DEVELOPMENT OF AN IMPROVED BAYESIAN SPAM FILTERING MODEL FOR ZIPPED ATTACHMENT SPAM FILES

[1]Odeniyi, O. A. [2]Kareem A. E. A. [3]Sarumi, O. A.
[4]Lawal, N. T.A. & [5]Akinwumi, A.R.
Department of Computer Science,
Osun State College of Technology, Esa-Oke, Nigeria.
Corresponding author's E-mail:odeniyioa@oscotechesaoke.edu.ng

ABSTRACT

Spam filtering system affords its users the opportunity to detect spam emails. Early Spam filtering system were only text-based; however, spammers have moved to more sophisticated spamming techniques that involve zipped files now generally termed zipped based spam. In most zipped-based spam, the entire spam message, which could be sometimes text, is embedded in any zipped document. This type of spam emails creates another dimension to the spam filtering problem scenario. Extracting text from the zipped file and filtering these text components is one method that has been used to deal with zipped spam with little success because Spammers modify their approaches to beat such filters even when such filters are based on Optical Character Recognition. An improved Bayesian Spam filtering model which uses pattern recognition with Bayesian algorithm and which analyses the extensions feature of an attached file in addition to the statistical based filtering system of Bayesian model is developed and presented in this paper. The developed improved Bayesian Spam filtering model was implemented using PHP (for the Web scripting) and HTML & CSS (for the User Interface design. Questionnaires were administered to test the user assessment of the developed model and the evaluation of the efficacy of the developed improved Bayesian Spam filtering model in terms of users' assessment based on the Likert item scale questionnaires' responses was done. The developed model was assessed by 100 users. The major metrics considered are ease to use (ETU), ease to learn (ETL), effectiveness of filtering (EOF), reduced reading time (RRT). The overall analysis of the developed system showed that the Response Mean of ETU, ETL, EOF and RRT were 4.00, 4.02, 3.84 and 3.92 respectively on a rating scale of 5. The overall evaluation of the system revealed that the system does not require much technical know-how; it is highly effective and has high degree of relevance. The developed model has greatly offered a great assistance in reducing the scourge of zip attachment-based spam e-mails.

***Keywords:** Spam filtering system, Spammers, Emails, Spam emails, Zipped files.*

1. INTRODUCTION

The internet has become an integral part of everyday's life and e-mail has become a powerful tool for information exchange. Along with the growth of the Internet and e-mail,there has been a dramatic growth in spam (email spam) in recent years(Christina *et al*., 2010a; Rao *et al*., 2016).E-mail spam, known as unsolicited bulk Email(UBE), junk mail, or unsolicited commercial email (UCE), is the practice of sending unwanted e-mail messages, frequently with commercial content, in large quantities to an indiscriminate set of recipients. (Christina *et al*., 2010a; Rao *et al*., 2016). Hence, email spams are unsolicited, unwanted or irrelevant large amount of messages sent by spammers through email in the mailboxes of users (Hassan, 2017). The inverse of "spam" email is called "ham" which is needed by recipients (Sao and Prashanthi, 2011).

Spam can originate from any location across the globe where Internet access is available. Despite the development of antispam services and technologies such as text or content-based, list-based, knowledge-based, clustering-based, machine learning-based, challenge/response system, collaborative content filtering, Domain Name System (DNS) lookup systems and many more (Christina *et al*., 2010b; Hassan, 2017; Bhuiyan *et al*., 2018), the number of spam messages continues to increase rapidly (Christina *et al*., 2010a). This is not because the filters are not powerful enough, it is due to the swift adoption of new techniques by the spammers and the inflexibility of spam filters to adapt to the changes. Until recent past, this problem was only stuck to text-basedspam mail. Presently, spammers have taken a new approach; where apart from sending the spams by text form; they send it via image files like .jpg, .png, or .gif formats and zip file attachment that contains different file formats like (*.doc, *.pdf, *.exe, *.ppt, jpeg, png, among others) or computer generated words that make reader gets angry (Sen *et al*., 2017).

Zip file attachment spam (ZIP) "is a kind of spam in which the text, image, message is embedded into attached zip file to defeat spam filtering techniques based on the analysis of e-mail's body text" (Biggio *et al*, 2007). Conventional text based spam filtering cannot handle the zip file attachment spam, thus they pass through the filters successfully. In spite of the existence of large number of methods and techniques available to filter spam, the volume of spam, particularly zip file attachment spam on the internet is still rising (More and Kulkarni, 2013); and still poses a big threat to all e-mail users and Internet Service Providers (ISPs).

Hence, this paper develops an improved Bayesian Spam filtering model which uses pattern recognition with Bayesian algorithm and which analyses the extensions feature of an attached file in addition to the statistical based filtering system of Bayesian model that effectively analyses and classifies zip attached e-mails; and subsequently enhances Internet Service Providers (ISPs) in combating the continuous growing spam phenomenon.

2. REVIEW OF LITERATURES

2.1 Email Spam and Anti-Spam Filtering System: An Overview

Literature establishes that 70% of today's emails are spam (Scholar, 2010; Mohammed *et al*., 2013; Harisinghaney *et al*., 2014). Most spam take the form of advertising or promotional materials, among which roughly half of all spam mails are related to money, debt reduction plans, getting-rich-quick schemes, gambling opportunities one third of spam mails are porn-based, health-related, and the rest of them cover a variety of topics in the way of promotion of products from companies or get rich quick schemes **(**Christina *et al*., 2010a**)**.

