**Available Online at www.ijcsmc.com**

**International Journal of Computer Science and Mobile Computing**

**A Monthly Journal of Computer Science and Information Technology**

**ISSN 2320–088X**

***IJCSMC, Vol. 2, Issue. 11, November 2013, pg.91 – 98***

**SURVEY ARTICLE**

# Survey of Spam Filtering Techniques and Tools, and MapReduce with SVM

**Amol G. Kakade[1], Prashant K. Kharat[2], Anil Kumar Gupta[3]**

[1]Department of Information Technology, Walchand College of Engineering, Sangli, India
[2]Department of Information Technology, Walchand College of Engineering, Sangli, India
[3]C-DAC, Pune, India
[1] agkakade@gmail.com; [2] prashant.kharat@walchandsangli.ac.in; [3] anilg@cdac.in

***Abstract***

***Spam is unsolicited, junk email with variety of shapes and forms. To filter spam, various techniques are used. Techniques like Naïve Bayesian Classifier, Support Vector Machine (SVM) etc. are often used. Also, a number of tools for spam filtering either paid or free are available. Amongst all techniques SVM is mostly used. SVM is computationally intensive and for training it can't work with large a dataset, these cons can be minimized by introducing MapReduce framework for SVM. MapReduce framework can work in parallel with input dataset file chunks to train SVM for time reduction. This paper aims at surveying of few such techniques and popular spam filtering tools with scope to introduce MapReduce with SVM.***

***Keywords - Spam filtering techniques; Naïve Bayes classifier; Support Vector Machine (SVM); Neural Networks (NN); K-Nearest Neighbor (KNN); spam filtering tools; MapReduce***

## I. INTRODUCTION

The email system has become the most ubiquitous modern day communication tool. Its popularity as well as wide-spread availability has created an opportunity for good business. Email is a quick and easy way to stay in touch with family, business, contacts, and strangers. It doesn't cost the price of a stamp and you don't have to wait for days to get a response or answer. Spam is nothing but the unwanted mail. Definition of spam is mail which is sent in bulk. Spam email, also known as junk email with numerous recipients by email. Clicking on links in spam email may send users to phishing web sites or sites that are hosting malware. Spam email may also include malware as scripts or other executable file attachments. The opposite of 'spam', email which one wants, is called 'ham'.

To separate such spam from important mails spam filtering is required. Techniques for such spam filtering are Naïve Bayesian classification, Support Vector Machine, K Nearest Neighbor, Neural Networks [2, 3]. These are content based filtering techniques. Also, a number of freeware and paid tools for spam filtering are available. Machine learning technique such as Support Vector Machines (SVM) can be applied in spam filtering. SVM training is a computationally intensive process so there is lot of scope to introduce the MapReduce platform for spam filter training [1]. MapReduce paradigm [5] works with Map and Reduce tasks and these tasks are independent. Sequential minimal optimization (SMO) is an algorithm for efficiently solving the optimization problem which arises during the training of support vector machines. Lot of work is done to make SVM parallel [4]. Common terms which are considered as threat over internet are spam, virus and spyware. These three terminologies have different impact and format. Difference in these is shown in Table 1.

**Table 1. Difference in virus, spam and spyware**

| Virus | Spam | Spyware |
|---|---|---|
| Virus is program. | Spam is unsolicited or junk email. It is having different format and shapes. | Spyware can be a software or a program which gathers information about internet user. |
| It can self replicate to cause more damage. Causes deletion, modification of files or other system resources. | It can't self replicate as not a program. | Transfers gathered information to third party in most cases unauthorized way. |

The rest of the paper is organized as follows. In Section 2 we briefly discuss some techniques of spam filtering. We present a short discussion on each technique. In Section 3 we describe in short some popular spam filtering tools along with their pros and cons. In Section 4 we describe in short need to introduce MapReduce in SVM training algorithm to make it worth for scalable dataset for training.

## II. SPAM FILTERING TECHNIQUES

In this section some mostly used spam filtering techniques are discussed.

