorizer + 70 :30 + Stopwords	0.957614588	31.94800019
Cross vectorizer + 75 :25	0.968066233	40.57599998
Cross vectorizer + 75 :25 + Stopwords	0.966292135	40.1789999
Cross vectorizer + 80 :20	0.966740576	32.79299998
Cross vectorizer + 80 :20 + Stopwords	0.954175905	41.28299999
TFIDF + 70 :30	0.959093149	31.20499992
TFIDF + 70 :30 + Stopwords	0.958107442	37.77100015
TFIDF + 75 :25	0.956830278	17.58700013
TFIDF + 75 :25 + Stopwords	0.962152572	32.04299998
TFIDF + 80 :20	0.96082779	37.94000006
TFIDF + 80 :20 + Stopwords	0.96082779	43.92900014

Few best results are graphically compared in Figure 5.10 for a better observation. Our best result with Random Forest shows the 96.03 % of ACC, 78.75% of SC and 0 .14 % of BH.

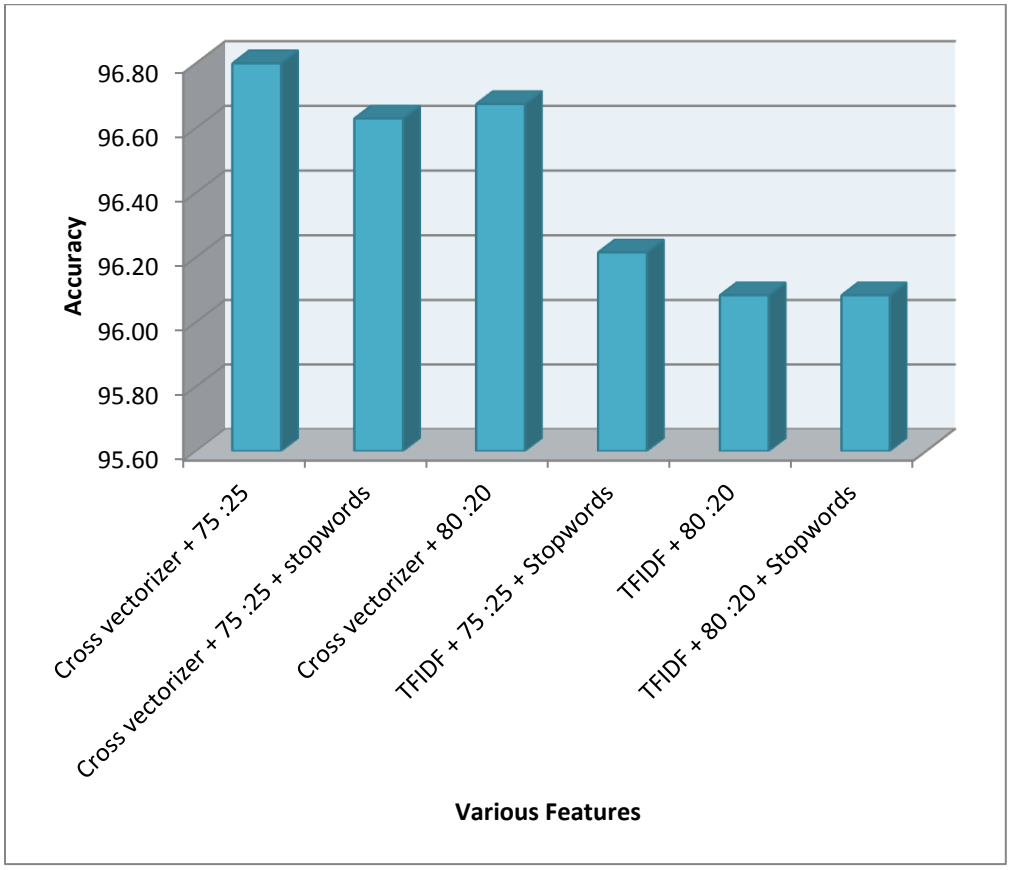


Figure 5.10: Dataset II: Comparison of various features with Random Forest

5.3.3 Support Vector Machine

It was applied on the dataset after various feature extraction. Different results of SVM classifier were obtained for various feature extractions and they are tabulated in table 5.7.

