EMAIL SECURITY AND TEXT SUMMARIZATION

Thesis submitted in partial fulfillment of the requirements for the Degree of

BACHELORS OF TECHNOLOGY

By

Tanvi Pruthi :121258

Under the guidance of

DR. RAJNI MOHANA

DEPARTMENT OF COMPUTER SCIENCE ENGINEERING

JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY, WAKNAGHAT

MAY, 2016

Department of Computer Science Engineering

Jaypee University of Information Technology Waknaghat, Solan-173234, Himachal Pradesh

DECLARATION BY THE SCHOLAR

I hereby declare that the work presented in this report entitled "**Email Security and Text Summarization"** in partial fulfilment of the requirements for the award of the degree of **Bachelor of Technology** in **Computer Science and Engineering** submitted in the department of Computer Science Engineering**, Jaypee University of Information Technology Waknaghat** is an authentic record of my own work carried out over a period from August 2015 to May 2016 under the supervision of **Dr. Rajni Mohana.** The matter embodied in the report has not been submitted for the award of any other degree or diploma.

Tanvi Pruthi (121258) .....

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

Department of Computer Science Engineering
Jaypee University of Information Technology,Waknaghat
MAY,2016

SUPERVISOR'S CERTIFICATE

This is to certify that the work reported in the B-Tech. project entitled "**EMAIL SECURITY AND TEXT SUMMARIZATION**", submitted by **Tanvi Pruthi-121258** at **Jaypee University of Information Technology, Waknaghat**, **India,** is a bonafide record of his / her original work carried out under my supervision. This work has not been submitted elsewhere for any other degree or diploma.

SUPERVISOR

_____

DR. RAJNI MOHANA

Department of Computer Science Engineering

Jaypee University of Information Technology, Waknaghat

MAY, 2016

ACKNOWLEDGEMENT

We would like to express our special thanks of gratitude to our project guide **Dr. Rajni Mohana** who helped us in conceptualizing the project and actual building of procedures used to complete the project. We would also like to thank our Head of department for providing us this golden opportunity to work on a project like this, which helped us in doing a lot of research and we came to know about so many things.

We wouls like to thank our Head of Department **Dr.S.P. Garera** for providing opportunity and the support required to work on this project.

We would like to thank our family and friends who guided us throughout the project so as to complete our project on time.

Thanking you,

Tanvi Pruthi (121258)

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

S No.	Abbreviations	Description
1	IDE	Integrated Development Environment
2	NLP	Natural Language Processing
3	HTML	Hypertext Markup Language
4	URL	Uniform Resource Locator
5	IP	Internet Protocol
6	JDT	Java Development Tools
7	PHP	Hypertext Preprocessor
8	PDT	PHP Development Tools
9	ADT	Anroid Development Tools
10	JSP	Java Server Pages
11	HTTP	Hypertext Transfer Protocol
13	JRE	Java Runtime Enterprise
14	SQL	Standardized Query Language
15	WAMP	Windows,Apache,MySQL,Perl
16	JDBC	Java Database Connectivity

ABSTRACT

There are hundreds of email summarization tools nowadays. One of the challenging issues of email summarization is to determine how to secure email summaries from spoofing and bombing and to provide preventive measures. Email is one of the most ubiquitous applications used on a daily basis by millions of people world-wide, traditionally accessed over a fixed terminal or laptop computer. In the past years, there has been an increasing demand for email access over mobile phones too. Our work focused on providing security review measures and preventing approaches that provide quality email summaries with secure transmissions over the network. A summary of document is a shorter text conveys the most important information from the sources .Summary of the text must contains important information from the documents. This paper presents the design and implementation of a system to summarize e mail messages. The system uses the subject and contents of the e mail message to classify e mails based on user's activities and generate summary of each incoming message.

1. INTRODUCTION

Not everyone in the organization needs to know how to secure the e-mail service, but anyone who handles patient information must understand e-mail's vulnerabilities and recognize when a system is secure enough to transmit sensitive information.

E-mail messages are generally sent over untrusted networks-external networks that are outside the organization's security boundary. When these messages lack appropriate security safeguards, they are like postcards that can be read, copied, and modified at any point along these paths.

Securing an e-mail system is the responsibility of an organization's IT department and e-mail administrator. However, anyone responsible for the confidentiality, integrity, and availability of the information sent via e-mail should be aware of the threats facing e-mail systems and understand the basic techniques for securing these systems.

1.1 The Email System in the nutshell

An e-mail system is made up of two primary components that reside in an organization's IT infrastructure: mail clients and mail servers.

Users read, compose, send, and store their e-mail using mail clients. Mail is formatted and sent from the mail client via the network infrastructure to a mail server. The mail server is the computer that delivers, forwards, and stores e-mail messages. All components—the mail servers, the mail clients, and the infrastructure that connects and supports them—must be protected.

E-mail security relies on principles of good planning and management that provide for the security of both the e-mail system and the IT infrastructure. With proper planning, system management, and continuous monitoring, organizations can implement and maintain effective security

1.2 Common Threats

Because e-mail is widely deployed, well understood, and used to communicate with untrusted, external organizations, it is frequently the target of attacks. Attackers can exploit e-mail to gain control over an organization, access confidential information, or disrupt IT access to resources. Common threats to e-mail systems include the following:

Malware. Increasingly, attackers are taking advantage of e-mail to deliver a variety of attacks to organizations through the use of malware, or "malicious software," that include viruses, worms, Trojan horses, and spyware. These attacks, if successful, may give the malicious entity control over workstations and servers, which can then be exploited to change privileges, gain access to sensitive information, monitor users' activities, and perform other malicious actions.

Spam and phishing. Unsolicited commercial e-mail, commonly referred to as spam, is the sending of unwanted bulk commercial e-mail messages. Such messages can disrupt user productivity, utilize IT resources excessively, and be used as a distribution mechanism for malware. Related to spam is phishing, which refers to the use of deceptive computer-based means to trick individuals into responding to the e-mail and disclosing sensitive information. Compromised e-mail systems are often used to deliver spam messages and conduct phishing attacks using an otherwise trusted e-mail address.

Social engineering. Rather than hack into a system, an attacker can use e-mail to gather sensitive information from an organization's users or get users to perform actions that further an attack. A common social engineering attack is e-mail spoofing, in which one person or program successfully masquerades as another by falsifying the sender information shown in e-mails to hide the true origin.

Entities with malicious intent. Malicious entities may gain unauthorized access to resources elsewhere in the organization's network via a successful attack on a mail server. For example, once the mail server is compromised, an attacker could retrieve users'

passwords, which may grant the attacker access to other hosts on the organization's network.

Unintentional acts by authorized users. Not all security threats are intentional. Authorized users may inadvertently send proprietary or other sensitive information via e-mail, exposing the organization to embarrassment or legal action.

1.3 Five facts every small business should know about Email based threats

Even with today's breakthroughs in online communication, email is still one of the main ways that most people connect and keep in touch. This is especially true in the business setting. Email is used to such an extent that the total worldwide email traffic including both business and consumer emails is estimated to be over 144 billion emails per day at the tail end of 2012.1 The amount of email traffic is also predicted to grow to over 192 billion emails every day by 2016. Email-based threats are still a problem for everyone, including small and medium-sized businesses (SMBs). With such a high degree of usage, it's easy to see why cybercriminals continue to use email to facilitate their attacks.

Email spam can clog up servers and inboxes leading to reduced work productivity.

Unsolicited bulk email aka spam are generally considered a nuisance because the majority consists of ads that sell particular services or products. The digital equivalent of junk mail, spam in this form may first appear as harmless, even innocent. .

Email is a common entry point of malware infection.

This malware logs keystrokes when users browse online banking websites and effectively steals their login credentials. Stolen information is then sent to the cybercriminals, enabling them to infiltrate and steal from users' bank accounts. A number of SMBs in the United States filed for bankruptcy due to the hundreds of thousands of dollars stolen by cybercriminals who utilized this malware.

Phishing attacks can reach SMBs through email.

Phishing is another email-based threat that SMBs have to deal with. Phishing is the act of tricking someone into voluntarily giving out personal information. Attackers typically send out spammed messages that point to or provide a link to malicious sites. Through social engineering, these phishing attacks trick users into voluntarily giving out sensitive information or download malware that steal.

Cybercriminals invest money in exploiting the usage of email.

Small businesses should be aware that cybercriminals put considerable resources into finding new ways to use email for their malicious deeds. An example of this is the Blackhole Exploit Kit. An exploit kit is a web application that allows cybercriminals to take advantage of known vulnerabilities in popular applications like Internet Explorer, Adobe Acrobat, Adobe Reader and Flash Player through malicious spam runs.

Email-based threats are going to be a continuous problem.