### *A. Naive Bayesian Classifier*

It makes use of Naive Bayes classifier [7]. Basically it operates with 'Bag of Words' which is represented as an unordered collection of words. Naive Bayes classifiers [24] work by correlating the use of tokens (typically words), with spam and non-spam e-mails and then using Bayesian inference to calculate a probability that an email is or is not spam.

For calculating spamicity of total message, it lets us combine the probability of multiple independent events into one number. Each word has particular probabilities of occurring in spam email and in legitimate email. Overall process has following steps

- Train the filter.
- Calculate probability of words.
- Combine the word probability to classify mail.

At start emails are manually classified as spam or ham. After training probability of each word in spam and legitimate mail is calculated by following formula and this data is stored by spam filter in its database. Filter also maintains spam token database and non-spam token database which contain count of each word in email.

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

Where,
Pr(S) = overall probability that any given message is spam.
Pr(S/W) = probability that a message is a spam.
Pr (W/S ) = probability that the specified word appears in spam message.
Pr (W/H) = probability that specified word appears in ham message.
Overall probability of email is calculated as

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

Where,
Pr(S) = probability that any given message is spam.
Pr (H) = probability that any given message is ham.
Disadvantages of Naïve Bayes classifier technique are

- Words which occur in spam are misspelled.
- Images are inserted in emails for sensitive words by spammers and Bayesian Classifier can't analyze images.

### *B. Support Vector Machine*

Support vector machine [26] is one of the most recent techniques used in text classification. It is technique of pattern recognition and data analysis. In machine learning the training sample is a set of vectors of *n* attributes. We can then assume that we are in a hyperspace of *n* dimensions, and that the training sample is a set of points in the hyper-space. Let us consider the simple case of just two classes (as it is the case of spam problem).The classification using Support Vector Machine look for the hyper plane able to separate the points of the first class from those of the second class. Distance between the hyper plane and points of each class, is kept maximum for good separation.

Here in figure 1 Plane1 is good classifier and Plane2 doesn't classify. In case we can't find the hyper plane when the classes are not linearly separable [27, 28] the hyper space is extended to more dimensions. This insures the existence of hyper plane that separates the two classes. One interesting feature is that to find the appropriate plane, SVM method explore just the nearest points.

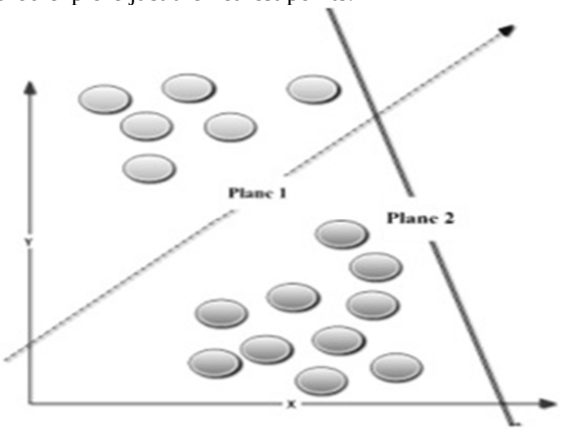


**Fig. 1. Hyper-plane that separate two classes**

### *C. K- Nearest Neighbor*

K Nearest Neighbor (KNN) [25] is non-parametric and lazy learning algorithm. In non-parametric technique, it does not make any assumptions on the underlying data distribution. KNN algorithm is also called as lazy learning algorithm. It means it does not use the training data points to do any generalization. In other words, there is no explicit training phase or it is very minimal. This means the training phase is pretty fast.

Lack of generalization means that KNN [24] keeps all the training data and training data is needed during the testing phase. This is in contrast to other techniques like SVM where you can discard all non-support vectors without any problem. KNN makes decision based on the entire training data set (in the best case a subset of them). Training phase is costly in terms of memory as we need to store all training data.

