EMAIL SPAM CLASSIFICATION USING HYBRID APPROACH OF RBF NEURAL NETWORK AND PARTICLE SWARM OPTIMIZATION

Mohammed Awad[1]and Monir Foqaha[2]

[1]Department of Computer Systems Engineering, Arab American University-Jenin, Palestine

[2]Department of Computer Science, Arab American University-Jenin, Palestine

ABSTRACT

Email is one of the most popular communication media in the current century; it has become an effective and fast method to share and information exchangeall over the world. In recent years, emails users are facing problem which is spam emails. Spam emails are unsolicited, bulk emails are sent by spammers. It consumes storage of mail servers, waste of time and consumes network bandwidth.Many methods used for spam filtering to classify email messages into two groups spam and non-spam. In general, one of the most powerful tools used for data classification is Artificial Neural Networks (ANNs); it has the capability of dealing a huge amount of data with high dimensionality in better accuracy. One important type of ANNs is the Radial Basis Function Neural Networks (RBFNN) that will be used in this work to classify spam message. In this paper, we present a new approach of spam filtering technique which combinesRBFNN and Particles Swarm Optimization (PSO) algorithm (HC-RBFPSO). The proposed approach uses PSO algorithm to optimize the RBFNN parameters, depending on the evolutionary heuristic search process of PSO. PSO use to optimize the best position of the RBFNN centers c. The Radii r optimize using K-Nearest Neighbors algorithmand the weights w optimize using Singular Value Decomposition algorithm within each iterative process of PSO depending the fitness (error) function. The experiments are conducted on spam dataset namely SPAMBASE downloaded from UCI Machine Learning Repository. The experimental results show that our approach is performed in accuracy compared with other approaches that use the same dataset.

KEYWORDS

Email Spam, Classification, Radial Basis Function Neural Networks, Particles Swarm Optimization.

1. INTRODUCTION

Email in the twenty-first century hasbecome one of the most important methods for communication among people; this is due to its free availability, fast and free or lower cost. The major problem has become in email messages is anunwanted message (spam). The person that sends the spam messagesis called spammer who collects email addresses from websites, chat rooms, and viruses. The spam traffic volume is so large,which negatively affects email servers storage space, networks bandwidth, processing power and user time [1].According to the statistics conducted between 2010-2014, 18% of the traffic is spam [2]. Another statistic according to [3] they found 13 billion of unwanted commercial email nearly 50% of all email sent. On the other hand, Kaspersky security bulletin in 2013 appeared that the counted spam during 2013 approximately equal 70% of the total email traffic [4]. Also, according to University of California – Irvine statistical, the total of blocked messages reached 690,849,027 on November 30, 2013 [5]. There are many classification techniques used to classify data into categories, including probabilistic, decision tree, artificial immune system [6], support vector machine (SVM) [7],

DOI: 10.5121/ijnsa.2016.8402

artificial neural networks (ANN) [8], and case-based technique [9]. In general, it is possible to use these classification techniques for spam filtering by using content-based filtering approach that will identify attributes (usually keywords often used in spam emails). The Frequency occurrence of these attributes within email determine the probabilities for each attribute within email, then it is compared to threshold value. Email messages that exceed the threshold value are classified as spam [10]. ANN is a non-linear model that tries to simulate the functions of biological neural networks. It consists of simple processing unit called neurons and processes information to do computation operations [11, 12]. Many types of researchused aneural network to classify spam using content-based filtering,these methods determines attributes to calculate the frequency of keywords or patterns in the email messages. Neural Network algorithms that are used in email filtering achieve reasonable classification performance. The most famous algorithms are Multilayer Perceptron Neural Networks (MLPNNs) and Radial Base Function Neural Networks (RBFNN). Researchers used MLPNNas aclassifier for spam filtering but very too few of them used RBFNNas aclassifier.In this paper, we will use RBFNN for spam filtering. RBFNN is one of the most important types of ANNs; which are characterized by other types of ANNs, including better approximation, better classification, simpler network structures and faster-learning algorithms [13].