Email has been part and parcel of any business that uses the Internet – and what company nowadays doesn't use email as part of their work process? SMBs should then acknowledge that the risk of email-based threats is very real, and should take measures to guard against such threats.

1.4 The Proposed Method

We propose a statistical method to extract the best sentences as a summary candidate based on features scores for each sentence. Therefore, the features score of each sentence that will be described in this section are used to obtain the significant sentences. This method consists of the following main steps:

a) Separation of sentences.

b) Perform Tokenization, the Removal of stop words and Stemming process.

c) The features are calculated to obtain the sentence score base on our proposed method.

d) A set of highest score sentences are extracted as document summary.

1.5 Natural Language Processing (NLP)

NLP has become a hot topic in recent days and we find working with such technologies very exciting. We decided to come with a nice Summarization tool that can give summary of a given URL along with top image, top video from the content, thus giving birth to our service, named 'PithPicker' Engine. Generally summary can be rewritten with original text or extracting key sentences from the text. Second approach will work better in most cases, hence becomes our preferred method.

- **Extracting the pivotal text from the given URLs**

  We have the script that takes web URLs to extract full content and then HTML content is passed through a template removal process so that we extract the main article after excluding headers, footers, advertisements and sidebars. Then it is fed to a DOM parser that can extract article data. Article data includes meta description, title, actual content, top image and top video.

- **Summarizing the given paragraph**

  The content and title from extraction step are used here as input. We use the following main criteria to identify top 5 sentences,

  - Title words' presence
  - Length of the sentences
  - Sentence position
  - Keywords' presence on sentences and their intersection
  - Frequency of occurrence[11]

1.6 The Art of Tokenization

The process of segmenting running text into words and sentences.

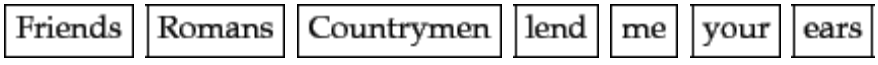
Electronic text is a linear sequence of symbols (characters or words or phrases). Naturally, before any real text processing is to be done, text needs to be segmented into linguistic units such as words, punctuation, numbers, alpha-numerics, etc. This process is called tokenization.

In English, words are often separated from each other by blanks (white space), but not all white space is equal. Both “Los Angeles” and “rock 'n' roll” are individual thoughts despite the fact that they contain multiple words and spaces. We may also need to separate single words like “I'm” into separate words “I” and “am”.

Tokenization is a kind of pre-processing in a sense; an identification of basic units to be processed. It is conventional to concentrate on pure analysis or generation while taking basic units for granted. Yet without these basic units clearly segregated it is impossible to carry out any analysis or generation.

Given a character sequence and a defined document unit, tokenization is the task of chopping it up into pieces, called tokens , perhaps at the same time throwing away certain characters, such as punctuation. Here is an example of tokenization:

Input: Friends, Romans, Countrymen, lend me your ears;

Output: 

1.7 Email Summarization

The goal of text summarization is to present the most important information in a shorter version of the original text while keeping its main content and helps the user to quickly understand large volumes of information. Email text summarization addresses both the problem of selecting the most important sections of text and the problem of generating coherent summaries. This process is significantly different from that of human based text summarization since human can capture and relate deep meanings and themes of text documents while automation of such a skill is very difficult to implement. With the ever increasing popularity of emails, email over-load becomes a major problem for many email users [1]. Users spend a lot of time reading, replying and organizing their emails. To help users organize their email folders, many forms of support have been proposed, including spam filtering [2], email classification [3] and email visualization [4]. In this research, we discuss a different form of support email summarization. The goal is to provide a concise, informative summary of email conversation. Email summarization can also be valuable for users reading emails with mobile devices. Given the small screen size of handheld devices, efforts have been made to re-design the user interface. However, providing a concise summary may be just as important.

This file is the main entry point of the program. The program starts and presents a simple prompt to select a choice. These choices are: Enter a paragraph manually or Enter Paragraph from a file.

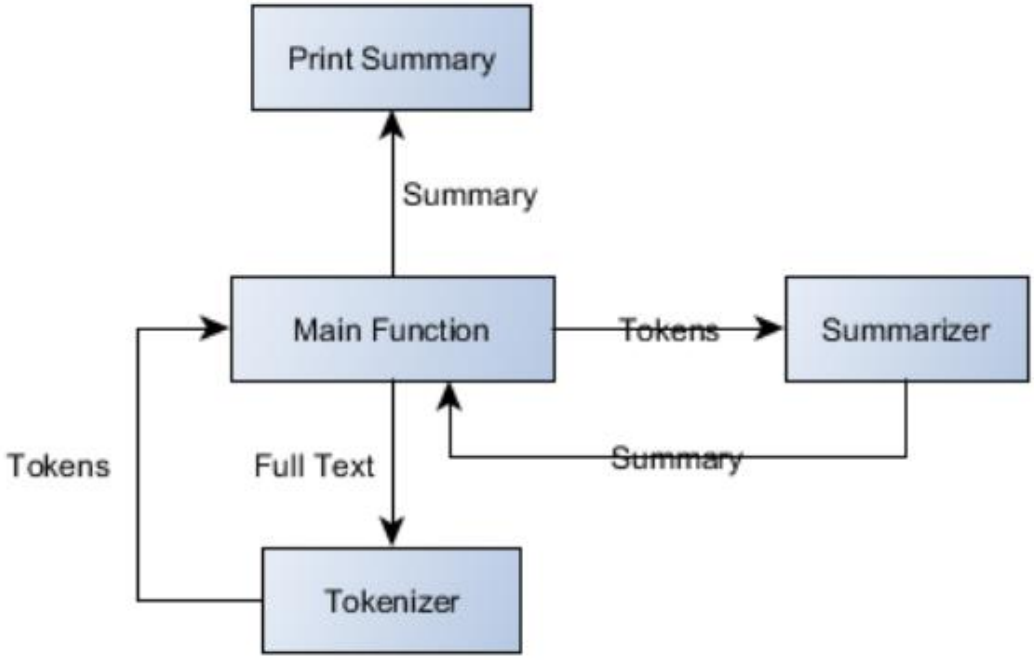


Figure 1: Email Summarizer[1]

This summarizer summarizes the emails and also extract dates from an email in order to make sure some important info like event, meeting, anniversary or birthday etc. is captured as important attribute. Let us have a look at the overall working on the system.

The system consists of four Java files, each file with its own working. These files are as follow:

1. Main file (EmailSummarizer.java)

2. Tokenizer.java

3. Summarizer.java

4. PorterStemmer.java

1) Summarizing Email Conversations with Clue Words

The goal is to provide a concise, informative summary of emails contained in a folder, thus saving the user from browsing through each email one by one. The summary is intended to be multi-granularity in that the user can specify the size of the concise summary (e.g., depending on how much time the user wants to spend on the folder). Email summarization can also be valuable for users reading emails with mobile devices. Given the small screen size of handheld devices, efforts have been made to re-design the user interface. However, providing a concise summary may be just as important.[10]

2) Summarizing Emails with Conversational Cohesion and Subjectivity

We propose new summarization approaches by sentence extraction, which rely on a fine-grain representation of the conversation structure. We first build a sentence quotation graph by content analysis. This graph not only captures the conversation structure more accurately, especially for selective quotations, but it also represents the conversation structure at the finer granularity of sentences. As a second contribution of this paper, we study several ways to measure the cohesion between parent and child sentences in the quotation graph: clue words, semantic similarity and cosine similarity. Hence, we can directly evaluate the importance of each sentence in terms of its cohesion with related ones in the graph. The extractive summarization problem can be viewed as a node ranking problem. We apply two summarization algorithms, Generalized ClueWordSummarizer and Page-Rank to rank nodes in the sentence quotation graph and to select the corresponding most highly ranked sentences as the summary.[9]

3) Regression-Based Summarization of Email Conversations

Usually, the information in a given document is not constant, which means that some parts of document are more important than others are less important. The main challenge

is to identify important parts of document and extract them for final summary. Here most work presented on single-document summarization using extraction method. In this section, some extractive techniques are discussed briefly, which are applied for extraction of sentences for final summary. Extraction technique is divided into two steps1. Pre Processing 2. Processing. Preprocessing phase involves three steps

a) Sentences boundary identification.

b) Stop-Word Elimination

c) Stemming.