The principle of this technique is rather simple. Let us suppose that similarities among messages are measurable using a measure of distance among the characteristic vectors. To decide whether a message is legitimate or not, we look at the class of the messages that are closest to it. Generally, this technique does not use a separate phase of training and the comparison between the vectors is a real time process. This has a time complexity of *O(nl)* where *n* is the size of the characteristic vector and *l* the sample size. To adjust the risk of false classification, *t/k* rule is introduced. If at least *t* messages in *k* neighbors of the message *m* are unsolicited, then *m* is unsolicited email, otherwise, it is legitimate. We should note that the use of an indexing method in order to reduce the time of comparisons induces an update of the sample with a complexity *O(m)*, where *m* is the sample size. An alternative of this technique is known as memory based approach.

### *D. Neural Networks*

An Artificial Neural Network (ANN) is an information processing paradigm that is inspired by biological nervous systems, such as the brain, process information [18,24]. Base element for this paradigm is 'neurons'. ANN learns by example. An ANN is configured for a specific application, such as pattern recognition or data classification, through a learning process.

#### *1) Perceptron*

Output of Perceptron paradigm is a function f(x) = *Xw*+b.

Where, *w* is a vector of weights and *b is* a bias vector. A graphical representation of the Perceptron is shown in above figure. The training of the Perceptron is performed using an iterative method, where the weight and bias vectors are initialized then adjusted each iteration in such manner to ensure the classification of a new occurrence of the training sample. For instance let *x* be a vector that the Perceptron fails to classify, and *wi*, *bi*

the vector of weight and bias which corresponds to the $i^{th}$ iteration. The training continues until the Perceptron manages to classify correctly all the messages of the training sample. In this case, we say that the Perceptron converges.

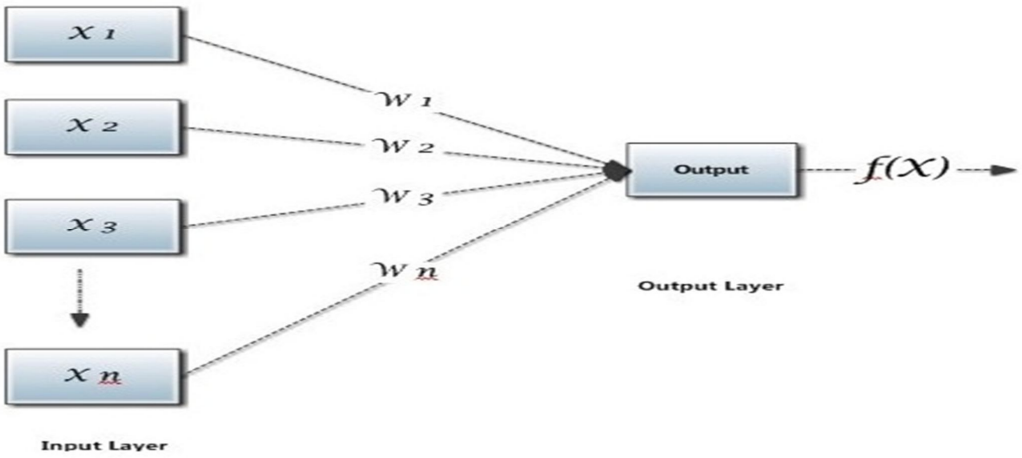


**Fig. 2. Perceptron**

The algorithm 1 summarizes the training of the Perceptron.

1: Initialize $w$ and $b$.
2: **while** $\exists x \in$ training sample such that $sign(wx + b) \neq c$ **do**
3: $w \leftarrow w + cx$ and $b \leftarrow b + c$
4: **end while**

**Algorithm 1. Training algorithm of Perceptron**

Here in algorithm $sign(wix+bn) \neq c$ , where $c$ is the sign corresponding to the real class of the message that has the characteristic vector $x$. The new vectors $wi+1$ and $bi+1$ are calculated as follow: $wi+1 = wi + cx$ *and* $bi+1 = bi + c$.