Table 5.7: Dataset II: Results of SVM

Features	Accuracy	Time – seconds
Cross vectorizer + 70 :30	0.974371612	234.497
Cross vectorizer + 70 :30 + Stopwords	0.973385904	228.625
Cross vectorizer + 75 :25	0.975753992	231.799
Cross vectorizer + 75 :25 + Stopwords	0.973388527	299.1459999
Cross vectorizer + 80 :20	0.977087953	312.6259999
Cross vectorizer + 80 :20 + Stopwords	0.975609756	313.783
TFIDF + 70 :30	0.980285855	369.6060002
TFIDF + 70 :30 + Stopwords	0.974371612	380.799
TFIDF + 75 :25	0.981667652	429.9630001
TFIDF + 75 :25 + Stopwords	0.973388527	431.552
TFIDF + 80 :20	0.982261641	467.6100001
TFIDF + 80 :20 + Stopwords	0.973392461	518.1470001

Few best results are graphically compared in Figure 5.11 for a better observation. Our best result with SVM shows the 98.22% of ACC, 92.88% of SC and 0.54 % of BH.

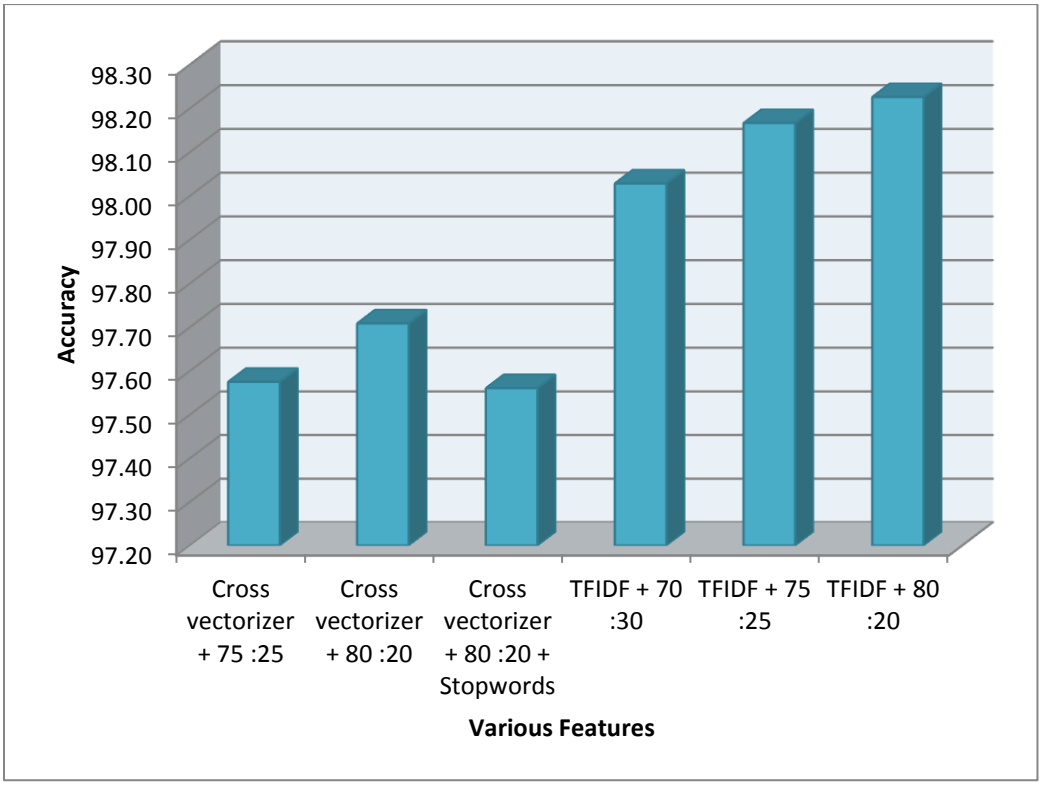


Figure 5.11: Dataset II: Comparison of various features with SVM

5.3.4 Adaboost

Different results of Adaboost classifier were obtained for various feature extractions and they are tabulated in table 5.8. Few best results are graphically compared in Figure 5.12 for a better observation. Our best result with Adaboost shows the 96.20% of ACC, 85.87% of SC and 1.61 % of BH.