In processing phase, feature value for every sentence is calculated. Score of every sentence between 0 to 1 and then weights are assigned to these features using weight learning method.[8]

4)Email Classification and Summarization: A Machine Learning Approach

Our contribution in this area is the consideration of the highest frequencies of words in email messages, with selection of the sentences that contain the most frequent words and re-arranging these in an order that generates a good summary. The algorithm is shown in the Figure 1 below: Summarization Algorithm Summary as Input: N, M, Message Output: Sentence list

1). Identify N most frequent words in the message

2). Select M sentences from email containing most frequent words

3). Order the selected sentences according to their occurrence in the message

4). Output the ordered sentences [1]

1.8 Effective spam detection

Spam is abuse of electronic messaging system to send unsolicited bulk messages. Emails are used by number of user to communicate around the world. Along with growth of internet and email, there has been dramatic growth in spam in recent year. Spam can originate from any location across globe, where internet access is available. Spam was created by Hornel in 1937 as the world's first canned meat that didn't need to be refrigated. It was originally named "Hornel Spiced Ham", but was eventually changed to the catchier name, "SPAM" . Usually they come in the form of advertisement, sometimes even containing explicit content or malicious code. Spam has been recognized as problem since1975. According to the statistics from ITU (International Telecommunication Union), 70% to 80% of emails in the internet are spams which have become worldly problem to the information infrastructure. In order to address growing problem there so many anti-spam methods.

Spam is abuse of electronic messaging system to send unsolicited bulk messages. Today large volumes of spam emails are causing serious problem for the users, and internet services. Such as, It degrades user search experience, It assists propagation of virus in network, It increase load on the network traffic, It wastes the resources such as bandwidth, storage, and computation power, It also wastes the user time and energy. General advices to avoid spam's are use the spam filter, Never reply the spam, Don't post your email address on your web site, and Never buy anything from spam.[6]

1.9 Spam Detection Methods

1.9.1 List Based or Rule Based Filters

- **Black List**

  Black list is the form of rule based filtering that uses one rule to decide which emails are spams. Black list are the list of IP address of machine or record of email addresses that have been previously used to send spam. When incoming message arrives, the spam filter checks to see if it's IP or email address is on the black list, if so, the message is considered spam and rejected. Blacklist can be used for on both large scale and small scales.

- **White List**

  While Blacklisting is used to decide which emails are spam, but White listing is used to decide which emails are ham and assume all other emails are spam.

- **Blackholes**

  Spam Blackholes work hand in hand with Blacklist. The way Blackholes work is someone posts message on websites, Usenet, forum, etc, showing their email address. The email address they use is generally a machine account that detects who sent the spam and the IP address of to a DNS Blacklist. Advantage is the email is received from one of these addresses the sending server can added to a Blacklist stopping it from sending any more messages.

- **Grey List**

  A relatively new spam filtering technique, it takes the advantage of the fact that many spammers only attempt to send a batch of junk mail once. Under the greylist system, the receiving mail server initially rejects messages from unknown users and sends a failure message to the originating server. If the mail server attempts to send the message second time- a step most legitimate server will take – the greylist assumes the message is not spam and let it proceed to the recipient's

inbox. At this time greylist filter will add the recipient's email or address to a list of allowed senders. Though greylist filter require fewer system resources than some other types of spam filters, they also delay mail delivery, which could be inconvenient when you're expecting time sensitive messages.

1.9.2 Content Based Filter

Content Based Filter is the most commonly used group of methods to filter spam. Content filter act either on the content, the information contained in the mail body, or on the mail headers (like "Subjects") to either classify, accept or reject a message

Particular words have particular probabilities of occurring in spam email and in legitimate email. The filter does not know these probabilities in advance, and must first be trained so it can build them up. To train the filter user must manually indicate whether a new email is spam or not. for all words in each training email, the filter will adjust the probabilities that each word will appear in spam or legitimate email in its database. For instance, Bayesian spam filter will typically have learned a very high spam probability for the words "award" and "viruses", but a very low spam probability for words seen only in legitimate email, such as names of friends and family members. After training the words probabilities are used to compute the probability that an email with a particular set of words in it belonging to either category. Each word in the email contributes to the email's spam probability. This contribution is called posterior probability and is computed using Bayes theorem. Then email's spam probability is computed over all words in the email, and if the total exceeds a certain threshold (for ex.: 95%), the filter will mark the email as spam. Email marked as spam then be automatically moved to a "Junk" email folder, or even deleted outright[7]

1.10 Spam Detection Framework

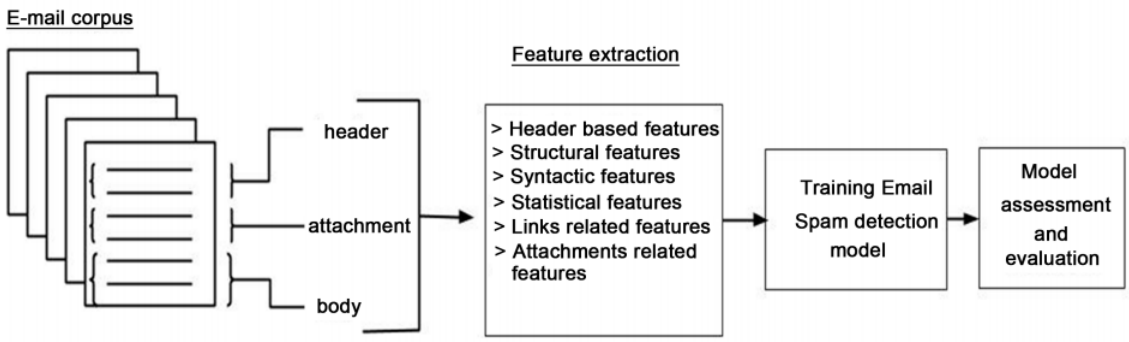


Figure 2: Spam Detection Framework[6]

2 LITERATURE REVIEW

- **Text summarization : An overview Elena Lloret Dept. Lenguajes y Sistemas Inform´aticos Universidad de Alicante Alicante, Spain elloret@dlsi.ua.es**

This paper presents an overview of Text Summarization. Text Summarization is a challenging problem these days. Due to the great amount of information we are provided with and thanks to the development of Internet technologies, needs of producing summaries have become more and more widespread. Summarization is a very interesting and useful task that gives support to many other tasks as well as it takes advantage of the techniques developed for related Natural Language Processing tasks. The paper we present here may help us to have an idea of what Text Summarization is and how it can be useful for. This paper addresses the current state-of the-art of Text Summarization, gives an overview of the field TS and we present the factors related to it, explains the different approaches to generate summaries, we present a number of Text Summarization systems existing today and presents the common measures to evaluate those systems, exposes the tendency adopted these days in Text Summarization.

- **A Survey of Unstructured Text Summarization Techniques Sherif Elfayoumy School of Computing University of North Florida Jacksonville, Florida Jenny Thoppil School of Computing University of North Florida Jacksonville, Florida**

Due to the explosive amounts of text data being created and organizations increased desire to leverage their data corpora, especially with the availability of Big Data platforms, there is not usually enough time to read and understand each document and make decisions based on document contents. Hence, there is a great demand for summarizing text documents to provide a representative substitute for the original documents. By improving summarizing techniques, precision of document retrieval through search queries against summarized documents is expected to improve in comparison to querying against the full spectrum of original documents.

Several generic text summarization algorithms have been developed, each with its own advantages and disadvantages. For example, some algorithms are particularly good for summarizing short documents but not for long ones. Others perform well in identifying and summarizing single-topic documents but their precision degrades sharply with multi-topic documents. In this article we present a survey of the literature in text summarization. We also surveyed some of the most common evaluation methods for the quality of automated text summarization techniques

- **Regression Based Summarization of Email Conversation JanUlrich and GiuseppeCarenini and GabrielMurray and RaymondNg {ulrichj, carenini, gabrielm, rng}@cs.ubc.ca Department of Computer Science University of British Columbia, Canada**

In this paper we present a regression-based machine learning approach to email thread summarization. The regression model is able to take advantage of multiple gold-standard annotations for training purposes, in contrast to most work with binary classifiers. We also investigate the usefulness of novel features such as speech acts. This paper also

introduces a newly created and publicly available email corpus for summarization research. We show that regression-based classifiers perform better than binary classifiers because they preserve more information about annotator judgements. In our comparison between different regression-based classifiers, we found that Bagging and Gaussian Processes have the highest weighted recall.

Summarization is a promising way to reducing this email triage. Email summarization has many more uses than just summarizing incoming emails. In the business world, email summarization can be used as a form of corporate memory, where the thread summaries represent all the previous business decisions that have been made. As another possibility, it also allows a new team member to more easily and quickly catch up on an ongoing conversation in a discussion forum. Home automation system faces four main challenges; these are high cost of ownership, inflexibility, poor manageability, and difficulty in achieving security.

