Journal of Theoretical and Applied Information Technology
15 May 2012. Vol. 39 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

OPTIMIZED SPAM CLASSIFICATION APPROACH WITH NEGATIVE SELECTION ALGORITHM

[1]ISMAILA IDRIS, [2]ALI SELAMAT

[1]Student, Faculty of Computer Science and Information System, Unuversiti Teknologi Malaysia.
[2]Assoc. Prof, Faculty of Computer Science and Information System, Unuversiti Teknologi Malaysia

Email: [1]Ismi_idris@yahoo.co.uk, [2]selamat.ali@gmail.com

ABSTRACT

This paper initializes a two element concentration vector as a feature vector for classification and spam detection. Negative selection algorithm proposed by the immune system in solving problems in spam detection is used to distinguish spam from non-spam (self from non-self). Self concentration and non-self concentration are generated to form two element concentration vectors. In this approach to e-mail classification, the e-mail are considered as an optimization problem using genetic algorithm to minimize the cost function that was generated and then classification of these cost function shall aid in creating a classifier. This classifier will aid in the new formation of algorithm that comprises of both greater efficiency detector rate and also speedy detection of spam e-mail. The algorithm implementation of the research work shall come in stages were spam and non-spam are detected in all phases for an efficient classifier.

Keywords: *Negative Selection Algorithm, Neural Network, Support Vector Machine, Model, Self, Non-self*

1. INTRODUCTION

There are several measure put in place by many companies in the area of creating anti spam software based on signatures and have a very efficient performance in detecting spam fast. Though, new variation of spam and unknown spam are very difficult to detect by this software. The traditional way of detecting spam based on signature is no more efficient for today systems. However discussions on approaches that help present security concern at the implementation stage of system development was introduced in [1] Several classification approach was introduced in[2-6]. Recent years, Researchers are interested in the field of immune system in achieving computer security. The function of computer security systems are meant to recognize and discard spam. Data mining, machine learning and the signature base techniques are proposed for spam detection [7] The signature base spam detection technique is not too reliable in detecting new spam since the number of spam grows concurrently in such a way that the signature based spam detection technique cannot meet up with its security challenges, putting in mind the increase signature database or the time it will take before matching takes place in signatures. The data mining techniques keep in memory specific bytes sequence obtain in the file content and also monitor the behavior of suspicious program. Our proposed approach is been implemented in other to detect formally unknown spam. There is a relationship between any detectors selected to one feature dimension leading to large dimensionality of feature. This will result in the reduction of high precision rate; urge cost of processing which result in to low precision rate. There are different classification techniques proposed with Artificial Immune System which includes Naïve Bayes, Artificial Neural Network (ANN), Support Vector Machine (SVM) and other hybrid approaches. [8-12]

In our approach, we shall be making use of two element concentration vector as the feature vector for spam detection by using a concentration based feature construction approach inspired by immune system. The spam and non-spam mail that is from the self gene library and the non-self gene library are generated for feature construction. The self gene library and the non-self gene library is used to construct the self concentration and the non-self concentration and are used to form two element concentration vector to classify e-mail. This is considered as an optimization issue with the main objective of optimizing the formulated cost

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

function. Genetic algorithm is use for the optimization process. An optimal self concentration and non-self concentration is obtained, the one whose cost function is associated to the classifier is minimum.

In the remaining part of the paper, related work and improved approach are discussed in section 2. Detector library generation and proposed architecture is presented in 3. Generation of gene library and feature selection is in section 4. Optimization approach is presented in 5. Neural network and support vector machine is discussed in section 6. Experimental result is reported in section 7. Finally, conclusion is done in section 8.

2. RELATED WORK

Naturally, people turn to brain and neural system for breakthrough in computing problem solving methods. Algorithm and methods are developed through inspired human system with constant upgrade of knowledge by researchers. Over the past years, rapid expansion of computer network system as change the world. It is essential for an effective computer security system because attacks and criminal intend are increasingly popular in computer network. [13]. There are several measure put in place by many companies in the area of creating anti spam software based on signatures and have a very efficient performance in detecting spam fast. Programs suffering from abnormality that can lead to unpredictable program behavior in [14]. Though, new variation of spam and unknown spam are very difficult to detect by this software. [15].The signature based techniques for spam detection is the most widely used method. It makes use of binary data mining to detect data when given a large number of data and then use this data to detect data that looks similar in future detection [16].