Evolutionary algorithms (EAs), which are optimization techniques that avoid various mathematical complications, manage populations of solutions, trying to identify individuals representing the best solutions for the problem [14].There are many previous studies that combined Genetic Algorithms and Neural Networks [15] to improve the performance of neural network algorithms. A similar method of evolutionary computation techniques such as Genetic Algorithms (GAs) is Particle Swarm Optimization (PSO), which is a method for optimizing several continuous nonlinear functions and classification method. PSO is inspired by the social behavior of bird flocks and school of fishes, it used in many applications, like aneural network, telecommunications, signal processing, data mining, and several other applications. PSO algorithm operates on a population (swarm), with the characteristic of no crossover and mutation calculation like genetic algorithm; which makes it is easy to implement. PSO only evolve their social behavior accordingly their movement towards the best solutions [16].

In this paper, we proposed a hybrid approach that combines RBFNN with PSO algorithm (HC-RBFPSO) for spam email filtering. We use Particle swarm Algorithm to optimize the centers of RBFNN. KNN and SVD algorithms used to optimize the radii and the weights respectively within the PSO iterative process, which means in each iterative process of PSO, the weights and Radii are updated depend the fitness (error) function. Each particle in PSO consists of centers of hidden neuron for RBFNN. After running PSO algorithm a number of iterations, the process selects the best values of the centers$c$, radii r and weights w. To determine the optimal number of RBF (Neurons); we use the incremental method which depends on increasing the number of RBF by 1 in each iteration, and terminated when the process gets the threshold value.

This paper is organized as follows; section 2, presents related works of the proposed field. Section 3, describes the RBFNN in details. Section 4, explains the PSO algorithm in details. Section 5, shows the methodology. Section 6, shows the experimental results of our approach, conclusions are presented in Section 7.

2.RELATED WORK

The most widely used methods for spam filtering are artificial neural networks (ANN) and Bayesian method, also support vector machines, there are many techniques described for mail filtering and spam detection. Initially, in [17], they present a LINGER based on ANN; LINGER is theANN-based system used for automatic email classification. Although LINGER was tested in the email classification field, LINGER is a public architecture for all text classifications. LINGER is flexible, adaptive system and compatible for most operations. ANN can be effective to be used for spam filtering and automated email filing into mailboxes.In [18], they proposed spam filtering approach for Turkish in particular. Their approach is dynamic and depends on Artificial Neural Networks (ANN) and Bayesian Networks. This approach is dealing with the characteristics of the incoming e-mails. Through their experiments for 750 e-mails (410 spams and 340 hams) the accuracy achieved about 90%.

In [19], they combined Support Vector Machine (SVM) and Genetic Algorithm (GA) namely GA-SVM, GA is used to select features that are most favorite to SVM classifier. The experiments show that GA-SVM approachesbetter results than main SVM.In [20], they present a novel approach of spam filtering includes the seven several steps, that depends on using the history of previous mails and spam mails which are specific for each mailbox of theuser. Using the knowledge base, detection of spam mails is performed by using artificial neural network techniques. It also using keywords list to get some words in the incoming mail, then perform the detection operation. The proposed approach works well with all kinds of spam mails (text spam and image spam). Experiments results show that the detected spam at least 98.17 %. But the limitation of this approach is needed higher memory space and more hardware for execution. So to implement this approach for large mail servers, we need intelligent mail servers. In [21], they present Continuous Learning Approach Artificial Neural Network CLA_ANN, which includesmodifying core modifications on ANN in the input layers, which allow the input layers to be changed over time and to replace useless layers with new promising layers which give best results. The experiment result of CLA_ANN by using 300 input layers and using Spam Assassin dataset, achieves results with 0.534 % false positive and 3.668% false negative.In [22], they present a new technique for filtering spam; this technique consisted of a single perceptron that was designed to distinguish between spam and legitimate mail messages. The perceptron algorithm gives suitable detection rates; this is due to the incorporation of a continuous learning feature. The results show that the best false positive value is found when the number of iterations is 900.In [23], they proposed PSO-LM approach which is a new learning method for process neural networks (PNNs) based on the Gaussian mixture functions and particle swarm optimization (PSO).According to experiments results, PSO-LM had better performance on time-series prediction and pattern recognition than basis function expansion based learning (BFE-LM) and back propagation neural networks (BPNNs). But in PSO-LM approach, it needed more computations for the global search strategy of PSO.