- **Summarizing Emails with Conversational Cohesion and Subjectivity**
  Giuseppe Carenini, Raymond T. Ng and Xiaodong Zhou Department of Computer Science University of British Columbia Vancouver, BC, Canada {carenini, rng, xdzhou}@cs.ubc.ca

In this paper, we study the problem of summarizing email conversations. We first build a sentence quotation graph that captures the conversation structure among emails. We adopt three cohesion measures: clue words, semantic similarity and cosine similarity as the weight of the edges. Second, we use two graph-based summarization approaches, Generalized ClueWordSummarizer and PageRank, to extract sentences as summaries. Third, we propose a summarization approach based on subjective opinions and integrate it with the graph-based ones. The empirical evaluation shows that the basic clue words have the highest accuracy among the three cohesion measures. Moreover, subjective words can significantly improve accuracy.

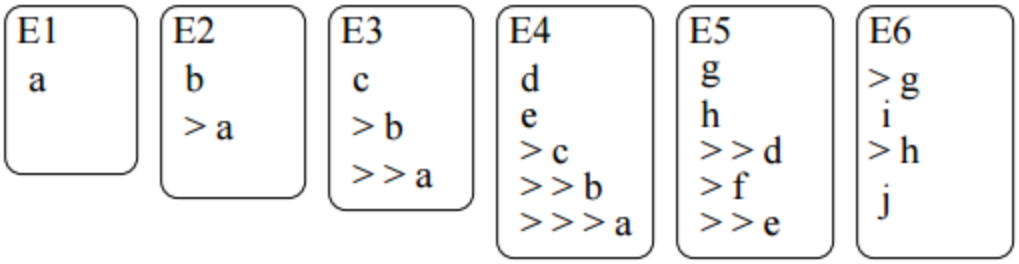


Figure 3: Conversation involving 6 Emails[9]

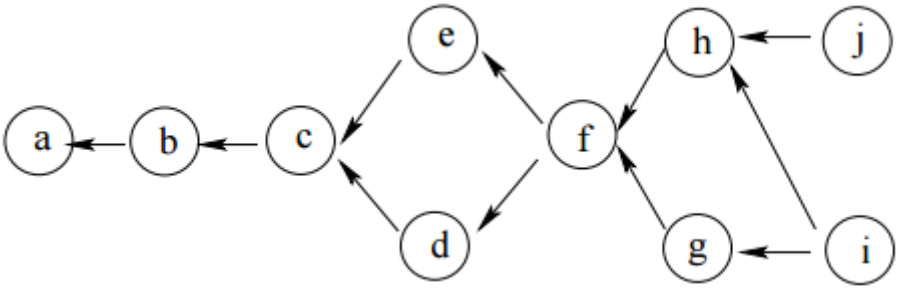


Figure 4: Fragment Quotation Graph[9]

- **Summarizing Email Conversations with Clue Words Giuseppe Carenini, Raymond T. Ng, Xiaodong Zhou Department of Computer Science University of British Columbia, Canada {carenini, rng, xdzhou}@cs.ubc.ca**

Accessing an ever increasing number of emails, possibly on small mobile devices, has become a major problem for many users. Email summarization is a promising way to solve this problem. In this paper, we propose a new framework for email summarization. One novelty is to use a fragment quotation graph to try to capture an email conversation. The second novelty is to use clue words to measure the importance of sentences in conversation summarization. Based on clue words and their scores, we propose a method called CWS, which is capable of producing a summary of any length as requested by the user. We provide a comprehensive comparison of CWS with various existing methods on

the Enron data set. Preliminary results suggest that CWS provides better summaries than existing methods.

- **A Survey of Text Summarization Extractive Techniques Vishal Gupta University Institute of Engineering & Technology, Computer Science & Engineering, Panjab University Chandigarh, India, Email: vishal@pu.ac.in Gurpreet Singh Lehal Department of Computer Science, Punjabi University Patiala, Punjab, India, Email: gslehal@yahoo.com**

Text Summarization is condensing the source text into a shorter version preserving its information content and overall meaning. It is very difficult for human beings to manually summarize large documents of text. Text Summarization methods can be classified into extractive and abstractive summarization. An extractive summarization method consists of selecting important sentences, paragraphs etc. from the original document and concatenating them into shorter form. The importance of sentences is decided based on statistical and linguistic features of sentences. An abstractive summarization method consists of understanding the original text and re-telling it in fewer words. It uses linguistic methods to examine and interpret the text and then to find the new concepts and expressions to best describe it by generating a new shorter text that conveys the most important information from the original text document. In this paper, a Survey of Text Summarization Extractive techniques has been presented

- **Effective Spam Detection Method for Email Savita Teli1 , Santoshkumar Biradar2 1 (Student, Dept of Computer Engg, Dr. D. Y. Patil College of Engg, Ambi, University of Pune, M.S, India) 2 (Asst. Proff, Dept of Computer Engg, Dr. D. Y. Patil College of Engg, Ambi, University of Pune, M.S, India)**

Spam emails are the emails receiver does not wish to receive; it is also called unsolicited bulk email. Emails are used daily by number of user to communicate around the world.

Today large volumes of spam emails are causing serious problem for Internet user and Internet service. Such as it degrades user search experience, it assists propagation of virus in network, it increases load on network traffic. It also wastes user time, and energy for legitimate emails among the spam. For avoiding spam there are so many traditional anti spam techniques includes Bayesian based filters, rule based system, IP blacklist, Heuristic based filter, White list and DNS black holes. These methods are based on content of the mail or links of the mail. In this paper, we presented our study on various existing spam detection methods and finding the effective, accurate, and reliable spam detection method.

- **Improving Knowledge Based Spam Detection Methods: The Effect of Malicious Related Features in Imbalance Data Distribution Ja'far Alqatawna, Hossam Faris, Khalid Jaradat, Malek Al-Zewairi, Omar Adwan King Abdullah II School for Information Technology, The University of Jordan, Amman, Jordan.**

Spam is no longer just commercial unsolicited email messages that waste our time, it consumes network traffic and mail servers' storage. Furthermore, spam has become a major component of several attack vectors including attacks such as phishing, cross-site scripting, cross-site request forgery and malware infection. Statistics show that the amount of spam containing malicious contents increased compared to the one advertising legitimate products and services. In this paper, the issue of spam detection is investigated with the aim to develop an efficient method to identify spam email based on the analysis of the content of email messages. We identify a set of features that have a considerable number of malicious related features. Our goal is to study the effect of these features in helping the classical classifiers in identifying spam emails. To make the problem more challenging, we developed spam classification models based on imbalanced data where spam emails form the rare class with only 16.5% of the total emails. Different metrics were utilized in the evaluation of the developed models. Results show noticeable improvement of spam classification models when trained by dataset that includes malicious related features.

- **Improving Knowledge Based Spam Detection Methods: The Effect of Malicious Related Features in Imbalance Data Distribution Ja'far Alqatawna, Hossam Faris, Khalid Jaradat, Malek Al-Zewairi, Omar Adwan King Abdullah II School for Information Technology, The University of Jordan, Amman, Jordan.**

A summary of document is a shorter text conveys the most important information from the sources .Summary of the text must contains important information from the documents. This paper presents the design and implementation of a system to summarize e mail messages. The system uses the subject and contents of the e mail message to classify e mails based on user's activities and generate summary of each incoming message.

- **Effective Spam Detection Method for Email Savita Teli1, Santoshkumar Biradar2 1(Student, Dept of Computer Engg, Dr. D. Y. Patil College of Engg, Ambi, University of Pune, M.S, India) 2 (Asst. Proff, Dept of Computer Engg, Dr. D. Y. Patil College of Engg, Ambi, University of Pune, M.S, India)**

We propose a unified neural network architecture and learning algorithm that can be applied to various natural language processing tasks including part-of-speech tagging, chunking, named entity recognition, and semantic role labeling. This versatility is achieved by trying to avoid task-specific engineering and therefore disregarding a lot of prior knowledge. Instead of exploiting man-made input features carefully optimized for each task, our system learns internal representations on the basis of vast amounts of mostly unlabeled training data. This work is then used as a basis for building a freely available tagging system with good performance and minimal computational requirements.

3 SYSTEM DEVELOPMENT

3.1 Tools and Technologies used

1. Eclipse IDE for Java EE developers

Eclipse is an integrated development environment (IDE) used in computer programming. It contains a baseworkspace and an extensible plug-in system for customizing the environment. Eclipse is written mostly in Java and its primary use is for developing Java applications. It can also be used to develop packages for the software Mathematica. Development environments include the Eclipse Java development tools (JDT) for Java and Scala, Eclipse CDT for C/C++ and Eclipse PDT for PHP, among others.

Eclipse supports development for Tomcat, GlassFish and many other servers and is often capable of installing the required server (for development) directly from the IDE. It supports remote debugging, allowing the user to watch variables and step through the code of an application that is running on the attached server.