A Zip Attachment Spam(ZIP) file is a container for other files. ZIP has two features: bundling multiple files into one, and compressing them. These two features make the ZIP file format one of the most common ways that files and collections of files are shared around the internet. It is also one of the oldest archives and compression formats still in use, dating back to 1989 (Notenboom, 2014). Zip attachment spam (ZIP) e-mails are e-mails in which the spam mails or text are embedded inside zip folder. ZIP e-mails are used as Obfuscation as ZIP files are not blocked. ZIP files are used by spammers to trick the user into executing the compressed malware. ZIP are also used as Phishing Bait, as ZIP file format usually bypass anti-malware scans and other restrictions to deliver a malicious package (Notenboom, 2014).A large number of spam messages with .zip attachments was reported between October 2014 and First week of January, 2015 in Symantec's Global Intelligence Network (GIN). These .zip attachments had various file names followed by 10 hexadecimal characters (Notenboom, 2014).

The process of sending email spams by spammers usually involves collection of recipients' addresses on the web and sending the messages through domain's username using the assortment of procedures and instruments that incorporate spoofing, bonnets, open intermediaries, mail transfers, bulk mail instruments called mailers, and so forth (Drucker *et al*., 1999).

Several types of researches have been carried out on email filtering, some acquired good accuracy and some are still going on. Email filtering is a process to sort email according to some criteria. As various methods exist for email filtering, among them, inbound and outbound filtering is well known. Inbound filtering is the process to read a message from internet address and outbound filtering is to read the message from the local user. Moreover, the most effective and useful email filtering is Spam filtering which performs through antispam technique (Bhuiyan *et al*., 2018). As spammers are proactive in nature using dynamic spam structures which have been changing continuously and avoiding the anti-spam procedures; thus making spam filtering a challenging task (Sahami *et al*., 1998; Wang *et al*., 2006).

Spam filtering is a challenging undertaking for an assortment of reasons. For spam email, users are facing several problems such as spam emails which causes annoyance and waste user's time to regularly check and delete this large number of unwanted messages. Flooding of mailboxes with spam e-mails, wastes storage space and overload the server; thus it may lead to losing legitimate e-mails, degrading the server performance, or even make it totally unavailable. Hence, spam consumes network bandwidth and server storage space, waste users time and it is also a threat for user security (Sahami *et al*., 1998; Wang *et al*., 2006; Sharma *et al*., 2015). Hence, it is crucial to have automatic spam filtration system for every individual user.

Spam filtering is a process to automatically detect unsolicited message and prevent such from entering into user's inbox. In order for strict spam email, several methods or spam filtering system have been constructed by using various concepts and algorithms. Bhuiyan *et al*. (2018) described few of spam filtering methods to understand the process of spam filtering and its effectiveness. Bhuiyan *et* *al*. (2018) grouped these methods into standard spam filtering method, client side and enterprise level spam filtering methods and case base spam filtering method. Standard spam filtering process as depicted in Figure 1 follows some rules (steps) and acts as a classifier with sets of protocols to determine either the message is spam or not.

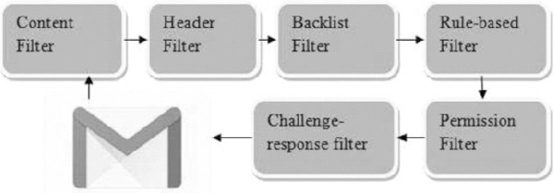


Figure 1: A standard process of Email spam filtering system (Bhuiyanet al., 2018)

Client level spam filtering provides some frameworks installed on personal computer for the individual client to secure mail transmission. This framework can interact with Mail User Agent (MUA) and filters the client inbox by composing, accepting and managing the messages. Enterprise level spam filtering is a process where provided frameworks are installed on mail server which interacts with the Mail Transfer Agent (MTA) for classifying the received messages or mail in order to categorize the spam message on the network. Figure 2 represents the method of clientside and enterprise level spam filtering

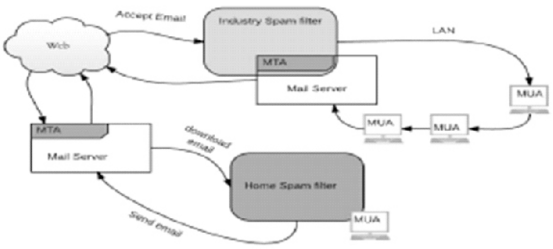


Figure 2: Client Side and Enterprise level Email spam filtering system (Cunningham et al., 2003; Bhuiyanet al., 2018)

Case base or sample base filtering is one of the prominent methods for Machine Learning Technique. The full process is performed through several steps illustrated by Figure 3.

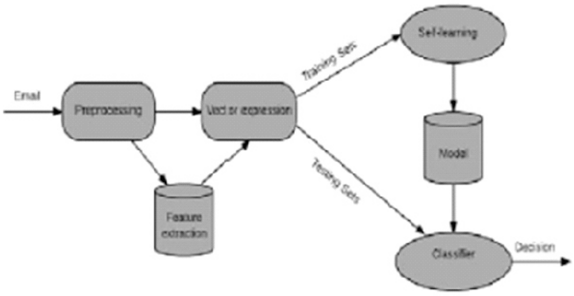


Figure 3: Case Base Spam Filtering System (Bhuiyan et al., 2018)

Conventionally, anti-spam filtering systems are divided into three components:

i. The first is a front-end module where the e-mail messages are parsed and tokenized.

ii. Classifier module in which a probability of being spam or legitimate is computed using the tokenized messages. Given this probability, the last component finally makes a decision based on the cost that one should pay from misclassification errors.

iii. In some cases, this last component involves getting users' feedback regarding the decision. Therefore, there might be some feedback paths back to the classifier module.