3.RADIAL BASIS FUNCTION NEURAL NETWORKS

Radial Basis Function Neural Networks are thetype of neural networks whose activation functions in the hidden layer are radially symmetrical.Its output depends on the distance between a vector that stores the input data and a vector of weights, which is called the center. RBFNN has been used in many applications, such as function approximation, system control, speech recognition, time-series prediction, and classification. [24]. The RBFNN has three feed-forward layers: the input layer, the hidden layer, and output layer. The input data flow from theinput layer to send the information to the hidden layer. The hidden layer neurons are activated depending on the distance between each input pattern with the centroid stores each hidden neuron, which determines the structure behavior of network. In hidden layer may be used different types of activation functions, but the most type commonly used in themost application is the Gaussian Function. The output layer calculates the linear sum of values of the hidden neuron and outputs it [25]. Figure 1 shows the architecture of RBFNN that including three layers for classification into two categories.

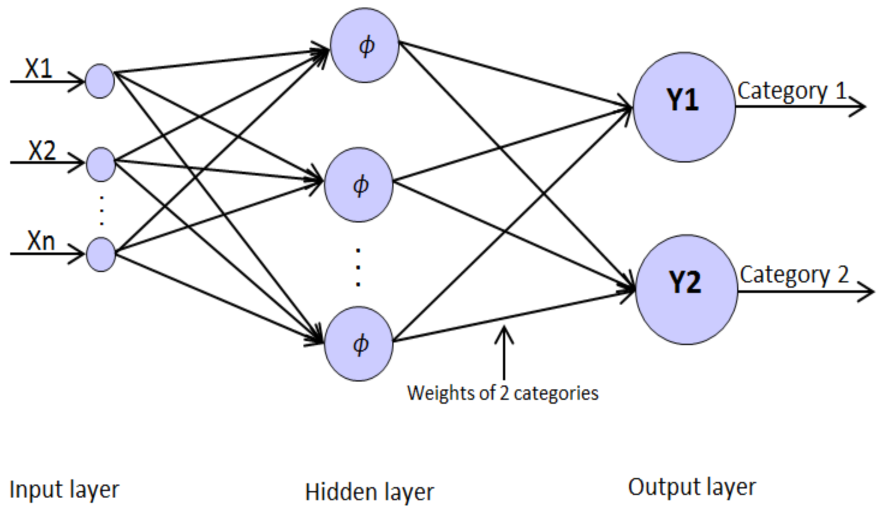

Figure 1. Architecture of RBFNN

Two groups of parameters need to be determined in RBFNN [26]: the first category is the center and radii, and the second one is the connection weights between the hidden layer and output layer. The Gaussian activation function in the hidden layer (Φ) is calculated as follows in equation 1:

$$\Phi = \exp\left[\frac{-(\mathrm{x}-\mathrm{cj})^2}{2\sigma\mathrm{j}^2}\right] \tag{1}$$

Where $x = (x_1, x_2, x_3 \ldots, x_n)$ is the input data, $c_j$is the center of *j-th* hidden neuron, and $\sigma_j$is the width of *j-th* hidden neuron. The output of RBFNN for each category is calculated as in the following equation 2:

$$Y = \sum_{k=1}^{m} Wjk * \Phi j(x) \tag{2}$$

Where $k=1, 2, 3 \ldots, m$ is the number of nodes in the hidden layer. $W_{jk}$are the connection weights values between the *j-th* hidden layer nodes and the *k-th* output nodes.

In RBFNN to reach the best classification, each output node computes a sort of score for the associated category. Generally, classification decision is done by assigning the input to the category with the highest score. The score is computed by taking a weighted sum of the activation values from every RBF neuron.