Eclipse supports a rich selection of extensions, adding support for Python via pydev, Android development via Google's ADT, JavaFX support via eclipse, and many others at the Eclipse Marketplace, as well as JavaScript and jQuery.

Eclipse PDT (PHP Development Tools)

The PHP (PHP hypertext Preprocessor) Development Tools project provides a PHP Development Tools framework for the Eclipse platform. The project encompasses all development components, including code-completion, develop PHP and facilitate extensibility. It leverages the existing Eclipse Web Tools Platform (WTP) and Dynamic Languages Toolkit (DLTK).

Eclipse ADT (Android Development Tools)

Android Development Tools (ADT) is a Google-provided plugin for the Eclipse IDE that is designed to provide an integrated environment in which to build Android applications. ADT extends the capabilities of Eclipse to let developers set up new Android projects, create an application UI, add packages based on the Android Framework API, debug their applications using the Android SDK tools, and export signed (or unsigned) .apk files in order to distribute their applications. It is free download. It was the official IDE for Android but was replaced by Android Studio.

Tools for Java developers creating Java EE and Web applications, including a Java IDE, tools for Java EE, JPA, JSF, Mylyn, EGit and others.

This package includes:

- Data Tools Platform
- Eclipse Git Team Provider
- Eclipse Java Development Tools
- Eclipse Java EE Developer Tools
- JavaScript Development Tools
- Maven Integration for Eclipse
- Mylyn Task List
- Eclipse Plug-in Development Environment
- Remote System Explorer
- Eclipse XML Editors and Tools

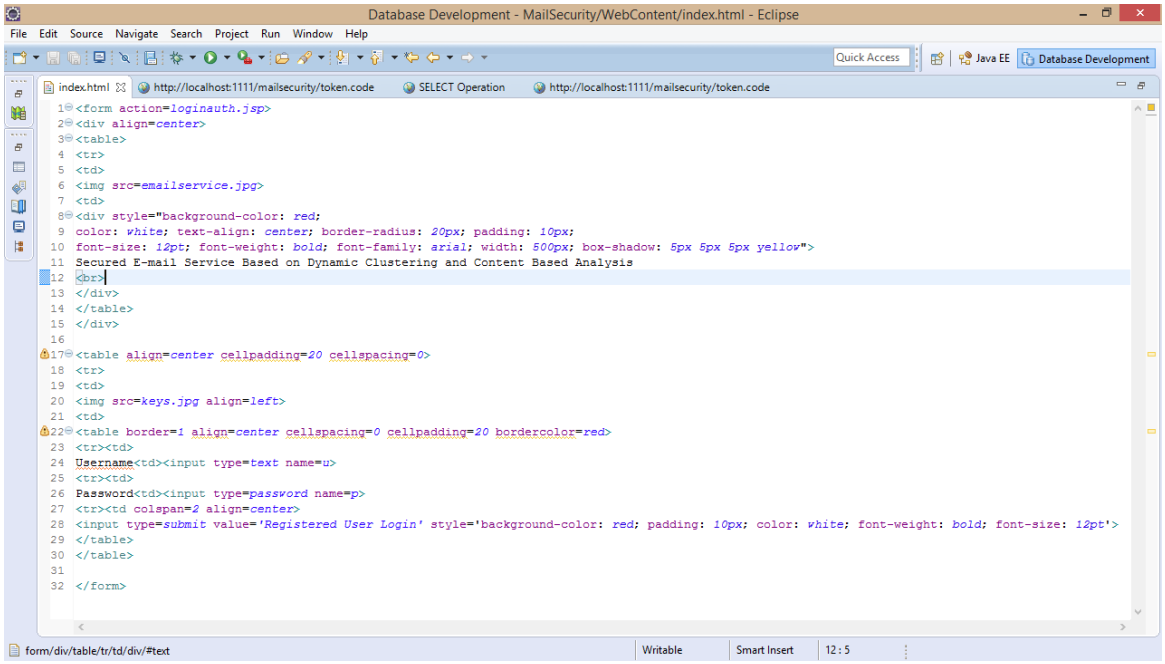


Figure 5: Eclipse IDE

2. Apache-tomcat-7.0.34

Apache Tomcat, often referred to as Tomcat, is an open-source web server developed by the Apache Software Foundation (ASF). Tomcat implements several Java EE specifications including Java Servlet, JavaServer Pages (JSP),Java EL, and WebSocket, and provides a "pure Java" HTTP web server environment in which Java code can run.

Tomcat is developed and maintained by an open community of developers under the auspices of the Apache Software Foundation, released under the Apache License 2.0 license, and is open-source software.

Tomcat 7.x implements the Servlet 3.0 and JSP 2.2 specifications. It requires Java version 1.6, although previous versions have run on Java 1.1 through 1.5. Versions 5 through 6 saw improvements in garbage collection, JSP parsing, performance and scalability. Native

wrappers, known as "Tomcat Native", are available for Microsoft Windows and Unix for platform integration.

Tomcat is an application server from the Apache Software Foundation that executes Java servlets and renders Web pages that include Java Server Page coding. Described as a "reference implementation" of the Java Servlet and the Java Server Page specifications, Tomcat is the result of an open collaboration of developers and is available from the Apache Web site in both binary and source versions. Tomcat can be used as either a standalone product with its own internal Web server or together with other Web servers, including Apache, Netscape Enterprise Server, Microsoft Internet Information Server (IIS), and Microsoft Personal Web Server. Tomcat requires a Java Runtime Enterprise Environment that conforms to JRE 1.1 or later.

3. Wamp Server

Stands for "Windows, Apache, MySQL, and PHP." WAMP is a variation of LAMP for Windows systems and is often installed as a software bundle (Apache, MySQL, and PHP). It is often used for web development and internal testing, but may also be used to serve live websites.

WampServer refers to a software stack for the Microsoft Windows operating system, created by Romain Bourdon and consisting of the Apache web server, OpenSSL for SSL support, MySQL database and PHP programming language.

The most important part of the WAMP package is Apache (or "Apache HTTP Server") which is used run the web server within Windows. By running a local Apache web server on a Windows machine, a web developer can test webpages in a web browser without publishing them live on the Internet.

WAMP also includes MySQL and PHP, which are two of the most common technologies used for creating dynamic websites. MySQL is a high-speed database, while PHP is a scripting language that can be used to access data from the database. By installing these two components locally, a developer can build and test a dynamic website before publishing it to a public web server.

While Apache, MySQL, and PHP are open source components that can be installed individually, they are usually installed together. One popular package is called "WampServer," which provides a user-friendly way to install and configure the "AMP" components on Windows.