In order to filter and classify email spams, some processing has to be done in order to extract the text and then classify the texts so that they are filtered. E-mails have to be classified depending on the content of the e-mail. After the analysis and extraction of the features of each e-mail, what follows is to classify the e-mail as to whether it is a spam or ham mail. The classification can be achieved in various ways, but the most popular approach is unsupervised machine learning approach such as the Bayesian spam-filtering technique.

2.2 Bayesian Spam Filtering Approach

Bayesian spam filtering, considered the most advanced form of content-based filtering employs the laws of mathematical probability to determine which ones messages are legitimate and which are spam. It makes use of a naive Bayes classifier to identify spam e-mail. Bayesian classifiers work by correlating the use of tokens (typically words, or sometimes other things), with spam and non-spam e-mails and then using Bayesian inference to calculate a probability that an email is or is not spam (Sahami *et al*., 1998).

Bayesian spam filtering is a very powerful technique for dealing with spam, that can tailor itself to the email needs of individual users; and gives low false positive spam detection rates that are generally acceptable to users. 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, the 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

(Sahami *et al*., 1998).

After training, the word probabilities (also known as likelihood functions) are used to compute the probability that an email with a particular set of words in it belongs to either category (Spam or ham). Each word in the email contributes to the email's spam probability, or only the most interesting words. This contribution is called the posterior probability and is computed using Bayes' theorem. Then, the email's spam probability is computed over all words in the email, and if the total exceeds a certain threshold (say 95%), the filter will mark the email as a spam (Sahami *et al*., 1998).

As in any other spam filtering technique, email marked as spam can then be automatically moved to a "Junk" email folder, or even deleted outright. Some software implements quarantine mechanisms that define a time frame during which the user is allowed to review the software's decision. The initial training can usually be refined when wrong judgments from the software are identified (false positives or false negatives). This allows the software to dynamically adapt to the ever evolving nature of spam (Sahami *et al*., 1998).

Bayesian email filters take advantage of Bayes' theorem. Bayes' theorem is used several times in the context of spam:(i) A first time, to compute the probability that the message is spam, knowing that a given word appears in this message; (ii) A second time, to compute the probability that the message is spam, taking into consideration all of its words (or a relevant subset of them); (iii) Sometimes a third time, to deal with rare words (Deshpande et al., 2007).

In general, a database is created where extracted tokens are stored. When analyzing a new message, the message is split into tokens and each token is given a value according to the following criteria:

I. The frequency of the token in good messages that the filter has been trained on

ii. The frequency of the token in spam messages that the filter has been trained on

iii. The number of good messages the filter has been trained on

iv. The number of spam messages the filter has been trained on Bayesian filtering approach involves computing the following

(i) ***Computing the probability that a message containing a given word is spam***:

The formula used by the software to determine whether a message containing a given word is spam that is derived from Bayes' theorem is

$$\Pr(S|W) = \frac{\Pr(W|S)\cdot\Pr(S)}{\Pr(W|S)\cdot\Pr(S)+\Pr(W|H)\cdot\Pr(H)}$$

eqn. 1 where:

$\Pr(S|W)$ is the probability that a message is a spam, knowing that the given word, for example, "replica" is in it;

$\Pr(S)$ is the overall probability that any given message is spam;

$\Pr(W|S)$ is the probability that the given word"replica" appears in spam messages;

$\Pr(H)$ is the overall probability that any given message is not spam (is "ham");

$\Pr(W|H)$is the probability that the given word "replica"appears in ham messages.

(ii) The Spamicity of a Word

Recent statistics show that the current probability of any message being spam is 80%, at the very least: $\Pr(S) = 0.8; \Pr(H) = 0.2$

However, most Bayesian spam detection softwares make the assumption that there is no *a priori* reason for any incoming message to be spam rather than ham, and considers both cases to have equal probabilities of 50%: $\Pr(S) = 0.5; \Pr(H) = 0.5$

The filters that use this hypothesis are said to be "not biased", meaning that they have no prejudice regarding the incoming email. This assumption permits simplifying the general formula to:

$$\Pr(S|W) = \frac{\Pr(W|S)}{\Pr(W|S)+\Pr(W|H)}$$

eqn. 2

This is functionally equivalent to asking, "what percentage of occurrences of the given word, for example, "replica" appear in spam messages?". This quantity is called "spamicity" (or "spaminess") of the word "replica", and can be computed. The number $\Pr(W|S)$ used in this formula is approximated to the frequency of messages containing "replica" in the messages identified as spam during the learning phase.

Similarly, $\Pr(W|H)$ is approximated to the frequency of messages containing "replica" in the messages identified as ham during the learning phase. For these approximations to make sense, the set of learned messages needs to be big and representative enough. It is also advisable that the learned set of messages conforms to the 50% hypothesis about repartition between spam and ham, i.e. that the datasets of spam and ham are of same size.