One important parameter of RBFNN is the determination of the suitable number of neurons in hidden layer, which affects the network complexity and the generalization of the RBFNN. If the number of neurons is too small, the accuracy of the output will decrease. On the other hand, if the number of neurons is too large, this cause over-fitting for the input data [27, 28]. In this paper, we use the incremental method to compute classification accuracy for a specific number of neurons.

3.PARTICLE SWARM OPTIMIZATION

Particle Swarm Optimization mimics the behaviors of birds flocking. In bird flocking, a group of birds is randomly searching food in thecertain area. All the birds do not know where the food is found, but they know how far the food in each iteration. PSO algorithm is a fast optimization method due to need to adjust few parameters, which has fast convergence speed, high robustness, and strong global search capability, does not require gradient information and is easy to implement [29].

The population in PSO is called swarm that is constitute of particles (solutions) and then searches for theoptimal position of particles by updating it for all iterations. In all iterations, each particle is updated by following two best fitness values that are evaluated by using proper fitness function according to theproblem. The first value is the best position it has achieved so far for each particle; this value is called personal best (pbest). Another value is the best position for the entire swarm obtained so far by any particle in the swarm; this best value is called a global best (gbest). All particles have avelocity which determines thedirection of the particles and moves it to the new position.

The basic algorithm of PSO is as following steps:

- **Initialize** each particle *i* of the swarm, with random values for theposition*(Xi)* and velocity *(Vi)* in the search space according to the dimensionality of theproblem.
- **Evaluate fitness** value of particle by using fitness function.
- **Compare** the value obtained from the fitness function from particle*i* with the value of Pbest.
- **If** the value of the fitness function is better than the Pbest value, then update the particle position to takes the place of Pbest.
- **If** the value of Pbestform any particle is better than gbest, then update gbest = Pbest.
- **Modify** the position *Xi* and velocity *Vi* of the particles using equations 3 and 4, respectively.
- **If** the maximum number of iterations or the ending criteria is not achieved so far, then return to step 2.
-

$$\mathrm{Vid}\left(t+1\right)=\omega^{*}\mathrm{Vid}\left(t\right)+\mathrm{c1r1}\left(\mathrm{Pid}\left(t\right)-\mathrm{Xid}\left(t\right)\right)+\mathrm{c2r2}(\mathrm{Pgd}\left(t\right)-\mathrm{Xid}(t)) \quad (3)$$

$$\mathrm{Xid}\left(t+1\right)=\mathrm{Xid}\left(t\right)+\mathrm{Vid}\left(t+1\right) \quad (4)$$

Where $i = 1,2,\ldots,M$; d = 1,2,…,n, $t+1$ is the current iteration number, t is the previous iteration number, ω is the inertia weight, c_1 and $c_2$are the acceleration constant which is usually between [0,2], $P_i = (P_{i1}, P_{i2},\ldots, P_{in})$ is the best previous position of particle*i*knew as the personal best position (pbest), $P_g = (P_{g1}, P_{g2},\ldots, P_{gn})$ is the position of the best particle among all the particles in the swarm known as the global best position (gbest), and $r_1$and r_2 are random numbers distributed uniformly in (0, 1). The fitness function that is used in step 2 of PSO is Root Mean Square Error (RMSE) for this paper; RMSE is shown in equation 5.

$$\mathrm{RMSE}=\sqrt{\frac{1}{n}\sum_{i=1}^{n}(T-Y)^2} \quad (5)$$

Where n is the number of input data, T is the target output, and Y is the real output.

4.METHODOLOGY

PSO has been used to optimizethe parameters of RBFNN in several sides like network architecture, learning algorithm, and network connections. In PSO every single solution is called a particle. Using fitness function (RMSE) to evaluate the particles for theoptimal solution. In this paper, we present a novel approach called HC-RBFPSO; this approach is a combined RBFNN with PSO to reduce the number of neurons for RBFNN and improve classification accuracy for spam filtering by using RBFNN, here select the best values of RBFNN centers by using PSO. The remaining parameters of RBF, we use tradition algorithms namely Knn and SVD to optimize radii and weight respectively within the PSO iterative process, which means in each iterative process of PSO, the weights and Radii are updated depending the fitness (error) function [41]. HC-RBFPSO approach is explained in the following pseudo code:

Start with one RBF (Neuron).
Initialize RBFNN parameters.