Though there are limitations in detecting spam using the traditional technique as it can only detect a small number of generics or extremely broad signatures. Some clustering algorithm was also introduced to improve intrusion detection. It finds it difficult in detecting new spam threats despite several filtering techniques proposed in [17-24]. The suspicious behavior technique provides protection from spam that are yet to exist in spam dictionaries which is not like signature based technique that is meant to detect existing spam. [23] proposed a neural network system meant for automated e-mail classification. He also presented an email classification NN-based system used for automated e-mail categorization problem. This system is referred to as LINGER. It is an architecture meant for all kind of text categorization. Linger is adaptable, flexible and most of its operation are configured. It recorded that neural network are used successfully in automated e-mail filing and filtering spam mail but it as slow rate of performance and inconsistencies in result with several classification. An anti-spam filtering technique was presented by [25]; His techniques are centered on artificial neural network (ANN) and Bayesian Networks. Algorithm that was created by levent is meant for specific user and they use the characteristics of the incoming e-mail to also make adjustment on themselves, therefore not able to make preparation in detecting an unknown spam. [26] used from one user neural network techniques on a corpus of e-mail messages in his research. Descriptive characteristics of words are the feature set used to determine spam massages, this messages are also similar to messages that a reader will use in identifying spam.

The experimental work used a corpus of 1654 e-mails which was over a period of some months received by an author. He states that the neural network like Naïve Bayes only needs few features to get result. This prompt our knowledge on optimization using genetic algorithm were we will optimize the important feature to get good result. Neural network technique for classification of spam was also presented by [27]. Attributes of the techniques comprises of the characteristics of the patterns that most network invaders deploy instead of making use of the context of keywords in the message. The dataset that was used in this experiment is corpus of 2788 non-spam and 1812 spam emails that was put together for several months. The result that was acquired from this experiment actually shows that ANN is good but is not the best as it is not suitable to be used alone as tool for filtering spam. This informed the inconsistencies in using ANN without aiding it with other algorithm. In knowing detection success, suspicious behavior

Journal of Theoretical and Applied Information Technology
15 May 2012. Vol. 39 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

method is one way of achieving it, which will as well depends on the observable element from an agent externally. Due to the efficiency in executing malicious intent, the proposed method went through criticism. [28, 29] proposed the most inspired spam detection technique whose framework comprises of three learning algorithms; The first frame work was the rule based learner that generate Booleans rule based on feature attributes. The second frame work was the probabilistic technique creating a probability of a class been giving some features and finally, is a multiple classifier system that put together results from other classifier to create a prediction. This method includes strings and byte sequence that are extracted from malicious executable on the dataset as different type of features. We actually relate the bytes sequence method with our work and excellent result was achieved with high accuracy. [8], Malicious executable were detected by the use of data mining and n-gram analysis, sequences of bytes was extracted from the executable, and then is been transformed in to n-grams which are then treated as features. This help in making a final agreement of the need of extracting the most efficient feature from set of executables in other to get good result. An error detection methodology to enable fault detection inspired on recent immune theory was proposed in [30]. The fault detection problem is a challenging problem due to processes increasing complexity and agility necessary to avoid malfunction or accidents. The key challenge is determining the difference between normal and potential harmful activities.

3. DETECTOR LIBRARY GENERATED AND PROPOSED ARCHITECTURE.

In generating self detector library and non-self detector library, we divide the detector into two set according to its tendency values and calculate the detector importance with the important once been retained as illustrated in the diagram below