Table 5.8: Dataset II: Results of Adaboost

Features	Accuracy	Time - seconds
Cross vectorizer + 70 :30	0.958107442	653.7969999
Cross vectorizer + 70 :30 + Stopwords	0.954657467	656.1430001
Cross vectorizer + 75 :25	0.957421644	582.954
Cross vectorizer + 75 :25 + Stopwords	0.954464814	568.2620001
Cross vectorizer + 80 :20	0.955654102	861.2850001
Cross vectorizer + 80 :20 + Stopwords	0.953436807	813.132
TFIDF + 70 :30	0.954164613	620.345
TFIDF + 70 :30 + Stopwords	0.962050271	522.2969999
TFIDF + 75 :25	0.958604376	601.3010001
TFIDF + 75 :25 + Stopwords	0.952690716	573.4679999
TFIDF + 80 :20	0.957871397	663.1239998
TFIDF + 80 :20 + Stopwords	0.959349593	636.3929999

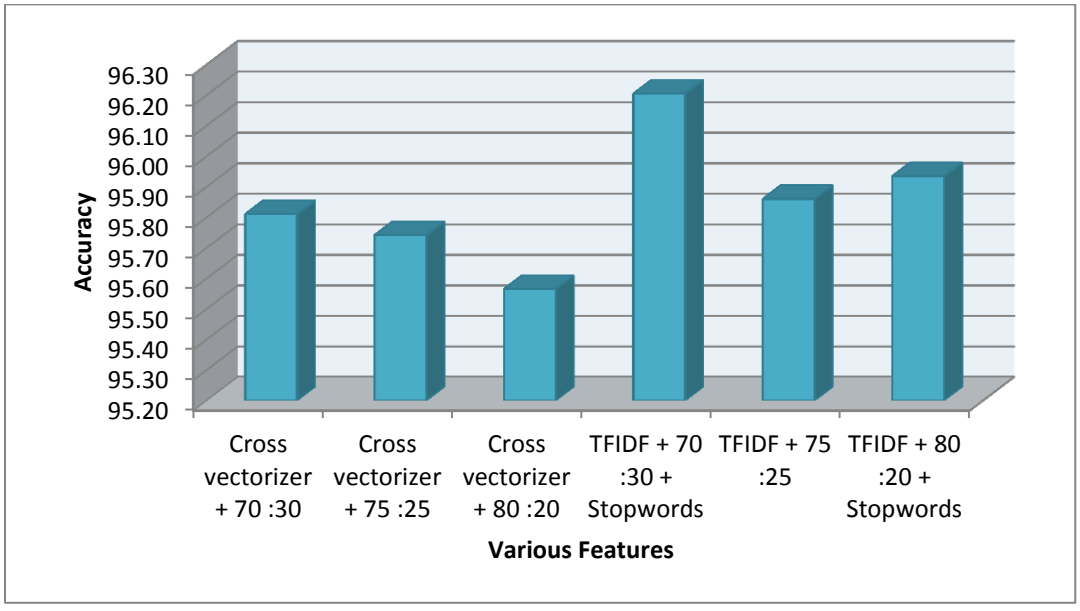


Figure 5.12: Dataset II: Comparison of various features with Adaboost

5.4 Results and Discussion

Various combinations of different pre-processing's and data transformations were applied with the classifiers on the datasets, and a numerous number of results were observed.The best results for each classifier are collected, for both the datasets. The various evaluation metrics like precision, recall, blocked hams, spams caught, f-measure etc. were calculated for the collected data and are tabulated in table 5.9 and table 5.10.

The immediate conclusion from the results table is, SVM has the best performance considering accuracy but it consumes a lot of time for the completion of classification process. For Dataset I, SVM got the 98.85% accuracy and for Dataset II, it got 98.23% accuracy.