Determining whether a message is spam or ham based only on the presence of the word "replica" is error-prone, which is why Bayesian spam software tries to consider several words and combine their spamicities to determine a message's overall probability of being spam.

(iii) Combining Individual Probabilities:

Most Bayesian spam filtering algorithms are based on formulas that are strictly valid (from a probabilistic standpoint) only if the words present in the message are independent events. This condition is not generally satisfied (for example, in natural languages like English the probability of finding an adjective is affected by the probability of having a noun), but it is a useful idealization, especially since the statistical correlations between individual words are usually not known. On this basis, the following formula is derived from Bayes' theorem:

$$p = \frac{p_1 p_2 \cdots p_N}{p_1 p_2 \cdots p_N + (1-p_1)(1-p_2)\cdots(1-p_N)}$$

eqn. 3

where:

p is the probability that the suspect message is spam;

p_1 is the probability $p(S|W_1)$ that it is a spam knowing it contains a first word (for example "replica");

p_2 is the probability $p(S|W_2)$ that it is a spam knowing it contains a second word (for example "watches");

p_N is the probability $p(S|W_N)$ that it is a spam knowing it contains an *N*th word (for example "home").

Spam filtering software based on this formula is sometimes referred to as a naive Bayes classifier. The result *p* is typically compared to a given threshold to decide whether the message is spam or not. If *p* is lower than the threshold, the message is considered as likely

(iv) Dealing with Rare Words

In the case a word has never been met during the learning phase, both the numerator and the denominator are equal to zero, both in the general formula and in the spamicity formula. The software can decide to discard such words for which there is no information available. More generally, the words that were encountered only a few times during the learning phase cause a problem, because it would be an error to trust blindly the information they provide. A simple solution is to simply avoid taking such unreliable words into account as well.

Applying again Bayes' theorem and assuming the classification between spam and ham of the emails containing a given word ("replica") is a random variable with beta distribution, some programs decide to use a corrected probability:

$$\Pr'(S|W) = \frac{s \cdot \Pr(S) + n \cdot \Pr(S|W)}{s+n}$$

eqn. 4

where: $\Pr'(S|W)$ is the corrected probability for the message to be spam, knowing that it contains a given word;

s is the *strength* we give to background information about incoming spam;

$\Pr(S)$ is the probability of any incoming message to be spam;

n is the number of occurrences of this word during the learning phase;

$\Pr(S|W)$ is the spamicity of this word.

This corrected probability is used instead of the spamicity in the combining formula.

$\Pr(S)$ can again be taken equal to 0.5, to avoid being too suspicious about incoming email. 3 is a good value for *s*, meaning that the learned corpus must contain more than 3 messages with that word to put more confidence in the spamicity value than in the default value.

Bayesian filtering learns easily, when it learns from new spam and new valid outbound mails, the Bayesian filter evolves and adapts to new spam techniques. Furthermore, as strong as Bayesian filtering is, it is very difficult to fool. This is very unlike a keyword filter. An advanced spammer who wants to trick a Bayesian filter can either use fewer words that usually indicate spam (GFI White Paper, 2009).

Lastly, Bayesian method for filtering of mail is multi-lingual and international. Because the Bayesian anti-spam filter is adaptive, it can be used for any language required. Most keyword lists are available in English only and are therefore quite useless in non-English-speaking regions. The Bayesian filter also takes into account certain languages deviations or the diverse usage of certain words in different areas, even if the same language is spoken. This intelligence enables such a filter to catch more spam" (GFI White Paper, 2009).

(v) Creating Bayesian Word Database

Sequel to filtering of spam mail using the Bayesian filtering method, a database needs to be created with the tokens of words collected from a sample of spam mail and valid mail (referred to as 'ham'). A *token* is an instance of a sequence of characters in some particular document that are grouped together as a useful semantic unit for processing. Next is the assignment of probability value to each word or token; this probability is based on calculations that take into account how often that word occurs in spam as opposed to legitimate mail (ham). This is done by analyzing the users' outbound mail and analyzing known spam: All the words and tokens in both pools of mail are analyzed to generate the probability that a particular word points to the mail being spam. Here is an example of how probability is calculated. If the word "mortgage" occurs in 400 of 3,000 spam mails and in 5 out of 300 legitimate e-mails, then its spam probability would be 0.8889 (that is, [400/3000] divided by [5/300 + 400/3000]).

After the two databases have been created, the word probabilities can then be calculated and the filter is ready for use. At the incoming of a new mail, it is broken down into words and the most relevant words that is, those that are most significant in identifying whether the mail is spam or not are singled out. From these words, the Bayesian filter calculates the probability of the new message being spam or not. If the probability is greater than a threshold, say 0.9, then the message is classified as spam. With Bayesian method, spam detection rates of over 99.7% can be achieved with a very low number of false positives. We chose Bayesian filters because if given appropriate time and training data, Bayesian filters can achieve a combination of extremely high accuracy rates with a low percentage of false positives (Chiemeke *et al*., 2007).