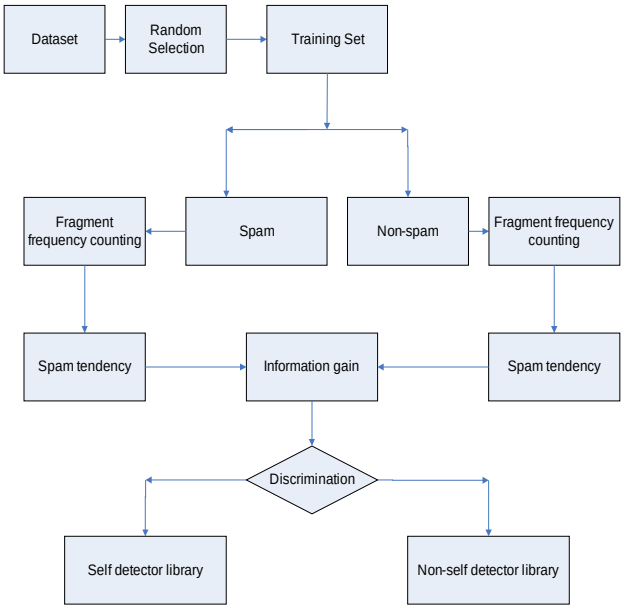

Fig 1. Detector library generating process

From the above generation of the self and non-self detector library which is composed of non-spam file and spam file respectively. Wang et al [14] proposed a fixed length fragment for detector in a library. Therefore, a fixed length (L-bits) fragment of binary data is considered to contain appropriate information of functional behaviors which represent the detector in giving the difference between a spam or non-spam. The length (L) is set not to be too short in other to be able to differentiate between self and non-self or not too long to make spam specific data that are hidden in the binary data of the files. Let's assume L as 32 bits, a sliding window is used to count the document frequency of a detector in spam and non-spam programs. The overlap of the sliding window is assumed to be L/2 bits. The difference in document frequency in both spam and non-spam program can as well trigger the tendency in the spam and non-spam program from been a spam and non-spam file.

4. GENERATION OF GENE LIBRARY.

From our previous work, [31], a self and non-self memory was generated. This two library is known as the self gene library and the non-self gene library. Both the self and non-self gene library is composed of fragment (words) with at most representation of non-spam e-mail and spam e-mail respectively. Fragments that are found in non-spam e-mail and rarely found in spam e-mail

Journal of Theoretical and Applied Information Technology
15 May 2012. Vol. 39 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

is a very good representation of non-spam e-mail. To reflect a fragment tendency, it can be generated by the difference of its frequency in non-spam e-mail minus that of the spam e-mail. Fragments are sorted out in order of their differences after each of the frequency for each fragment as been calculated.

The generation of gene libraries are describe in the algorithm below.

Algorithm 1: Construction of Gene Library

DEFINITIONS:
sm is a self detector library (spam detector)
nsm is a non-self detector library (non-spam detector)
x is the fragment (word) which is the sample of the training set (f_{sm} and f_{nsm})
T is the Tendency
sm(0)=0, nsm(0)=0;
INPUT:
T /* Tendency
b /* b is the gene library of x;
a /* a is the gene library of y;
OUTPUT:
Finding the tendency of fragment x in both sm and nsm
BEGIN
Input x;
Input sm(1),nsm(1) /*sm and nsm is its frequency appearing in both self detector and non-self detector;
For i=1 to x
sm(i) = sm(i) + sm (1- i);
Next;
For i =1 to x
nsm (i) = nsm (i) +nsm (i - 1);
If $f_{affinity}$ **>= T**
$f_{affinity}$ (sm) = max;
$f_{affinity}$ (smn) = max;
end if
if x = =.T.
(b . sm) >=T;
else
(a . nsm) >=T;
end for
For each fragment x in the sample of training set do
f(T) = f_{sm}–f_{nsm}
else
if f(T) < 0 then
x + sm
else
x + smn
end if
end if
end for

Parameter (P_{sm} and P_{nsm}) are to be adjusted. We remove both P_{sm} and P_{nsm}% in front and rear of the queue to form self and non-self gene library.

From the algorithm above, the tendency is acquired by the difference of its frequency in non-spam e-mail minus that in spam e-mail. The fragments are also sorted out accordingly in order of their difference after calculating the difference between each fragment frequency. For example, the two different fragment that are obtained from both front and rear of a queue with some population can be use to generate the self gene library and the non-self gene library.