Table 5.9: Best results of various classifiers on Dataset I

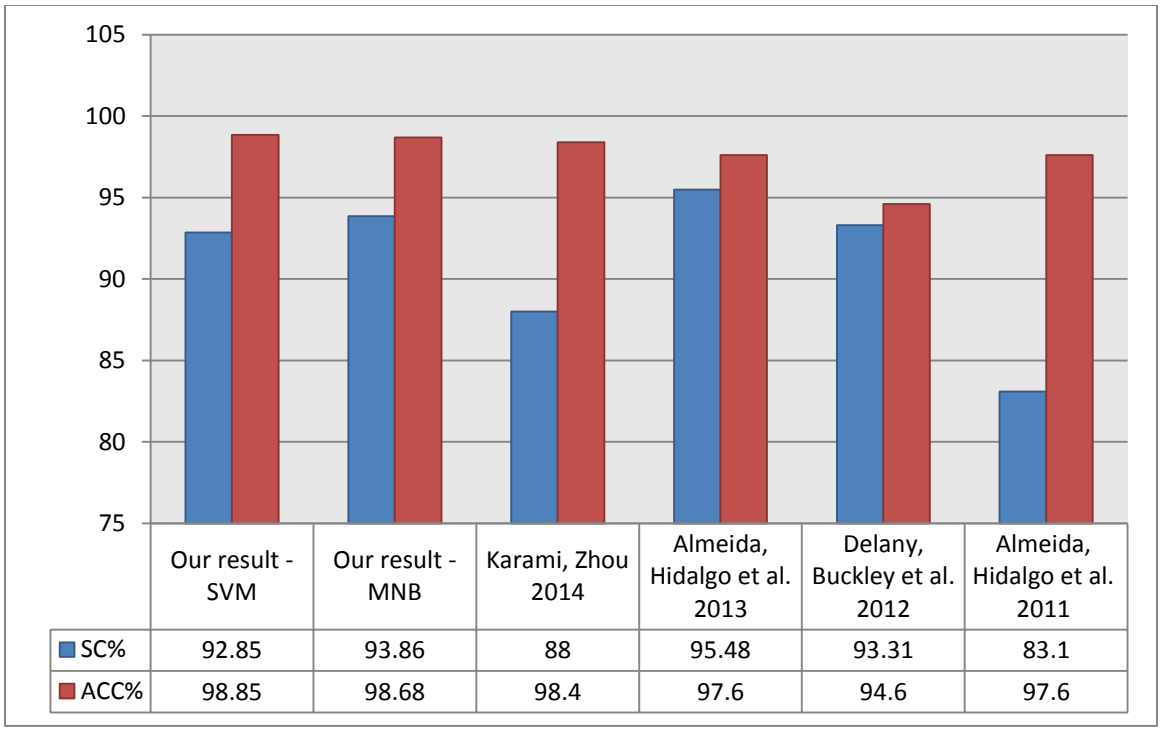
Classifiers	**MNB**	**RF**	**SVM**	**Adaboost**
ACC%	98.68	97.49	98.85	98.12
BH%	0.55	0.10	0.25	0.52
SC%	93.86	79.55	92.86	88.97
F-Measure	0.987	0.973	0.988	0.981
MCC	0.944	0.875	0.949	0.915
Precision	0.987	0.973	0.988	0.981
Recall	0.987	0.973	0.989	0.981
Time Taken (seconds)	0.534	7.134	133.269	361.082
AuC	0.967	0.880	0.960	0.940
TPR	0.987	0.973	0.989	0.981

Table 5.10: Best results of various classifiers on Dataset II

Classifiers	MNB	RF	SVM	Adaboost
ACC%	97.86	96.04	98.23	96.20
BH%	0.81	0.144	0.545	1.62
SC%	92.09	78.75	92.88	85.87
F-Measure	0.978	0.958	0.982	0.961
MCC	0.928	0.862	0.940	0.865
Precision	0.978	0.961	0.982	0.961
Recall	0.978	0.960	0.982	0.962
Time (Seconds)	2.03	26.84	443.97	515.21
AuC	0.956	0.893	0.961	0.921
TPR	0.978	0.960	0.982	0.962