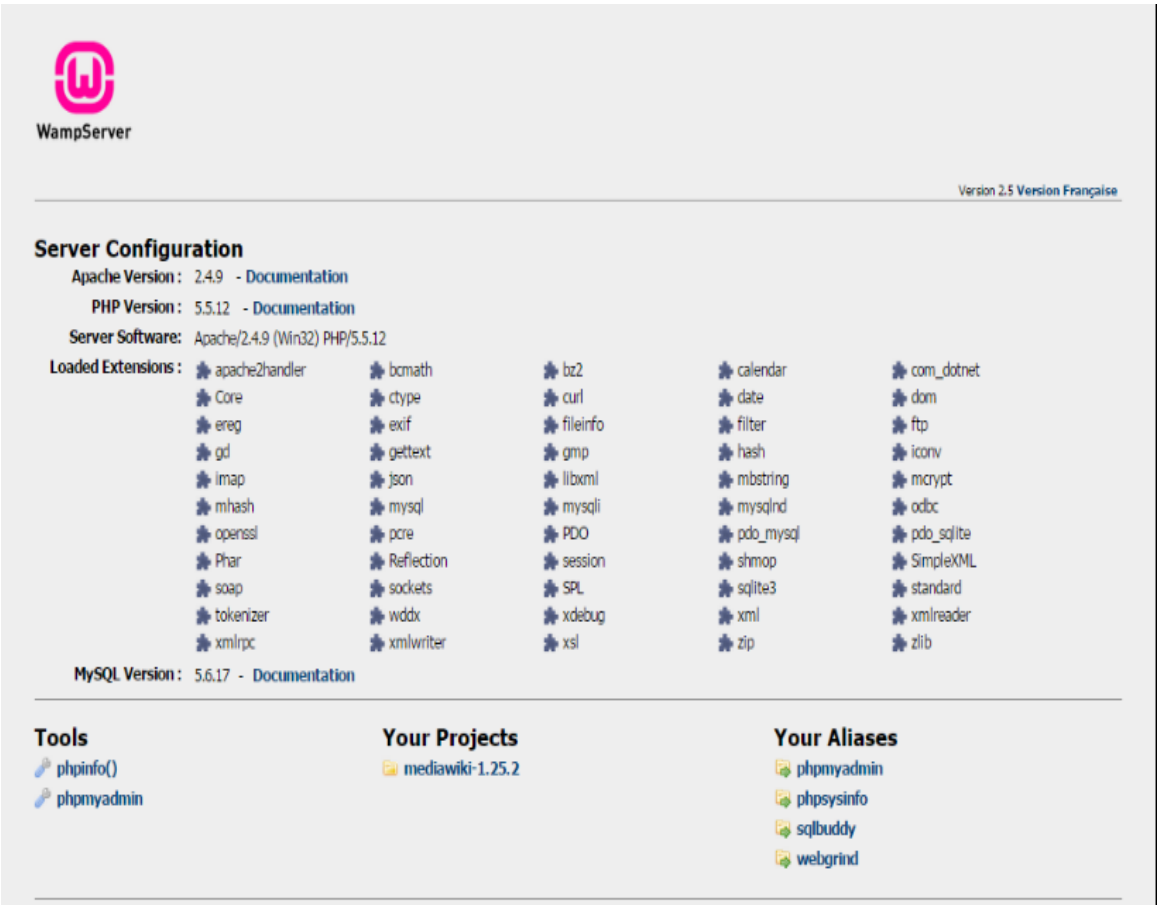


Figure 6: Wamp Server

4. Mysql-connector-java 5.1.38

MySQL Connector/J is the official JDBC driver for MySQL.
MySQL provides connectivity for client applications developed in the Java programming language with MySQL Connector/J, a driver that implements the Java Database Connectivity (JDBC) API.

For large-scale programs that use common design patterns of data access, consider using one of the popular persistence frameworks such as Hibernate, Spring's JDBC templates or Ibatis SQL Maps to reduce the amount of JDBC code for you to debug, tune, secure, and maintain.

Driver/Datasource Class Names, URL Syntax and Configuration Properties for Connector

The name of the class that implements java.sql.Driver in MySQL Connector/J is com.mysql.jdbc.Driver. Theorg.gjt.mm.mysql.Driver class name is also usable for backward compatibility with MM.MySQL, the predecessor of Connector/J. Use this class name when registering the driver, or when configuring a software to use MySQL Connector.

- **JDBC URL format**

The general format for a JDBC URL for connecting to a MySQL server is as follows, with items in square brackets ([ ]) being optional:

jdbc:mysql://[***host1***][:***port1***][,[***host2***][:***port2***]]...[/[***database***]] »
[?***propertyName1***=***propertyValue1***[&***propertyName2***=***propertyValue2***]...]

- **Host and Port**

If no hosts are not specified, the host name defaults to 127.0.0.1. If the port for a host is not specified, it defaults to 3306, the default port number for MySQL servers.

- **Initial database for connection**

If the database is not specified, the connection is made with no default database. In this case, either call the setCatalog() method on the Connection instance, or fully specify table names using the database name (that is, SELECT dbname.tablename.colname FROM dbname.tablename...) in your SQL. Opening a connection without specifying the database to use is generally only useful when building tools that work with multiple databases, such as GUI database managers.

- **Setting configuration properties**

Configuration properties define how Connector/J will make a connection to a MySQL server. Unless otherwise noted, properties can be set for a DataSource object or for a Connection object.
Configuration properties can be set in one of the following ways:

- Using the set*() methods on MySQL implementations of java.sql.DataSource (which is the preferred method when using implementations of java.sql.DataSource):

  - com.mysql.jdbc.jdbc2.optional.MysqlDataSource
  - com.mysql.jdbc.jdbc2.optional.MysqlConnectionPoolDataSource

  As a key/value pair in the java.util.Properties instance passed to DriverManager.getConnection() or Driver.connect()

  As a JDBC URL parameter in the URL given to java.sql.DriverManager.getConnection(), java.sql.Driver.connect() or the MySQL implementations of the javax.sql.DataSource setURL() method. If you specify a

configuration property in the URL without providing a value for it, nothing will be set; for example, adding useServerPrepStmts alone to the URL does not make Connector/J use server-side prepared statements; you need to add useServerPrepStmts=true.

- **Connection/Authentication**

Properties and Descriptions
User The user to connect as Since version: all versions
Password The password to use when connecting Since version: all versions
socketFactory The name of the class that the driver should use for creating socket connections to the server. This class must implement the interface 'com.mysql.jdbc.SocketFactory' and have public no-args constructor. Default: com.mysql.jdbc.StandardSocketFactory Since version: 3.0.3
connectTimeout Timeout for socket connect (in milliseconds), with 0 being no timeout. Only works on JDK-1.4 or newer. Defaults to '0'. Default: 0

Properties and Descriptions
Since version: 3.0.1
socketTimeout Timeout on network socket operations (0, the default means no timeout). Default: 0 Since version: 3.0.1
connectionLifecycleInterceptors A comma-delimited list of classes that implement "com.mysql.jdbc.ConnectionLifecycleInterceptor" that should notified of connection lifecycle events (creation, destruction, commit, rollback, setCatalog and setAutoCommit) and potentially alter the execution of these commands. ConnectionLifecycleInterceptors are "stackable", more than one interceptor may be specified via the configuration property as a comma-delimited list, with the interceptors executed in order from left to right. Since version: 5.1.4
useConfigs Load the comma-delimited list of configuration properties before parsing the URL or applying user-specified properties. These configurations are explained in the 'Configurations' of the documentation. Since version: 3.1.5
authenticationPlugins Comma-delimited list of classes that implement com.mysql.jdbc.AuthenticationPlugin and which will be used for authentication unless disabled by "disabledAuthenticationPlugins" property.

Properties and Descriptions
Since version: 5.1.19
defaultAuthenticationPlugin Name of a class implementing com.mysql.jdbc.AuthenticationPlugin which will be used as the default authentication plugin (see below). It is an error to use a class which is not listed in "authenticationPlugins" nor it is one of the built-in plugins. It is an error to set as default a plugin which was disabled with "disabledAuthenticationPlugins" property. It is an error to set this value to null or the empty string (i.e. there must be at least a valid default authentication plugin specified for the connection, meeting all constraints listed above). Default: com.mysql.jdbc.authentication.MysqlNativePasswordPlugin Since version: 5.1.19
disabledAuthenticationPlugins Comma-delimited list of classes implementing com.mysql.jdbc.AuthenticationPlugin or mechanisms, i.e. "mysql_native_password". The authentication plugins or mechanisms listed will not be used for authentication which will fail if it requires one of them. It is an error to disable the default authentication plugin (either the one named by "defaultAuthenticationPlugin" property or the hard-coded one if "defaultAuthenticationPlugin" property is not set). Since version: 5.1.19
disconnectOnExpiredPasswords If "disconnectOnExpiredPasswords" is set to "false" and password is expired then server enters "sandbox" mode and sends ERR(08001, ER_MUST_CHANGE_PASSWORD) for all commands that are not needed to set a new password until a new password is set. Default: true

Properties and Descriptions
Since version: 5.1.23
interactiveClient Set the CLIENT_INTERACTIVE flag, which tells MySQL to timeout connections based on INTERACTIVE_TIMEOUT instead of WAIT_TIMEOUT Default: false Since version: 3.1.0
localSocketAddress Hostname or IP address given to explicitly configure the interface that the driver will bind the client side of the TCP/IP connection to when connecting. Since version: 5.0.5
propertiesTransform An implementation of com.mysql.jdbc.ConnectionPropertiesTransform that the driver will use to modify URL properties passed to the driver before attempting a connection Since version: 3.1.4
useCompression Use zlib compression when communicating with the server (true/false)? Defaults to 'false'. Default: false Since version: 3.0.17

- **Connection properties**

THESE MYSQL DATA TYPES	CAN ALWAYS BE CONVERTED TO THESE JAVA TYPES
CHAR, VARCHAR, BLOB, TEXT, ENUM, and SET	java.lang.String, java.io.InputStream, java.io.Reader, java.sql.Blob, java.sql.Clob
FLOAT, REAL, DOUBLE PRECISION, NUMERIC, DECIMAL, TINYINT, SMALLINT, MEDIUMINT, INTEGER, BIGINT	java.lang.String, java.lang.Short, java.lang.Integer, java.lang.Long, java.lang.Double, java.math.BigDecimal
DATE, TIME, DATETIME, TIMESTAMP	java.lang.String, java.sql.Date, java.sql.Timestamp

3 PERFORMANCE ANALYSIS

3.1 OUTPUT SCREENSHOTS

- First user logging in to the system

Figure 7: First user logging in to the system

- First user sending mail to second user with span content, for example, with the word “award”