2.3 Pattern Recognition Approach

Bayesian filtering, which is a popular spam-filtering technique, is now becoming compromised as spammers have started using methods to weaken it. To a rough approximation, Bayesian filters rely on word probabilities. For example, if a message contains many words which are only used in spam, and few which are never used in spam, it is likely to be spam. To weaken Bayesian filters, some spammers, alongside the sales pitch, now include lines of irrelevant, random words, in a technique known as *Bayesian poisoning*. (E-mail spam, 2009). Spam can also be hidden inside a fake "Undelivered mail notification" which looks like the failure notices sent by a mail transfer agent (a "Mailer-Daemon") when it encounters an error. Hence, the need for pattern recognition approach. In a pattern-recognition system, characteristics or features of an unknown pattern are analyzed, and the pattern is placed or classified by the system into one of several classes. It is an error for the system to place the pattern into the wrong class. A simple speech recognizer might classify a spoken word as "yes" or "no", considering only these two possibilities or classes. On the other hand, a large-vocabulary speech recognizer determines which word in a dictionary has been spoken.

Pattern recognition is central to optical character recognition (OCR), in which a computer system analyzes a scanned document page and attempts to identify the characters in the page. If the characters are printed clearly, an OCR system will accurately recognize them.

In this work, pattern recognition helps a great deal in recognizing a zip attachment spam, with the extension (*.zip). It also helps in recognizing file format like (*.doc, *.pdf, *.exe, *.ppt, JPEG, PNG etc) or other computer generated words that makes reader gets angry.

2.4 Review of Existing Anti-Spam Filtering Systems

Significant effort has been directed toward developing anti-spam filtering system using several techniques to make email more efficient to the users. Surveys of such techniques can be found in Bhowmick and Hazarika (2016), Rao *et al* (2016), Hassan (2017), Bhuiyan *et al*. (2018) among others.

However, Byun *et al* (2009) developed an anti-spam filtering framework that combines text-based and image-based anti-spam filters. It is an incremental framework that starts by reducing mismatches between training and test data sets to resolve the problem of a lack of training data for legitimate e-mails that contain both text and images. Thereafter, the outputs of text-based and image-based filters are combined with the weights determined by a Bayesian framework.

The system follows what is related to the conventional framework, but there are three main differences. First, in addition to an e-mail message parser and a tokenizer at the front-end, there is an image analysis module in which distinctive features of spam images are extracted if the e-mail contains images. Second, an additional classifier that computes the probability for each attached image to be spam or legitimate is trained given such features. Lastly, the back-end module fuses the probabilities computed so that the final decision can be made. The diagram is illustrated in Figure 4:

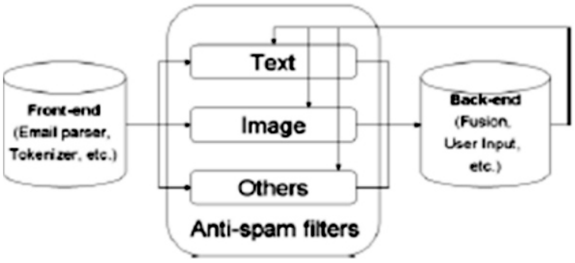


Figure 4: Block diagram of an Anti-Spam Filter Combination Framework for Text-and-Image E-mails through Incremental Learning(Source: Byun et al., 2009)

There is a feedback path given to the text-based and image-based filters from the back-end module. These paths provide the possibility of ensuring that the anti-spam filters in the middle will be trained incrementally as they see new samples. More anti-spam filters that exploit other features, such as structural information of e-mails, and so on can also be embedded.

Another method of fooling or distorting (original image or add colorful or noisy backgrounds so that only humans can identify the intended message) images in order to escape the filter is the use of the "Completely Automated Public Turing test to tell Computers and Humans Apart" (CAPTCHA). The technique is an effective tool in telling computers and humans apart. E-mail spammers create an image or many images using an image creation application and then proceed to make it obscure and apply randomization techniques to detect with computer vision algorithms. What the additional randomization does is to defeat the hash-based detection mechanisms (Zheng and Xiangjian, 2005).

Computer vision techniques have always been used in filtering noisy images but it has not been able to work against CAPTCHA. Researches carried out had been done to distinguish spam images from non-spam images by using four recognizing embedded test or monitoring color saturations in an image, but such methods tend to have high false positive rates. Furthermore, it is difficult to predict what spam images will look like as they are constantly evolving to evade detection. In addition, sophisticated computer vision techniques often require substantial CPU resources, making them less practical in high-volume environments.

In a publication by *Wu et al (*2006*), anti-spam image filtering was analyzed using* useful visual features. It is a necessity now to use visual information to achieve high accuracy for anti-spam filtering because such information is now evident in e-mails. However, one striking feature of the images used in spam e-mails is that they are artificially generated and contain embedded text. In achieving the set objectives, three classes of features were considered which are very helpful: (i) Embedded-text features (ii) Banner and graphic features, and (iii) Image location features

It was noted that what can really be of help is to have the idea whether the image contains any images or the area of the text region with the total image area. To derive such information, a text-in-image detector, whose responsibility is to detect the text region in an image is developed.

Banners and computer generated graphics form a great number of the images in spam e-mails and they are used for advertisements. While banner images are usually very narrow in width or height with large aspect ratio vertically or horizontally graphic, images in contrast, usually contain homogeneous background and very little texture. In a bid to extract graphics features, wavelet transformation is applied on the input images. Then the features are extracted in three orientations (vertical, horizontal, and diagonal) at fine resolution. A certain threshold value would have been set, and perhaps, if any of the extracted texture features falls below the threshold, then the image is likely to be a computer-generated graphic. Graphic features based on the detected number of graphic images are then calculated.