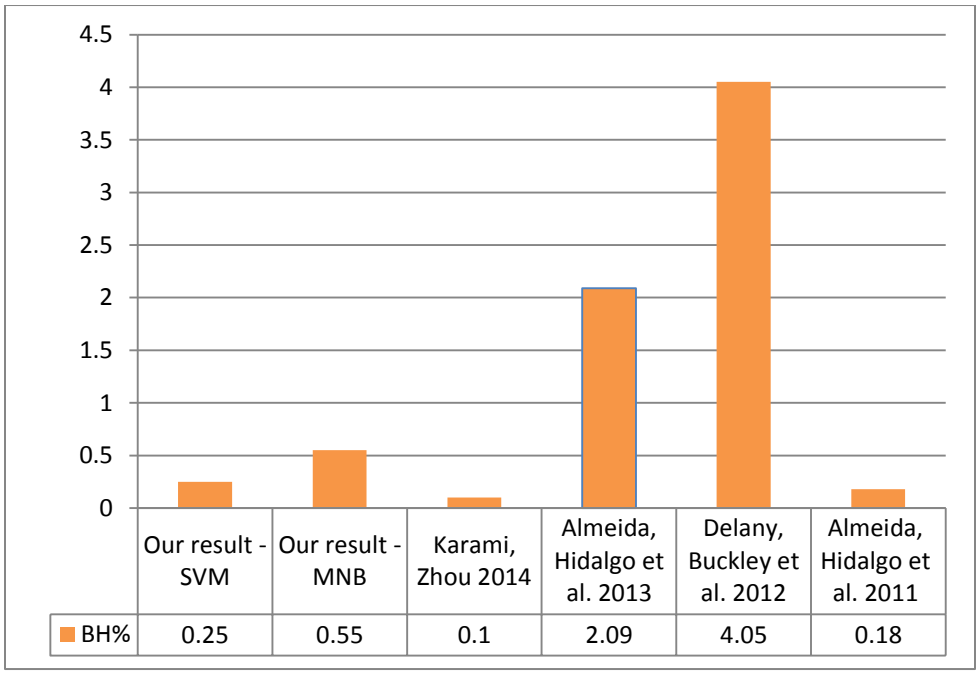
5.5 Comparison with Existing Works

Dataset I: SMS Spam Collection Data Set by T. A. Almeida *et al.* has been used globally for the research work of SMS spam classification. Our Experiment I is also based on the same dataset. We applied different classifiers with various features on this dataset and observed that on the application of SVM and MNB classifiers**,** the results have improved from the existing researches. Our work is compared graphically with the recent researches on the same dataset. Percentage of Accuracy (ACC) and Spam Caught (SC) are compared in the figure 6.1 and percentage of Blocked Hams (BH) is compared in figure 6.2. In our whole experiment, the maximum accuracy was achieved by SVM as 98.85% and second highest by MNB as 98.68%, which are more than the previous observed results.



	Our result - SVM	Our result - MNB	Karami, Zhou 2014	Almeida, Hidalgo et al. 2013	Delany, Buckley et al. 2012	Almeida, Hidalgo et al. 2011
SC%	92.85	93.86	88	95.48	93.31	83.1
ACC%	98.85	98.68	98.4	97.6	94.6	97.6

Figure 5.13: SC% and ACC% comparison with previous research results



	Our result - SVM	Our result - MNB	Karami, Zhou 2014	Almeida, Hidalgo et al. 2013	Delany, Buckley et al. 2012	Almeida, Hidalgo et al. 2011
BH%	0.25	0.55	0.1	2.09	4.05	0.18

Figure 5.14: BH% comparison with previous research results

Chapter 6

Conclusion and Future Work

The task of automatic SMS spam filtering is still a real challenge now days. The major problem faced in detection of spams in SMS is due the small number of characters in short text messages and the usual practice of idioms and abbreviations.