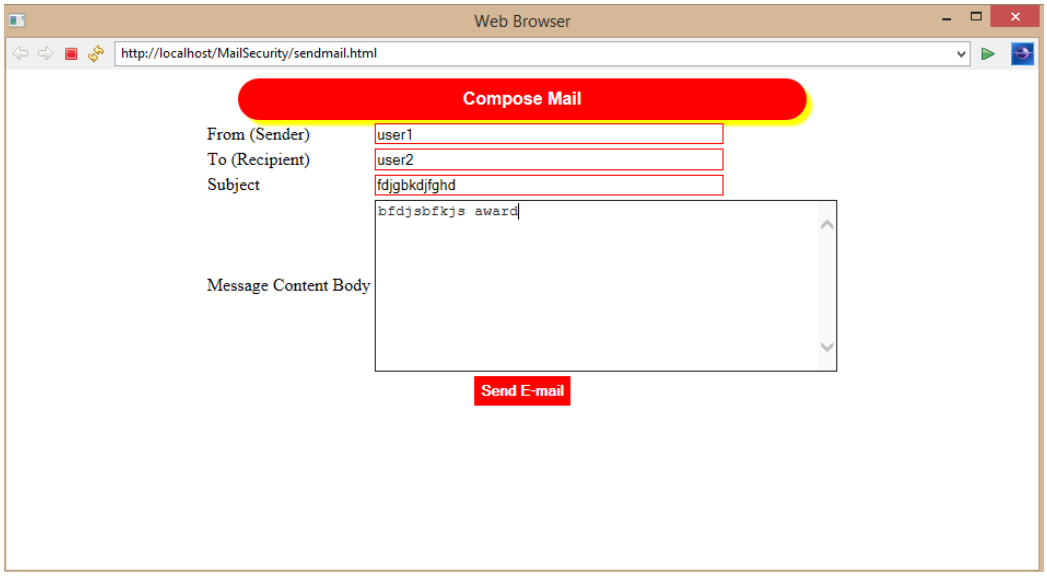


Figure 8: First user sending mail to second user with span content

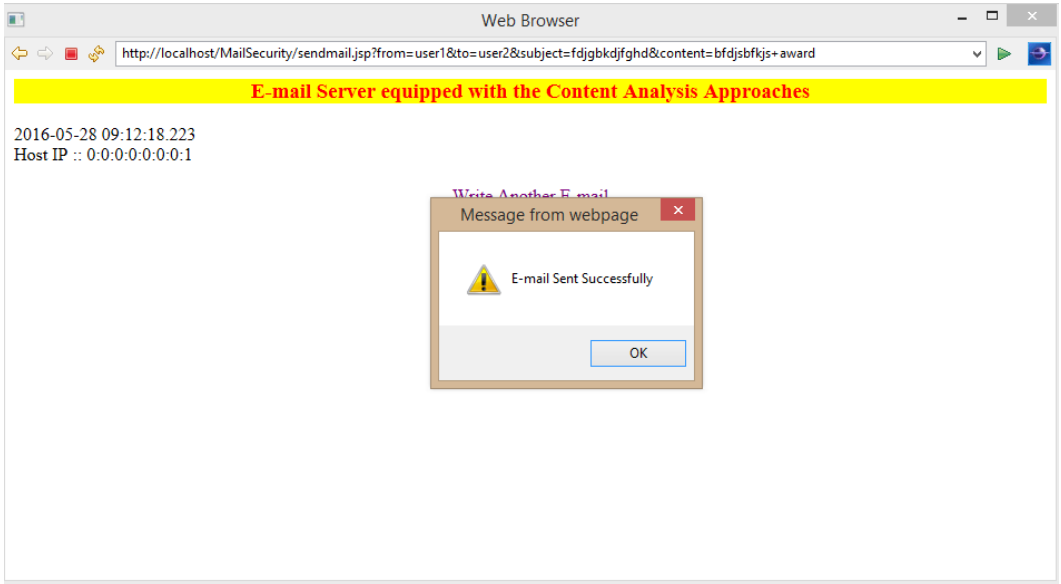


Figure 9: First user sending mail to second user with span content

- Now the second user is logging in to the system

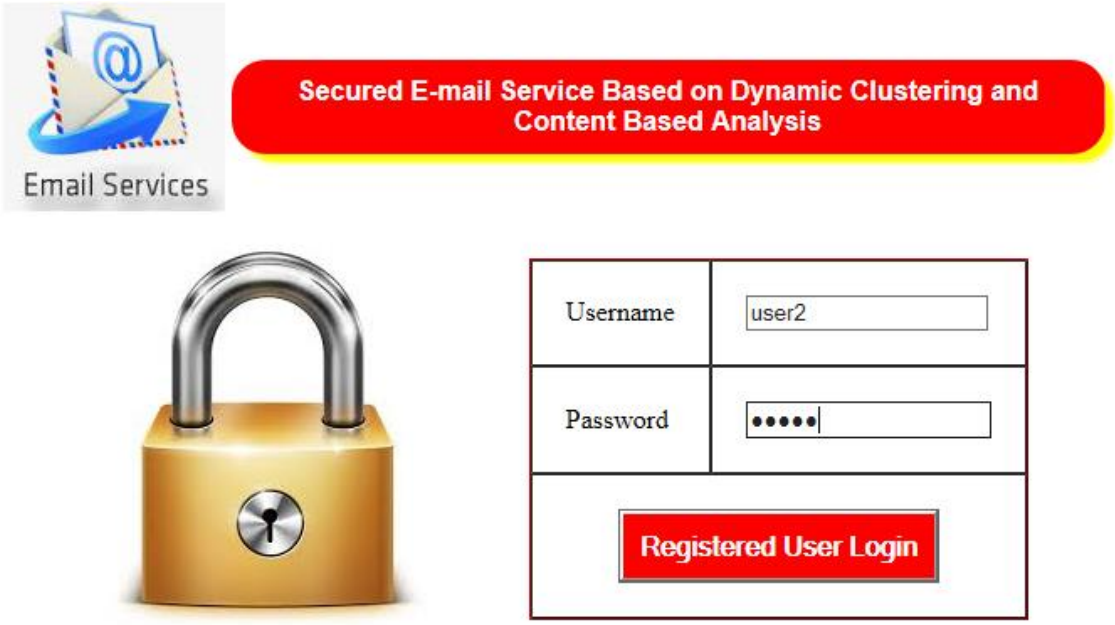
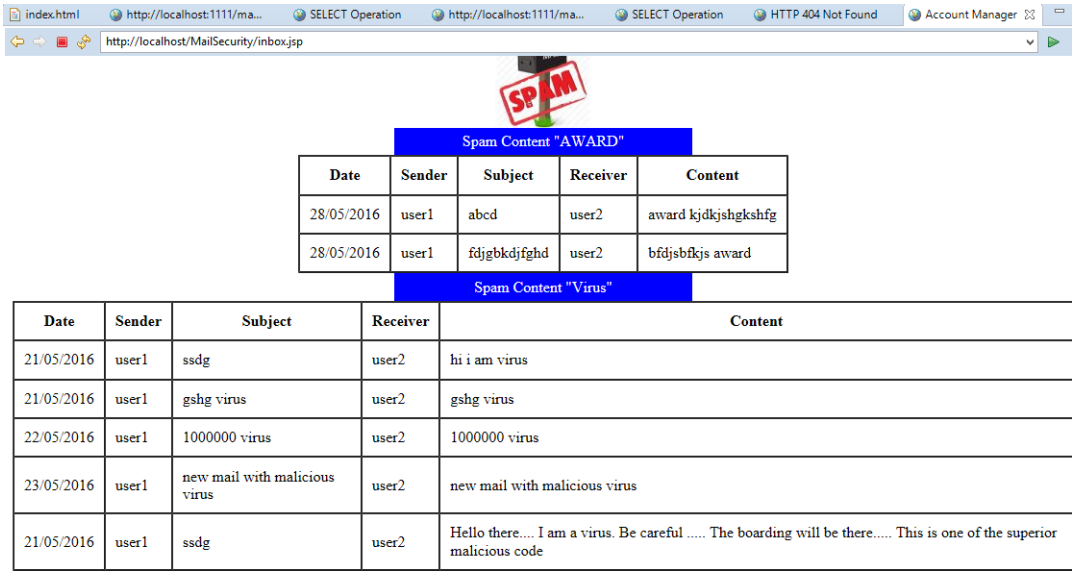


Figure 10: Second user logging in to the system

- The mail from the first user will go to the spam folder because it contains spam content to prevent the viruses and for the security of the system



Date	Sender	Subject	Receiver	Content
28/05/2016	user1	abcd	user2	award kjdkjshgkshfg
28/05/2016	user1	fdjgbkdjfghd	user2	bfdjsbfkjs award