*2) Multi-layer neural network*

This network has three layers as shown in Figure 3 input layer, hidden layer(s) and output layer. A vector of values ($x_1 ... x_p$) is presented to the input layer. The input layer standardizes these values so that the range of each variable is -1 to 1. The input layer distributes the values to each of the neurons in the hidden layer.

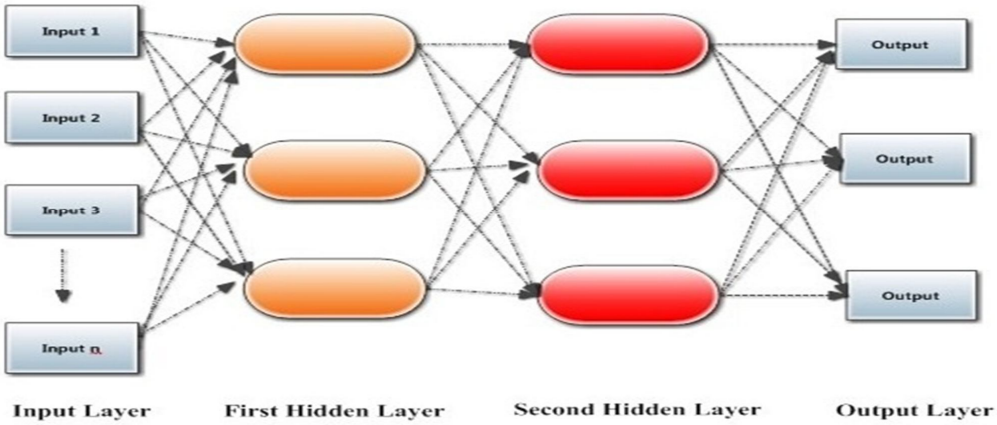


**Fig. 3. Multi-layer neural network**

In addition to the input variables, there is a constant input of 1.0, called the *bias* that is fed to each of the hidden layers; the bias is multiplied by a weight and added to the sum going into the neuron. At a neuron in the hidden layer, the value from each input neuron is multiplied by a weight ($w$), and the resulting weighted values are

added together producing a combined value. The outputs from the hidden layer are distributed to the output layer. At a neuron in the output layer, the value from each hidden layer neuron is multiplied by a weight ($w$), and the resulting weighted values are added together producing a combined value. The values are the outputs of the network. The function of each neuron is somewhat different from the simple Perceptron, although the training is also made in an iterative way as the simple Perceptron. The output function is:

$$o = \phi\left(\sum_{i=1}^{k} w_i x_i + b\right)$$

Where, φ is a nonlinear function such as tanh(x).
Spam filter tools which work on neural network are NAGS spam filter [13], Spam Bouncer [14] etc.

## III. SPAM FILTERING TOOLS

In programs used to filter spam, two main problems are false positive and false negative. In false positive, legitimate message mistakenly marked as spam. In false negative spam is mistakenly marked as legitimate message. Messages that are determined to be spam may be rejected by a server or client-side spam filter program and returned to the sender as bounce email. One problem with many filtering tools is that if they are configured stringently enough to be effective, there's a fairly high chance of getting false positives. The risk of accidentally blocking an important message has been enough to deter many companies from implementing any anti-spam measures at all.

Numbers of tools are available on internet for spam filtering. List of various spam filtering tools is available at Wikipedia [15,16].

### *A. POPFile*

POPFile [8] is a powerful and flexible email classification tool to filter spam efficiently and categorize good mail automatically. POPFile uses Bayesian statistical analysis to classify incoming messages.

- POPFile works between your email client and mail server.
- Main features are 'POPFile Magnets', automatic configuration of Outlook, bit heavy in terms of memory, remotely accessible web interface.
- 'Buckets' can be added for further in depth classification of emails.
- Average Bucket count used worldwide is 6 from POPFile statistics.
- Average accuracy of classification is 97.90%.
- Requires training.

Snapshot of results page is shown below.