3. METHODOLOGY

The developed anti-spam filtering system is based on both the Anti-Spam Filter Combination Framework for Text-and-Image E-mails through Incremental Learning by Byun *et al*. (2009) and Generic Spam Filtering System framework by Conner (2008)shown in Figures4 and 5 respectively. The architecture for the developed anti-spam filtering system is as shown in Figure 6. In the extract module of the developed Zip Attachment Based Spam Filtering System architecture shown in Figure 6, the existing Bayesian Spam filtering model was improved by using pattern recognition technology with Bayesian algorithm to analyse the extensions feature of an attached file in addition to the statistical based filtering system of Bayesian model.

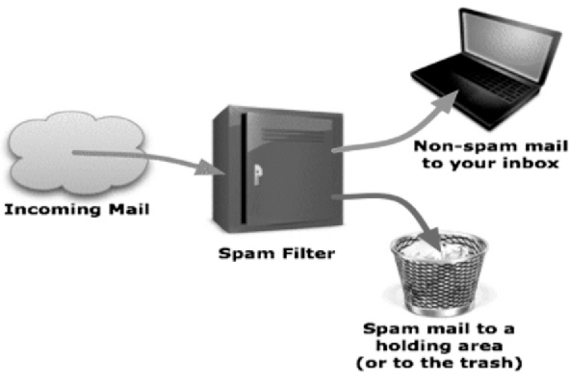

Incoming Mail
Spam Filter
Non-spam mail to your inbox
Spam mail to a holding area (or to the trash)

Figure5: Generic Spam Filtering System(Source: Conner, 2008)

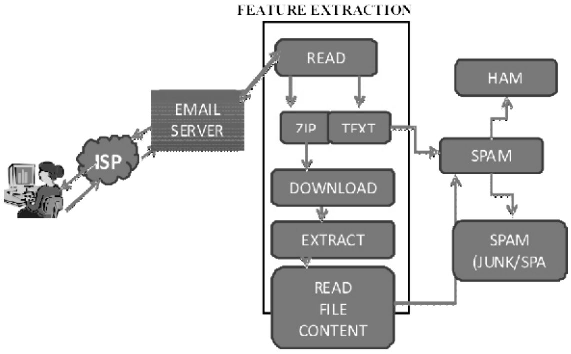


Figure 6: Developed Architectural Framework for Zip Attachment Spam Filtering System

The developed model consists different layers of functionality. It includes setting up a mail server for example, a Gmail server to receive the emails, a given e-mail address is read and search for zip attachment files which is the target for this framework. Any zip attachment files found in the e-mail is then downloaded into a temporary folder to extract the features of the zip attachments. The extracted features of this downloaded zip attachments are now read to display the file name and the file content. The file content (text) is then passed to the classifier, (Bayesian classifier) which will determine from the result whether the zip file attached is a spam or ham.

Questionnaires based on the Likert item scale of 1 to 5 were administered to collect users' objective assessment of the adequacy of the developed system. In terms of System Ease of Usage (SEU), System Effectiveness of Filtering (SEF) and System Degree of Relevance (SDR). Responses were received from a hundred users. The data from the duly filled questionnaires were captured, compiled and analyzed using Microsoft excel version 2016.

4. RESULT AND DISCUSSION

The developed improved Bayesian Spam filtering model was implemented using Hyper Text Markup Language (HTML) in the Microsoft Visual Studio Integrated Development Environment. The overall system (an application platform capable of sending and receiving test mails) was developed on the Microsoft.Net Framework using Visual Studio.Net (Visual C#) and MS SQL Server 2008. Bayesian aspect of Machine learning is employed to train the system. The system is of two parts, mail server side and clientside. Some of the graphical user interface of the developed system is depicted in Figures 7 – 10. The developed system was evaluated based on users' assessment by one hundred users.

Three major metrics which includes System Ease of Usage (SEU), System Effectiveness of Filtering (SEF) and System Degree of Relevance (SDR) were used for evaluation. The response mean of the SEU, SEF and SDR were 3.92, 4.02 and 3.84 respectively on a rating scale of 1 to 5 as depicted in Figure 10. This shows that the users find the system relatively easy to use the system, as the technical knowhow requirement to use the system has an appreciable degree of integrity and it is relevant in the delivery of secured and credible email systems.

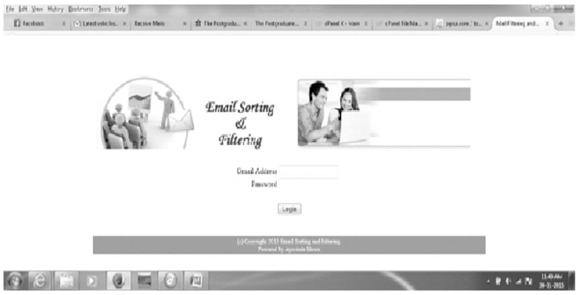

File Edit View History Bookmarks Tools Help
Facebook
Receive Mails
Email Sorting & Filtering
Email Address
Password
Login
(c)Copyright 2015 Email Sorting and Filtering
11:49 AM