4.1 *Generating Feature Vector*

Frequency of detector in both spam and non-spam are counted using sliding window. In other to construct feature vector from the sliding window, we assume L as 32 bits, a jumping window is moved to obtain many fixed length L with S as the segment bits in the program. The overlap sliding window L/2 is used to obtain the self and non-self local concentration. The local concentration of segment x in every window is defined as

$$XC_x = \frac{XN_x \cdot L}{S} \quad (1)$$

$$YC_x = \frac{YN_x \cdot L}{S} \quad (2)$$

Where XC_x denote self local concentration and YC_x represent non-self local concentration. Also, XN_x is the number of detector appearing in non-self detector library and the detecting segment of the file while YN_X is the number of detector appearing in self detector library and the detecting segment of the file.
The above equation illustrates how the self local concentration and the non-self local concentration are generated in each window. The self and non-self local concentration of this program are combine together to create a feature vector, $[(XC_x, YC_x), (XC_2, YC_2)\ldots\ldots, (XC_n, YC_n)]$. There should be consistency in the dimensionality by the use of the feature vector as an input to

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

classifier for spam detection. Rear dimensionality is discarded due to application of truncated operation. N × S bits of each of the program is applied. N is the number of segment covered by each jumping window and S is the segment bit in the program. Figure 3 express the feature concentration process.

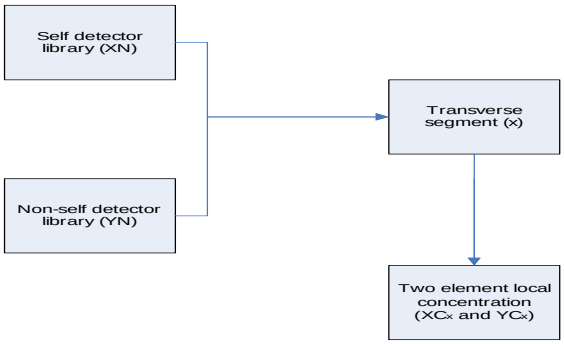

Figure 2. Feature concentration generation

Algorithm 2. For feature concentration generation

DEFINITIONS:
N= Number of segment covered by each jumping window
S= bits segment in the window
x= local concentration of segment
L= sliding window fixed length
L/2 = overlapping sliding window
INPUT:
XN /* is the number of non-self detector of a;
YN /* is the number of self detector of b;
x /* is the segment
OUTPUT:
Two element concentration to generate a feature vector. (XC_x *and* YC_x)
BEGIN:
Input a;
Input XN(1), YN(0); /* self and non-self detector
For i= 1 to a
XN(i) = XN(i) + XN (1-i);
Next;
For i= 1 to b
YN(i) = YN(i) + YN(1-i);
Traverse the segment x using L and L/2;
Truncate N × S bits of the file and discard dimensionality of the file;
$XC_x = XN_x$. L / S /* self local concentration
$YC_x = YN_X$. L / S /* Non-self local concentration
End for
Connect the self and non-self elements local concentration to generate feature vector

Figure 3 below shows the generation of feature vector that is to be used as input for classifier. This is generated from the self and non-self local concentration that was generated from self detector library and non-self detector library.

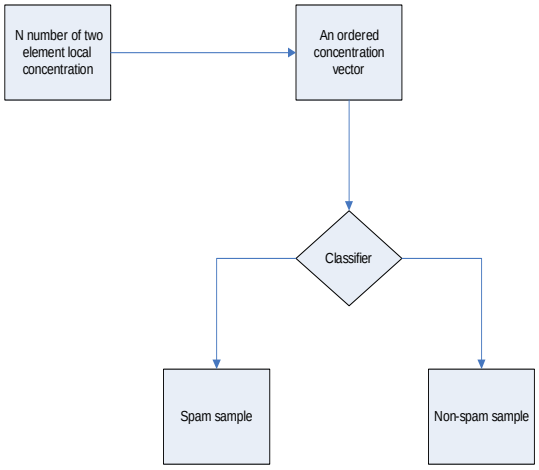

Figure 3. Feature vector generated

5. THE OPTIMIZATION APPROACH

The feature vector that was defined serves as the inputs of the classifier while the binary value serves as the output of the classifier. Both self and non-self concentration that forms the two element concentration vector is considered as an optimization problem. The optimal vector P* = {$P*_S$, $P*_N$, $P*_1$, $P*_2$,…….$P*_M$} which composed of the detector library determinants m and the parameters $P*_1$, $P*_2$,……$P*_M$ which is linked to a certain classifier. When the detector library classifier m is set to a different value, a different detector library will be realized. A unique feature vector can be constructed with a different detector library determinant, for characterization of file in which self and non-self local concentration representing their similarity to non-spam and spam respectively are different. Different classifier as different parameters, so therefore {$P*_1$, $P*_2$,……$P*_M$} are classifier related parameters that can influence the performance of the classifier based on the parameter value. For example, neural network parameter will include the structure of the network, the number of layer, the number of nodes present within a layer and each connection weight between two nodes while parameters that are related to Support Vector

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Machine (SVM) determine the position of optimal hyper plane in the feature space which includes cost parameter C and kernel parameters.