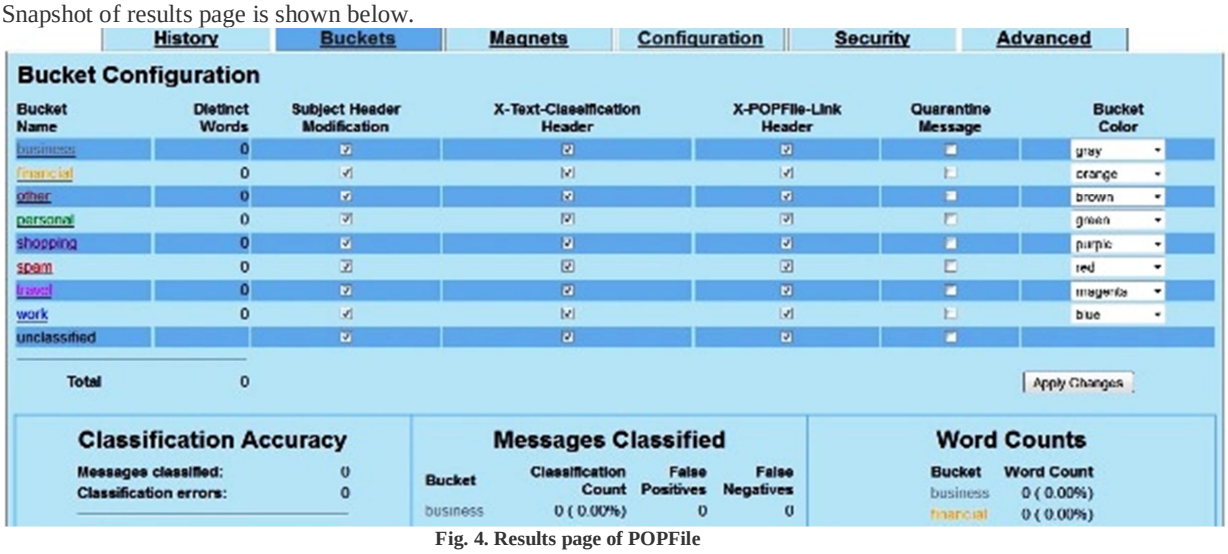


**Fig. 4. Results page of POPFile**

### *B. Spamfence*

Spamfence [9] is effective spam and virus filtering service. Its only real shortcoming is that Spamfence relies on forwarding and requires two different email accounts.

- Spamfence requires no configuration, maintenance or training.
- Can't fetch mail from a POP or IMAP account itself, requires forwarding of email so two two different email accounts are needed.
- Spam detection rate is little slower than Bayesian filter.
- Emails classified as clean, bulk, spam or dangerous.
- For above each category an address can be designated at which you want to receive emails.
- Statistics and graphs of each class can be obtained.

### *C. Spamihilator*

Spamihilator [10] is easy to use spam filtering tool. Its working is based on Naïve Bayesian classifier and it can work with any email client. Drawback of Spamihilator is that you cannot remotely administer or remotely teach. In Spamihilator

- The Learning Filter calculates a certain spam probability for every email.
- According to some keywords mails is marked as spam. To achieve this feature Spamihilator also contains Spam Word Filter that searches messages for known keywords.
- Suitable to work with Outlook, Mozilla Thunderbird, Eudora, Opera etc.
- Accuracy rate of classification is about 98%.
- Main aspect to save internet resource is it deletes spam without downloading it.

### *D. MailWasher*

MailWasher [11] Free is quite effective spam filtering solution. MailWasher Free only works with one POP account instead IMAP. MailWasher-Pro is paid version having lot of good fetures.

- Two lists of senders are maintained namely white and black list. These lists are very easy to maintain in MailWasher Free.
- MailWasher Free go on checking a POP account for new mail, and new spam.
- Has facility of preview mails on server itself where we can delete mails directly.
- You can build up sophisticated filters to automatically remove spam, chain letters and viruses.
- Filtering of mails performed against personal black and white lists.
- Can launch your default email program after processing messages automatically.
- MailWasher Free offers new mail notification in the Windows system tray and supports Windows 9x/ME/NT/2000/3/XP.