Various classification techniques were applied on both the datasets: SMS Spam Collection Data Set by T. A. Almeida et al. and Our Altered SMS Spam Collection Data Set including Indian content. The statistics of this study proves:

i) The combination of various features can lead to the improvement of the classification performance.

ii) Selection of features depends upon the type of classification technique being used.

iii) The data representation with TFIDF normalized weighting technique mostly contributed to the good results.

iv) SVM classifier with the Linear kernel had the best accuracy but the time required for the process was few minutes, which was quite high for SMS spam detection.

v) MNB with Laplace smoothing also had its accuracy very close to SVM but the time taken by MNB was far lesser than SVM.

vi) To make any anti-spammer in future for mobile messages, Multinomial Naive Bayes will surely prove to be the best classifier.

vii) According to our evaluations Random Forest yields the worst accuracy among the most common classifiers.

Future work must practice several approaches to escalate the aspect of the feature plot. Adding more meaningful features like certain thresholds for the length and analyzing the learning curves can contribute to the improvement in results. An application can be made for smartphones using this technique in future for protecting our cell phones from spam message.

References

[1] Network Security (29 Jul 2012). [Online]. Available: http://www.etymonline.com/index.php?term=secure. [Accessed: Oct 2014].

[2] I. Kuwatly, M. Sraj, Z. Al Masri and H. Artail. "A dynamic honeypot design for intrusion detection", *Pervasive Services IEEE/ACS International Conference*, pp. 95-104, 2004.

[3] G. Padmavathi and D. Shanmugapriya. "A Survey of Attacks, Security Mechanisms and Challenges in Wireless Networks", *International Journal of Computer Science and Information Security*, vol. 4, no. 1 & 2, 2009.

[4] M. Li, W. Lou and K. Ren. "Data security and privacy in wireless body area Networks", *IEEE Wireless Communications*, vol. 17, no. 1, pp. 51-58, 2010.

[5] SMS (Nov 2010). [Online]. Available: http://searchmobilecomputing.techtarget.com. [Accessed: Mar. 28, 2015].

[6] E. Bones, P. Hasvold, E. Henriksen and T. Strandenaes. "Risk analysis of information security in a mobile instant messaging and presence system for healthcare", *International Journal of Medical Informatics*, vol. 76, no. 9, pp. 677–687, September 2007.

[7] Portio Research (1996). [Online]. Available: http://www.portioresearch.com/en/home.aspx. [Accessed: Apr. 15, 2015].

[8] T. A. Almeida, J. Maria, G. Hidalgo and A. Yamakami. "Contributions to the study of SMS spam filtering: new collection and results", *Proceedings of the 11th ACM symposium on Document engineering*, 2011.

[9] Cloudmark Report (1998). [Online]. Available: https://www.cloudmark.com/en/s/products/cloudmark-gsma-spam-reporting-service. [Accessed: Apr. 17, 2015]

[10] W. Enck. "SMS Spam and Mobile Messaging Attacks-Introduction, Trends and Examples", GSMA Spam Reporting Service, January 2011.

[11] C. T. Wu, K. T. Cheng, Q. Zhu and Y. L. Wu. "Using Visual Features for anti-spam Filtering", *IEEE Image Processing International Conference*, vol. 3, pp. 509, 2005.

[12] Types of Spam (Mar 2013). [Online]. Available: http://www.theemailadmin.com/2010/09/6-different-types-of-spam-and-how-to-avoid-them/ [Accessed: Nov 2014].

[13] D. Ndumiyana, M. Magomelo and L. Sakala. "Spam detection using a Neural Network classifier", *Journal of Physical and Environmental Science Research*, vol. 2, no. 2, pp. 28-37, April 2013.

[14] T. M. Mahmoud and A. M. Mahfouz. "SMS Spam Filtering Technique Based on Artificial Immune System", *International Journal of Computer Science*, vol. 9, no. 1, 2012.

[15] P. Heymann, G. Koutrika and H. G. Molina. "Fighting spam on social web sites: A survey of approaches and future challenges", *IEEE Internet Computing*, vol. 11, no. 6, pp. 34-35, 2007.