$P*_S$ and $P*_N$ and parameter $P*_1$, $P*_2$,…….$P*_M$ is the optimal vector, whose cost function is associated to a classifier whose classification is minimum. The cost function for a minimum classification CF(P) is given as

CF(P) = Err(P)
(3)

Where Err(P) is the classification error on the training set.
The input vector P is made up of two parts, the detector library determinant m, $P*_S$ and $P*_N$; and the parameter $P*_1$, $P*_2$, …….$P*_M$ which is limited to certain classifier. The vector P is the objective of the optimization problem whose performance is measured by CF(P).
P* which is the optimization concentration is defined as

$P^* = \{P*_S, P*_N, P*_1 , P*_2 ,----------P*_M\}$
(4)

Therefore, CF(P*) = min CF(P)
$\{P*_S, P*_N, P*_1 , P*_2 , -----P*_M\}$ (5)

CF(P) = Err(P*)
(6)

The optimization approach is used to optimize the input vector in the objective function such as the Genetic Algorithm (GA), Particle Swarm Optimization (PSO) and so on can be employed in solving optimization problem.
The figure below shows an optimization process with the use of genetic algorithm to design the local concentration feature of the corresponding classifier.

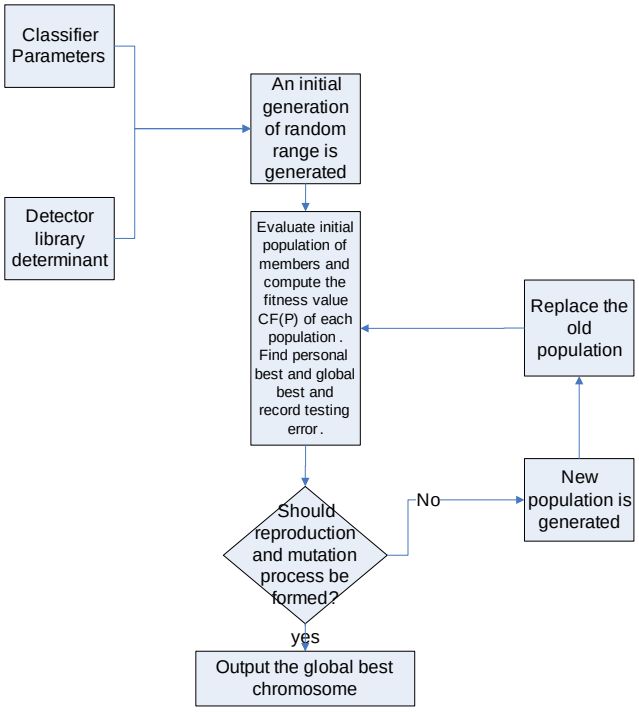

Fig 4. Genetic algorithm classification process

6. NEURAL NETWORK AND SUPPORT VECTOR MACHINE (SVM) CLASSIFICATION.

The sample dataset are characterized by self and non-self feature vector from which feature vector are made using neural network. It is an adaptive system which changes the structure base on information that goes through the network in the process of learning as we are trying to simulate the biological function ability of neural network. The neural network topology uses the radial basis function as the activation function. It comprises of three layer: an input layer, a hidden layer with a non-linear activation function and a linear output layer. The sigmoid function represents the output layer of linear combination of hidden layer values, representing an inner probability that is made up of one node which serves as the label of the detected file. During classification with neural network, the number of nodes of input layer is equal to the size of concentration vector, therefore, the concentration vector as a single component. The number of nodes of input layer is tested. There is only one node in the output layer. When 1 is output, it indicates a spam e-mail and when 0 is output it indicate a non-spam e-mail.

Journal of Theoretical and Applied Information Technology
15 May 2012. Vol. 39 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

The transfer function for hidden is the 'tansig' function while that of hidden is 'purelin' and the transfer function for output layer are is 'trainlm'. The performance function is MSE. The network is trained for a maximum of 200 epoch to 0.01 of error goal.