Figure 7: Application Login Screen

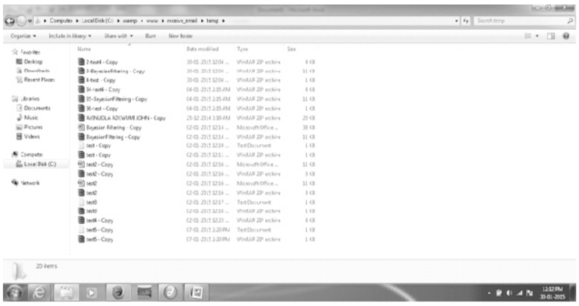
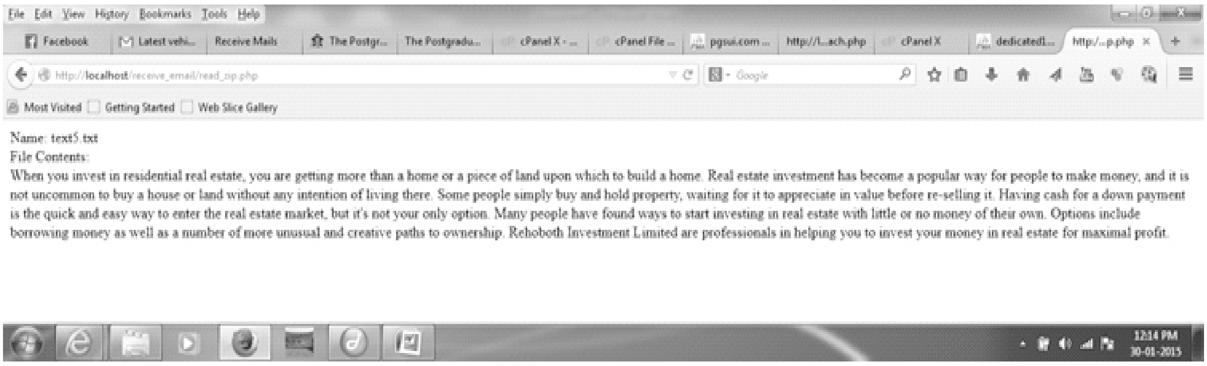
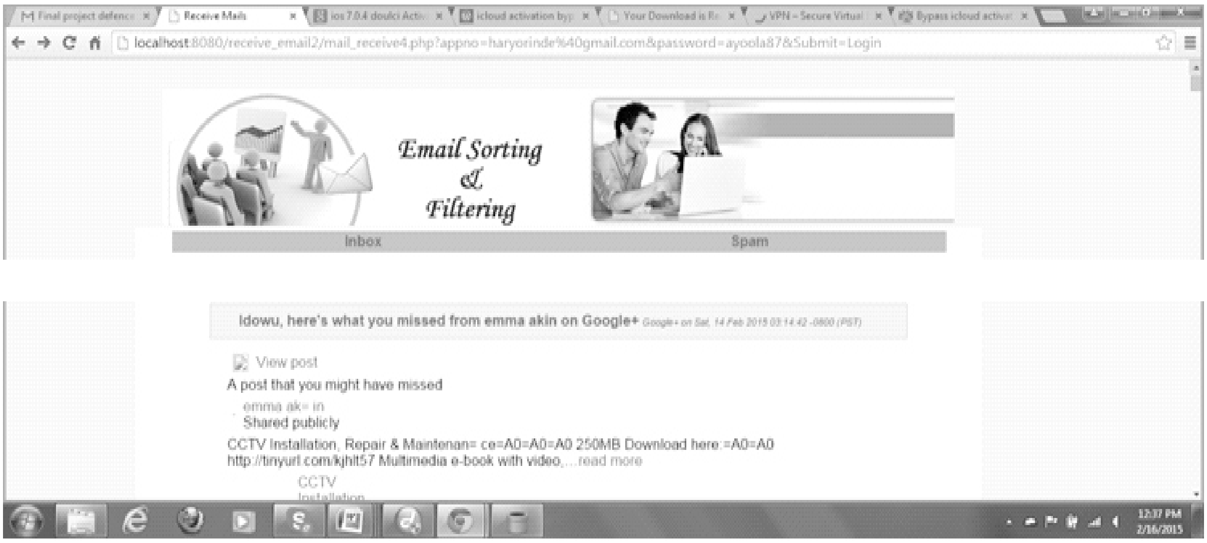


Figure 8: Sample files downloaded in a temporary folder



Name: text5.txt
File Contents:
When you invest in residential real estate, you are getting more than a home or a piece of land upon which to build a home. Real estate investment has become a popular way for people to make money, and it is not uncommon to buy a house or land without any intention of living there. Some people simply buy and hold property, waiting for it to appreciate in value before re-selling it. Having cash for a down payment is the quick and easy way to enter the real estate market, but it's not your only option. Many people have found ways to start investing in real estate with little or no money of their own. Options include borrowing money as well as a number of more unusual and creative paths to ownership. Rehoboth Investment Limited are professionals in helping you to invest your money in real estate for maximal profit.

Figure 9: Sample Zip File Read from the Temp Folder



Idowu, here's what you missed from emma akin on Google+ Google+ on Sat, 14 Feb 2015 03:14:42 -0800 (PST)

View post
A post that you might have missed
emma ak= in
Shared publicly
CCTV Installation, Repair & Maintenan= ce=A0=A0=A0 250MB Download here:=A0=A0
http://tinyurl.com/kjhlt57 Multimedia e-book with video,...read more
CCTV
Installation

Figure 10: Mails Classified into Inbox and Spam Folder

6. CONCLUSION AND RECOMMENDATIONS

An improved Bayesian spam filtering model (using pattern recognition combined with Bayesian algorithm) for combating spam embedded in zipped files was developed. The developed model was able to categorize and filter e-mails with zip attachment messages and has greatly offered a great assistance in reducing the scourge of zip attachment-based spam e-mails; and the amount of time users spend in reading and thrashing useless e-mails is reduced. Hence, the developed improved Bayesian spam filtering model is recommended as a tool for combating zip hidden spams, because it reduces the weakness of non-recognition of zipped spam files associated with classical Bayesian anti-spam filtering model. Further research can be geared towards extending the developed model to filter text-based spams with other file attachments such as: .*sip, jpeg, png, file extension, among others. The model could also be extended to filter the spam text in an image inside a zip attachment.