[16] T. S. Guzella and W. M. Caminha. "A review of machine learning approaches to Spam filtering", *Expert Systems with Applications* , vol. 36, no. 7, pp. 10206–10222, September 2009.

[17] Y. Yang, S. J. Adelstein and A. I. Kassis. "Target discovery from data mining approaches", *Drug discovery today*, vol. 14, no. 3, pp. 147-154, 2009.

[18] Clustering (Aug 2009). [Online]. Available: http://www.tutorialspoint.com/data_mining/dm_cluster_analysis.html. [Accessed: Dec. 3, 2014].

[19] H. Zhang. "The optimality of naive Bayes", *AA* 1, vol. 3, no. 2, 2004.

[20] T. Joachims. "Text categorization with support vector machines: Learning with many relevant features", *Springer Berlin Heidelberg*, pp. 137-142, 1998.

[21] M. A. Friedle and C. E. Brodley. "Decision tree classification of land cover from remotely sensed data", *Remote sensing of environment*, pp. 399-409, 1997.

[22] M. R. Sikonja. "Improving random forests", *Machine Learning Springer Berlin Heidelberg*, pp. 359-370, 2004.

[23] G. Ratsch, T. Onoda and K. R. Muller. "Soft margins for AdaBoost", *Springer Berlin Heidelberg*, vol. 42, no. 3, pp. 287-320, 2001.

[24] Ron Bekkerman. "Distributional Clustering Of Words For Text Categorization", *Masters Thesis*, Israel Institute of Technology, Israel, 2003.

[25] J. M. Gomez Hidalgo, G. C. Bringas, E. P. Sanz and F. C. Garcia. “Content Based SMS Spam Filtering”, *Proceedings of the 2006 ACM Symposium on Document Engineering*, Amsterdam, The Netherlands, pp. 107–114, 2006.

[26] G. V. Cormack, J. M. Gomez Hidalgo and E. P. Sanz. “Spam Filtering for Short Messages”, *Proceedings of the 16th ACM Conference on Conference on information and Knowledge Management*, Lisbon, Portugal, pp. 313–320, 2007,

[27] M. T. Nuruzzaman, C. Lee, M. F. A. b. Abdullah and D. Choi. “Simple SMS spam filtering on independent mobile phone”, *Security and Communication Networks*, vol. 5, no.10, pp. 1209–1220, 2012.

[28] B. Coskun and P. Giura. “Mitigating SMS spam by online detection of repetitive near-duplicate messages”, *IEEE International Conference on Communications*, pp. 999–1004, 2012.

[29] S. J. Delany and M. Buckley. “Expert Systems with Applications”, *Expert Systems with Applications*, vol. 39, pp. 9899-9908, 2012.

[30] T. Almeida, J. M. Gomez Hidalgo and T. P. Silva. “Towards SMS Spam Filtering: Results under a New Dataset”, *International Journal of Information Security Science*, vol. 2, no. 1, 2013.

[31] H. S. Mehar. “SMS Spam Detection using Machine Learning Approach”, *International Journal of Information Security Science*, vol. 2, no. 2, 2013.

[32] M. Sahami, S. Dumais, D. Heckerman and E. Horvitz. “A bayesian approach to filtering junk e-mail”, *Learning for Text Categorization Papers from the AAAI Workshop*, pp. 55–62, 1998.

List of Publications

Status: Accepted

[1] S. Agarwal, S. Kaur and S. Garhwal. "Antispammer for Mobile Messages", *International Conference on Advanced Computer Science and Information Technology (ICACSIT*), Singapore, May 2015.

[2] S. Agarwal, S. Kaur and S. Garhwal. "SMS Spam Detection for Indian Market", *International Conference on Advances in Computer Science and Information Technology (ACSTY)*, Bangalore, June 2015.

Status: Communicated

[3] S. Agarwal, S. Kaur and S. Garhwal. "Improving SMS Spam Detection Using Supervised Machine Learning", *IEEE Transactions on Dependable and Secure Computing*.

Video Presentation

The video presentation for the topic “Design and Development of Antispammer for SMS Spam Detection” is available at the link mentioned below:

https://youtu.be/K5ObGe3ZgFw