7. EXPERIMENT AND RESULTS

The dataset used to test the proposed technique is from the center of machine learning and intelligent system. The corpus is made up of 4601 instances with spam rate of 39.4%. The corpus is divided into partitions with approximate number of instances and spam rate. The spam dataset after division has 1813 instances while the non-spam dataset has 2788 instances.

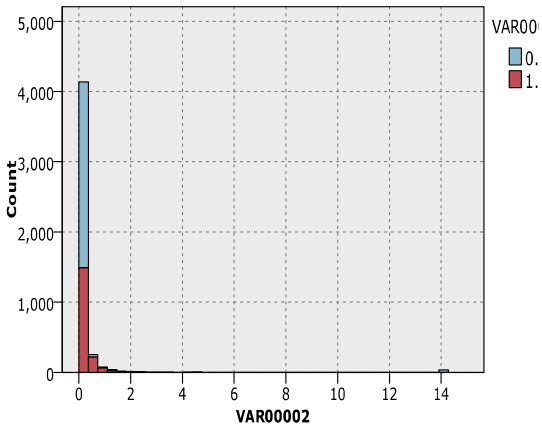

Fig. 5 Dataset Analysis

This is as represented in fig. 4 above analyzing the spam and non-spam ratio. The red indicates spam (1) while the green represent non-spam (0). A performance index was used for neural network and SVM to verify the effectiveness of the proposed approach. Clementine and MATLAB of version R2009a software package are implemented for both SVM and neural network.
The selection of P_s and P_N and also the parameter for neural network and support vector machine that is to be used in this experiment is considered as a constant change optimization process carried out by genetic algorithm. The cost function in equation (6) is the objective function for optimization and the fitness value of each population is the classification error measured by 10 fold cross validation on the training set. If the classification error is low, we have a better fitness evaluation but if the classification error is high, the fitness evaluation will not be good. Each partition of the corpus is 10% the original corpus. In each partition, we use 70% as the training data and 30% as the testing data using neural network and SVM as the classification algorithm. The performance of 10 fold cross validation shows 'self concentration' with P_s =30% and 'non-self concentration' with P_N =20% performs best on the corpus.

Table 1. Performance of neural network and SVM without optimization

Classifier	Acc (%)	Pre (%)	Rec (%)	MR (%)
NN	95.01	95.34	92.95	3.01
SVM	96.30	97.01	93.24	1.12

Self concentration and non-self concentration which corresponds to self gene library and non-self gene library with different classification are trained and tested using Neural network and SVM aiming to find the concentration with the best performance. The result of both P_s and P_N is illustrated in table 1 and fig. 5 below.