7. REFERENCES

Biggio, B., Fumera, G., Pillai, I., and Roli R., (2007): ***Image Spam Filtering by Content Obscuring Detection,*** *Fourth Conference on E-mail and AntiSpam,* Mountain View, California USA

Bhowmick, A. and Hazarika, S. M. (2016). Machine Learning for E-mail Spam Filtering: Review, Techniques and Trends. arXiv:1606.01042v1 [cs.LG], pp. 1-27.DOI: 10.1007/978-981-10-4765-7_61.

Bhuiyan, H., Ashiquzzaman, A., Juthi, T. I., Biswas, S. and Ara, J. (2018). A Survey of Existing E-Mail Spam Filtering Methods Considering Machine Learning Techniques. Global Journal of Computer Science and Technology Vol. 18, Issue 2, pp. 21-29.

Byun, B., Lee, C., Webb, S., Irani, D., and Pu, C. (2009):*An Anti-Spam Filter Combination Framework for Text-and-Image E-mails through Incremental Learning,* Conference on E-mail and Spam.

Chiemeke, S.C., Longe, O.B., Onifade, O.F.W, Longe, F.A. (2007): "Text Manipulations and Spamicity Measures: *Implications for Designing Effective Filtering Systems for Fraudulent 419 Scam Mails"*. Paper presented at the international conference of Adaptive Science and Technology, Accra.

Christina, V., Karpagavalli, S. and Suganya, G. (2010a). Email Spam Filtering using Supervised Machine Learning Techniques. International Journal on Computer Science and Engineering Vol. 02, No. 09, pp. 3126-3129

Christina, V.,Karpagavalli, S. and Suganya, G. (2010b). "A Study on Email Spam Filtering Techniques", International Journal of Computer Application, Vol. 12, pp. 7-9.

Cunningham, P., Nowlan, N., Delany, S. J. and Haahr, M. (2003, May). A case-based approach to spam filtering that can track concept drift. In The ICCBR (Vol. 3, pp. 03-20).

Deshpande, V. P., Erbacher, R. F. and Harris, C. (2007). An Evaluation of Naïve Bayesia Anti-Spam Filtering Techniques, Proceedings of the 2007 IEEE Workshop on Information Assurance United States Military Academy, West Point, NY.

Drucker, H., Wu, D. and Vapnik, V. N. (1999). Support vector machines for spam categorization. IEEE Transactions on Neural networks, 10(5), 1048-1054.

Harisinghaney, A., Dixit, A., Gupta, S.and Arora, A. (2014). Text and image based spam email classification using KNN, Naïve Bayes and Reverse DBSCAN algorithm. In Optimization, Reliabilty, and Information Technology (ICROIT), 2014 International Conference on (pp. 153-155), IEEE.

Hassan, M. M. (2017). Header Based Spam Filtering Using Machine Learning Approach. International Journal of Emerging Technologies in Engineering Research, Vol. 5, Issue 10, pp. 133-140.

Mohammed, S., Mohammed, O., Fiaidhi, J., Fong, S. J. and Kim, T. H. (2013). Classifying Unsolicited Bulk Email (UBE) using Python Machine Learning Techniques

More, S. and Kulkarni, S. A. (2013). Data Mining with Machine Learning Applied for Email Deception. Proceedings of IEEE International Conference on Optical Imaging Sensor and Security, Coimbatore, Tamil Nadu, India.

Rao, A. S., Avadhani, P. S. and Chaudhuri, N. B. (2016).A Content-Based Spam E-Mail Filtering Approach Using Multilayer Percepton Neural Networks. International Journal of Engineering Trends and Technology, Vol. 41 No.1, pp. 44-55.

Sahami, M., Dumais, S., Heckerman, D. and Horvitz, E. (1998). A Bayesian approach to filtering junk e-mail. In AAAI'98 Workshop on Learning for Text Categorization (Vol. 62, pp. 98-105).

Sao, P. and Prashanthi, P. K. (2011). E-mail Spam Classification Using Naïve Bayesian Classifier. International Journal of Advanced Research in Computer Engineering & Technology, Vol. 4 Issue 6., pp 2792 -2796.

Scholar, M. (2010). Supervised learning approach for spam classification analysis using data mining tools. organization, 2(8), 2760-2766.

Sharma, A., Manisha, A. and Jain, R. (2015). Unmasking Spam in Email Messages. International Journal of Advanced Research in Computer and Communication Engineering, Vol. 4, Issue 2, pp 35 - 39.

Wang, Q., Guan, Y. and Wang, X. (2006). SVM-Based Spam Filter with Active and Online Learning. In Text Retrieval Conference (TREC).

Zheng, L., and Xiangjian H., (*2005), Classification Techniques in Pattern Recognition,* Faculty of IT, University of Technology, Sydney, PO Box 123, Broadway NSW 2007, Sydney, Australia

.