### *E. SPAMfighter*

SPAMfighter [12] has community of its users. If any user found spam/junk mail then he reports junk to other users in community.

- SPAMfighter Pro is simple and easy to use but can’t work with IMAP accounts.
- It is less accurate as compared to Bayesian filtering.
- It's impossible to find out why a message was filtered or what reporting it really reports.
- SPAMfighter Pro filters spam in Outlook, Windows Mail, Outlook Express and Mozilla Thunderbird.
- Blocking and unblocking mail is easy via a SPAMfighter Pro toolbar.
- Along with community support it maintains local black and white lists of bad and good senders just like MailWasher.
- Filter sensitivity of SPAMfighter Pro can be adjusted to your needs. It maintains filtering statistics.
- It supports most operating systems of Windows family like Windows 98, Windows XP, Windows Vista etc.

## IV. SVM AND MAPREDUCE

Training phase of SVM needs much more time than actual classification phase. SVM training time minimization is global minimization problem. During training entire dataset is used to get the final output, so there is lot of room for parallelization. Generally algorithm Sequential Minimal Optimization (SMO) is used for implementing SVM. Commonly used approach is to split the training data and use a number of SVMs to process the individual data chunks. Another problem with SMO is its scalability [1]. It can’t handle dataset of large size. We can wipe out these two problems by introducing MapReduce framework [5].

There are popular implementations of MapReduce like Mars [6], Phoenix [19], Hadoop [20] etc. Amongst above implementations Hadoop is mostly used in research field because of it 1 Terabyte sort [21, 22] achievement and support.

The application of MapReduce in spam filtering is also common within the industry, and is used widely by Microsoft, Yahoo, and Google etc. MapReduce programming paradigm allows for massive scalability across hundreds or thousands of nodes in a MapReduce cluster. With MapReduce [17] large input file for training

*© 2013, IJCSMC All Rights Reserved* 

SVM is spited into small size data chunks. Data chunks are then assigned to Map tasks. Figure 4 shows splitting of input file and working of MapReduce paradigm. Each Map task can almost independently run serial SMO on their respective training set. MapReduce gives output in form of {key, value} pair. Reduce task has {key, value} pairs generated by each Map task as input and combines result of all Map tasks to get final output. All map and reduce tasks run independently.

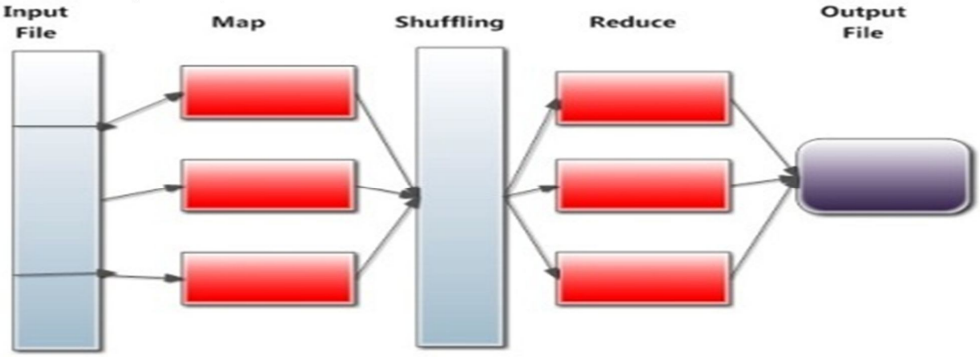


**Fig. 5. Working of MapReduce paradigm**

*ACKNOWLEDGMENT*

We express our sincere thanks to all the authors, whose papers in the area of spam filtering are published in various conference proceedings and journals.