Date	Sender	Subject	Receiver	Content
21/05/2016	user1	ssdg	user2	hi i am virus
21/05/2016	user1	gshg virus	user2	gshg virus
22/05/2016	user1	1000000 virus	user2	1000000 virus
23/05/2016	user1	new mail with malicious virus	user2	new mail with malicious virus
21/05/2016	user1	ssdg	user2	Hello there.... I am a virus. Be careful ..... The boarding will be there..... This is one of the superior malicious code

Figure 11: Spam content

- This is how the content of any email can be summarized and useful information can be extracted to detect the spam content

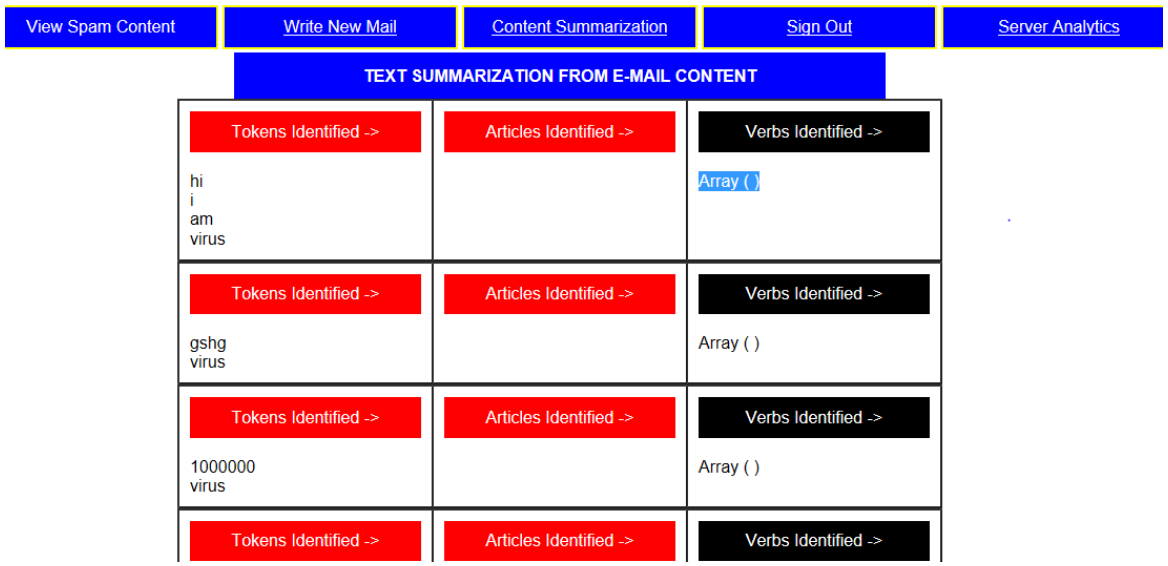


Figure 12: Text Summarization

- This is how the IP address can be detected from where the mail was being sent

TimeStamp	IP Addresse	Execution Time (MilliSeconds)
21/05/2016		
21/05/2016	127.0.0.1	
22/05/2016	127.0.0.1	11674604
23/05/2016	0:0:0:0:0:0:0:1	31
21/05/2016		
21/05/2016	127.0.0.1	
22/05/2016	127.0.0.1	11674604
28/05/2016	0:0:0:0:0:0:0:1	99

Figure 13: IP Address

4 CONCLUSION AND FUTURE WORK

Internet is mainly used by Individuals, Co-operatives and Governments. They have send information through internet. But there is a possibility to hack the information. So to protect information, we need to encrypt/decrypt information. Techniques employed in recent years are very similiar to the classical ones but they have to be adapted to each particular kind of system and its objectives. Improvements in machine learning techniques have allowed that they can be used to train and develop summarization systems these days as well.

Throughout the recent years summarization has experienced a remarkable evolution. Due to the evaluation programmes that take place every year, the field of Text Summarization has been improved considerably. For example, the tasks performed in The Document Understand Conferences (DUC) have changed from simple tasks to more complex ones. At the beginning, efforts were done to generate simple extracts from single documents usually in English. Lately, the trend has evolved to generate more sophisticated summaries such as abstracts from a number of documents, not just a single one, and in a variety of languages. Different tasks have been introduced year after year so that, apart from the gereral main task, it is possible to find taks consisting of producing summaries from a specific question or user-need, or just to generate a summary from updated news.

Research on this field will continue due to the fact that text summarization task has not been finished yet and there is still much effort to do, to investigate and to improve. Definition, types, different approaches and evaluation methods have been exposed as well as summarization systems features and techniques already developed. In the future we plan to contribute to improve this field by means of improving the quality of summaries, and studying the influence of other neighbour tasks techniques on summarization.

The biggest challenge for text summarization is to summarize content from a number of textual and semi structured sources, including databases and web pages, in the right way (language, format, size, time) for a specific user. The text summarization software should produce the effective summary in less time and with least redundancy. Summaries can be evaluated using intrinsic or extrinsic measures. While intrinsic methods attempt to measure summary quality using human evaluation and extrinsic methods measure the same through a task based performance measure such the information retrieval oriented task.

5 REFERENCES

[1] Email Classification and Summarization: A Machine Learning Approach Taiwo Ayodele Rinat Khusainov David Ndzi Department of Electronics and Computer Engineering University of Portsmouth, United Kingdo {taiwo.ayodele, rinat.khusainov, david.ndzi}@port.ac.ukhttps://www.researchgate.net/publication/224386748_Email_classification_and_summarization_A_machine_learning_approach [accessed May 28, 2016].

[2] A Survey of Text Summarization Extractive Techniques Vishal Gupta University Institute of Engineering & Technology, Computer Science & Engineering, Panjab University Chandigarh, India, Email: vishal@pu.ac.in Gurpreet Singh Lehal Department of Computer Science, Punjabi University Patiala, Punjab, India, Email: gslehal@yahoo.com

[3] Email Summarization-Extracting Main Content from the Mail Mubashir Alam1 ,Mohit Kakkar2 1 M.Tech Student, Dept. of CSE, Desh Bhagat University, Mandi Gobindgarh, Punjab, India 2Assistant Professor, Dept. of CSE, Desh Bhagat University, Mandi Gobindgarh, Punjab, India

[4] Survey of Unstructured Text Summarization Techniques Sherif Elfayoumy School of Computing University of North Florida Jacksonville, Florida Jenny Thoppil School of Computing University of North Florida Jacksonville, Florida.

[5] TEXT SUMMARIZATION : AN OVERVIEW Elena Lloret Dept. Lenguajes y Sistemas Inform´aticos Universidad de Alicante Alicante, Spain elloret@dlsi.ua.es

[6] Effective Spam Detection Method for Email Savita Teli1, Santoshkumar Biradar2 1(Student, Dept of Computer Engg, Dr. D. Y. Patil College of Engg, Ambi, University of Pune, M.S, India) 2 (Asst. Proff, Dept of Computer Engg, Dr. D. Y. Patil College of Engg, Ambi, University of Pune, M.S, India)

[7] Improving Knowledge Based Spam Detection Methods: The Effect of Malicious Related Features in Imbalance Data Distribution Ja'far Alqatawna, Hossam Faris, Khalid Jaradat, Malek Al-Zewairi, Omar Adwan King Abdullah II School for Information Technology, The University of Jordan, Amman, Jordan Email: J.Alqatawna@ju.edu.jo

[8] Regression-BasedSummarizationofEmailConversationsJanUlrichand GiuseppeCarenini and GabrielMurray and RaymondNg {ulrichj, carenini, gabrielm, rng}@cs.ubc.ca Department of Computer Science University of British Columbia, Canada.

[9] Summarizing Emails with Conversational Cohesion and SubjectivityGiuseppe Carenini, Raymond T. Ng and Xiaodong Zhou Department of Computer Science University of British Columbia Vancouver, BC, Canada {carenini, rng, xdzhou}@cs.ubc.ca

[10] Summarizing Email Conversations with Clue Words Giuseppe Carenini, Raymond T. Ng, Xiaodong Zhou Department of Computer Science University of British Columbia, Canada {carenini, rng, xdzhou}@cs.ubc.ca

[11] Natural Language Processing Gobinda G. Chowdhury Dept. of Computer and Information Sciences University of Strathclyde, Glasgow G1 1XH, UK e-mail: gobinda@dis.strath.ac.uk

[12] Special issue: Natural Language Processing and its Applications Research in Computing Science Series Editorial Board Comité Editorial de la Serie

.[13] Natural Language Processing (Almost) from Scratch Ronan Collobert JasonWeston†Leon Bottou Michael Karlen Koray Kavukcuoglu§ Pavel Kuksa¶ PKUKSA@CS.RUTGERS.EDU NEC Laboratories America 4 Independence Way Princeton, NJ 08540