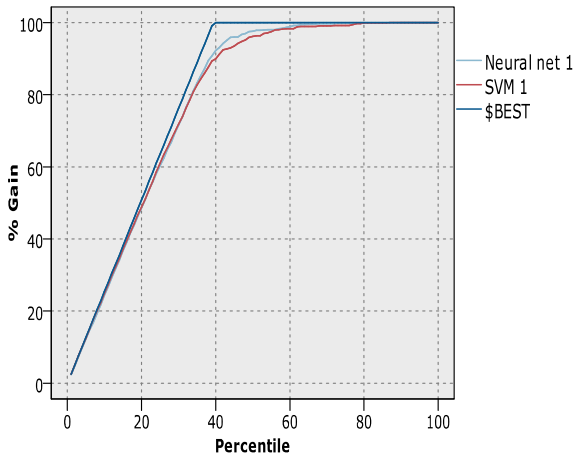

Fig.6 Training result for both Neural Network and SVM

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

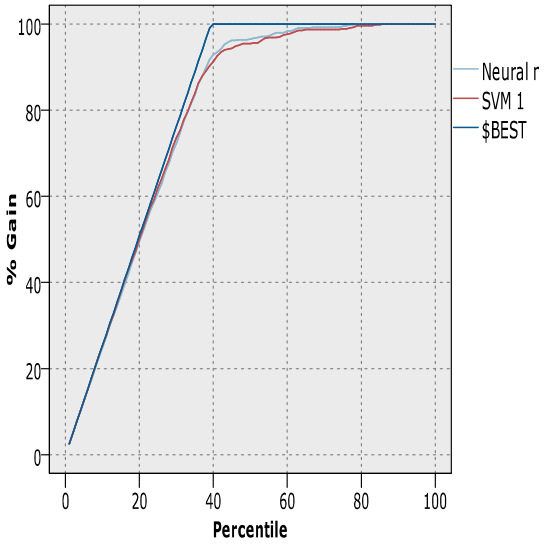

Fig.7 Testing result for both Neural network and SVM

Fig. 5 and Fig. 6 above illustrated both training and testing results of the proposed model. Classification using SVM for training is at its best at 96.30% of accuracy while neural network is 95.01% accuracy. The testing accuracy for the SVM is at 96.12% with false positive rate of 0.18% while neural network is at 93.98% with false positive rate of 1.03% respectively.

Also, experiment was done with optimization using genetic algorithm before classification process. From equation (6) which is our optimization equation, the value of P_s and P_N are both optimized in the real number interval [0, 0.5]. P_1, P_2 …….P_M are classifier related. Parameters that are required for neural network are the number of nodes of the hidden layer which are optimized in the integer number interval [3, 15]. Parameter C in the SVM is optimized in the real number integer [1, 200]. The maximum epoch or iteration for the study is set at 200 and the number of population is set at 20. The difference between the experiment and the previous one is that optimization was done on the values. The test was conducted with 10 different partitions. In the process of optimization, due to the randomness of genetic algorithm, the performance and result for P_s and P_N varies while the best classified partition result is presented for this paper. The result for the optimized technique is as presented in Table 2.

Table 2. Performance of neural network and SVM with optimization

Classifier	Acc (%)	Pre (%)	Rec (%)	MR (%)
NN	97.26	96.85	96.84	1.58
SVM	97.89	97.58	97.78	1.48

8. CONCLUSION

A frame work was established for new approach in spam detection. Measurement of classification of e-mail was made possible by the introduction of cost function resulting in to optimal concentration because the cost function was minimized by genetic algorithm. Neural network and SVM was employed to show the effectiveness of the proposed method with respect to optimization with genetic algorithm and the non optimized approach. Comparison of the two approach performance conducted in different classifier and the best result of all partitioned experiment was acquired. The experimental results confirm that the performance of the optimized approach is better than the non optimized approach.

REFERENCES

[1] R. Matulevičius, *et al.*, "An Approach to Assess and Compare Quality of Security Models," *Computer Science and Information Systems. ComSIS* vol. Vol. 8, No. 2, , 2011.

[2] C.-H. Wu, *et al.*, "Patent classification system using a new hybrid genetic algorithm support vector machine," *Applied Soft Computing*, vol. 10, pp. 1164-1177, 2010.

[3] P. Lingras and C. Butz, "Rough set based 1-v-1 and 1-v-r approaches to support vector machine multi-classification," *Information Sciences*, vol. 177, pp. 3782-3798, 2007.

[4] V. Mitra, *et al.*, "Text classification: A least square support vector machine approach," *Applied Soft Computing*, vol. 7, pp. 908-914, 2007.