REFERENCES

[1] Godwin Caruana, Maozhen Li, Yang Liu, "An ontology enhanced parallel SVM for scalable spam filter training," *Neurocomputing Elsevier*, vol. 108, pp. 45-57, 2013.
[2] E. Blanzieri, A. Bryl, "A survey of learning-based techniques of email spam filtering," *Artificial Intelligence Revolution*, vol. 29, pp.63–92, 2008.
[3] L. Zhang, J. Zhu, T. Yao, "An evaluation of statistical spam filtering techniques," ACM Transaction on Asian Language Information Process, vol. 3, pp.243–269, 2004.
[4] H. P. Graf, E. Cosatto, L. L. Bottou, I. Durdanovic, V.Vapnik, "Parallel support vector machines: the cascade SVM," *In Advancement of Neural Information Process System*, pp.521–528, 2004.
[5] J. Dean, S. Ghemawat, "MapReduce," *Communication of ACM*, vol. 51, pp.107, 2008.
[6] B.He, W.Fang, Q.Luo, N.K.Govindaraju, and T.Wang, "Mars: a MapReduce framework on graphics processors," *Proceedings of the 17th International Conference on Parallel Architectures and Compilation Techniques(PACT)*, pp.260–269, 2008.
[7] Naïve Bayes classifier. [Online]. Available:http://en.wikipedia.org/wiki/Naive_Bayes_classifier.
[8] POPFile - Automatic Email Classification- Trac. [Online]. Available: http://www.getpopfile.org/.
[9] Uber spamfence: spamfence powered by eleven.[Online]. Available:http://www.spamfence.net/about-spamfence.html.
[10] Spamihilator – Free Anti-Spam-Filter. [Online]. Available: http://www.spamihilator.com/.
[11] Mailwasher Free Spam Filter Software: The Reliable Free Spam Blocker. [Online]. Available:http://www.mailwasher.net/.
[12] SPAMfighter. [Online]. Available: http://www.spamfighter.com/.
[13] [Online]. Available:http://www.nags.org/index.html.
[14] The SpamBouncer: a Procmail-Based Spam Filter. [Online]. Available:http://www.spambouncer.org/.
[15] Email filtering-Wikipedia, [Online]. Available:http://en.wikipedia.org/wiki/Email_filtering.
[16] Top 12 Free Spam Filters for Windows, [Online]. Available:http://email.about.com/od/windowsspamfightingtools/tp/free_spam.htm.
[17] Apache Hadoop, http://www.hadoop.apache.org/.
[18] Christos Stergiou and Dimitrios Siganos, Neural Networks. [Online]. Available: http://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol4/cs11/report.html.
[19] K. Taura, K. Kaneda, T. Endo, A. Yonezawa, Phoenix: a parallel programming model for accommodating dynamically joining/leaving resources, ACM SIGPLAN 38 (2003) 216–229.
[20] Welcome to Apache Hadoop! [Online]. Available: /http://hadoop.apache.org/.
[21] A.Anand, Apache Hadoop Wins Terabyte Sort Benchmark. [Online], Available: http://developer.yahoo.com/blogs/hadoop/apache-hadoop-wins-terabyte-sort-benchmark-408.html.

[22] O. O'Malley, Terabyte Sort on Apache Hadoop, [Online], Available: /http:// sortbenchmark.org/YahooHadoop.pdf.
[23] Margaret H. Dunham, Data Mining - Introduction and advanced topic. Prentice Hall; 2003.Print.
[24] A Detailed Introduction to K-Nearest Neighbor (KNN) Algorithm, [Online], Available: http://saravananthirumuruganathan.wordpress.com/2010/05/17/a-detailed-introduction-to-k-nearest-neighbor-knn-algorithm/.
[25] G. Huang, K.Z. Mao, C. Siew, D. Huang, S. Member, "Fast modular network implementation for support vector machines," *IEEE Trans. Neural Networks*, vol.16, pp.1651–1663, 2005.
[26] Support vector machines: The linearly separable case, [Online],Available:http://nlp.stanford.edu/IR-book/html/htmledition/support-vector-machines-the-linearly-separable-case-1.html.
[27] Support Vector Machines (SVM) – StatSoft, [Online], Available: https://www.statsoft.com/textbook/support-vector-machines.

*© 2013, IJCSMC All Rights Reserved* 