[5] K. Igawa and H. Ohashi, "A negative selection algorithm for classification and reduction of the noise effect," *Applied Soft Computing*, vol. 9, pp. 431-438, 2009.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[6] D.-H. Shih, *et al.*, "Classification methods in the detection of new malicious emails," *Information Sciences*, vol. 172, pp. 241-261, 2005.
[7] M. Christodorescu, *et al.*, "Mining Specifications of Malicious Behavior," *Proceedings of the 6th joint meeting of the european software engineering . New york , USA*, pp. 5-14, 2007.
[8] D. M. Cai, *et al.*, "Comparison of feature selection and classification algorithms in identifying malicious executables," *Computational Statistics & Data Analysis*, vol. 51, pp. 3156-3172, 2007.
[9] I. Santos, *et al.*, "Opcode sequences as representation of executables for data-mining-based unknown malware detection," *Information Sciences*, 2011.
[10] E.-S. M. El-Alfy and R. E. Abdel-Aal, "Using GMDH-based networks for improved spam detection and email feature analysis," *Applied Soft Computing*, vol. 11, pp. 477-488, 2011.
[11] B. Yu and D.-h. Zhu, "Combining neural networks and semantic feature space for email classification," *Knowledge-Based Systems*, vol. 22, pp. 376-381, 2009.
[12] B. Jin, *et al.*, "Support vector machines with genetic fuzzy feature transformation for biomedical data classification," *Information Sciences*, vol. 177, pp. 476-489, 2007.
[13] D. Fisch, *et al.*, "On the versatility of radial basis function neural networks: A case study in the field of intrusion detection," *Information Sciences*, vol. 180, pp. 2421-2439, 2010.
[14] João Lourenço, *et al.*, "Detecting Concurrency Anomalies in Transactional Memory Programs," *Computer Science and Information System. ComSIS*, vol. Vol. 8, No. 2, , May,2011.
[15] Z. Yuanchun and T. Ying, "A Local-Concentration-Based Feature Extraction Approach for Spam Filtering," *Information Forensics and Security, IEEE Transactions on*, vol. 6, pp. 486-497, 2011.
[16] O. Henchiri and N. Japkowicz, "A Feature Selection and Evaluation Scheme for Computer Virus Detection," in *Data Mining, 2006. ICDM '06. Sixth International Conference on*, 2006, pp. 891-895.
[17] C. Catal and B. Diri, "Investigating the effect of dataset size, metrics sets, and feature selection techniques on software fault prediction problem," *Information Sciences*, vol. 179, pp. 1040-1058, 2009.
[18] L. Gong, *et al.*, "Text stream clustering algorithm based on adaptive feature selection," *Expert Systems with Applications*, vol. 38, pp. 1393-1399, 2011.
[19] B. Biggio, *et al.*, "A survey and experimental evaluation of image spam filtering techniques," *Pattern Recognition Letters*, vol. 32, pp. 1436-1446, 2011.
[20] T. S. Guzella and W. M. Caminhas, "A review of machine learning approaches to Spam filtering," *Expert Systems with Applications*, vol. 36, pp. 10206-10222, 2009.
[21] C. Laorden, *et al.*, "Word sense disambiguation for spam filtering," *Electronic Commerce Research and Applications*.
[22] X. Yue, *et al.*, "Immune-inspired incremental feature selection technology to data streams," *Applied Soft Computing*, vol. 8, pp. 1041-1049, 2008.
[23] Y. Yue and S. Elfayoumy, "Anti-Spam Filtering Using Neural Networks and Baysian Classifiers," in *Computational Intelligence in Robotics and Automation, 2007. CIRA 2007. International Symposium on*, 2007, pp. 272-278.
[24] Mrutyunjaya Panda, *et al.*, "SOME CLUSTERING ALGORITHMS TO ENHANCE THE PERFORMANCE OF THE NETWORK INTRUSION Detection System," *Journal of Theoretical and Applied Information Technology* vol. 4, p. 8, 2008.
[25] L. Özgür, *et al.*, "Adaptive anti-spam filtering for agglutinative languages: a special case for Turkish," *Pattern Recognition Letters*, vol. 25, pp. 1819-1831, 2004.
[26] C. Dragos, "Stability issues in a Biological Model of Self and Non-self Immune Regulation," in *Electrical and Computer Engineering, 2006. CCECE '06. Canadian Conference on*, 2006, pp. 1739-1742.
[27] A. H. Mohammad and R. A. Zitar, "Application of genetic optimized artificial immune system and neural networks in spam detection," *Applied Soft Computing*, vol. 11, pp. 3827-3845, 2011.
[28] G. Jacob, *et al.*, "Behavioral detection of malware: from a survey towards an established taxonomy," *Journal in Computer Virology*, vol. 4, pp. 251-266, 2008.
[29] M. G. Schultz, *et al.*, "Data mining methods for detection of new malicious executables," in *Security and Privacy, 2001. S&P 2001. Proceedings. 2001 IEEE Symposium on*, 2001, pp. 38-49.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[30] C. A. Laurentys, *et al.*, "A novel Artificial Immune System for fault behavior detection," *Expert Systems with Applications*, vol. 38, pp. 6957-6966, 2011.

[31] I. Idris and A. Selamat, "A Spam Detection Model Based on Negative Selection Algorithm.," *IJMIA: International Journal on Data Mining and Intelligent Information Technology Applications* (Accepted October 27, 2011) 2011.