- *Initialize the centers c randomly from theSPAMBASE dataset.*
- *Use **Knn** to initialize Radii r.*
- *Use **SVD** to initialize Weights w.*

Start optimizing centers of RBFNN using PSO.
Initialize particles position randomly from the dataset and initialize velocity randomly between (0, 1)
While not reach the maximum numbers of iteration ***do***
For each particle do
Calculate fitness value (RMSE between Real output of RBFNN and Target)
If fitness value is better than best fitness value ***pbest*** in particle then
Set current position as***pbest***
End
Select*gbest* of the particle which the best fitness value among all particles in current iteration
Calculate particle velocity based on equation 3.
Update particle position (centers) based on equation 4.
End
End
Take the ***pbest*** of particle as centers ***c*** of RBFNN,
Complete training RBFNN using Knn and SVD.
Calculate the real output of RBFNN.
Calculate the classification accuracy for training and testing phase.
Repeat

It should be noted that inertia weight ω in PSO was first introduced by Shi and Eberhart [30] in order to control the search speed and make particles converge to local minima quickly. Inertia weight ω often have restricted between two numbers during a run. Here in this work, we calculate the inertia weight by using equation 6.

$$\omega = \omega\max - \frac{(\omega\max - \omega\min)*t}{T\max} \tag{6}$$

Where ω is the inertia weight, ω_{max} is the maximum of ω (here ω_{max}=0.9), ω_{min} is the minimum of ω (here ω_{min}=0.4), t is the current iteration number, and T_{max} is the maximum iteration number.
In this paper, the training RBFNN is stopped when reaches the maximum number of iterations or got the percentage of given accuracy. K-Nearest Neighbors is used to determining the width r of

each RBF. Knn is a simple algorithm used for classification and regression [31]. Knn stores all available cases and classifies new cases based on a similarity measure (e.g. Euclidean distance functions). Knn is types of thelazy learning algorithm, where the function is only approximated locally and all computation continues until classification [32] but this algorithm has been successful in many numbers of classification and regression problems; such as handwritten digits and satellite image scenes [33]. The number k is used to decide how many neighbors influence the classification for new value. Mathematical calculations regarding Knn algorithm; shown in equation 7.

$$D(x,y)^n = sqrt\left[\ \sum_{i=1}^{k} (xi - yi)^2\ \right] \tag{7}$$

We use Singular Value Decomposition (SVD) to optimize the weights of output layer for RBF, SVD is a powerful and useful matrix decomposition has been used in many fields such as data analysis, reducing dimension transformations of images, data compression, signal processing, and pattern analysis [34]. If $A \in R^{m\,x\,n}$, there exist orthogonal matrices $S \in R^{m\,x\,m}$ and $H \in R^{n\,x\,n}$ such that:

$$S^T\ A\ H\ = diag(\sigma_1,...,\sigma_p) \tag{8}$$

Where p is the minimum of (m,n), σ are the singular values of A. The use of SVD technique to calculate the optimal weights w of the RBFNN depends on using matrix notation described in the following reducing expression:

$$\vec{Y}\ =\ \vec{w}\ \Phi \tag{9}$$

Where Y is the real output of RBFNN, w are the weights vector, and Φ is the Gaussian activation function matrix. Using the next following expression:

$$A = H\ diag(\frac{1}{k})\ S^T \tag{10}$$

Where $k\ = diag(\sigma_1,...,\sigma_p)$, by replacing Φ in (9), using (10); the weights vector (11):

$$\vec{w}\ = \left[H\ diag(\frac{1}{k})\ S^T\right]\vec{y}\,(11)$$

SVD can solve any equation system. In the proposed case SVD effect in reducing the of the output error, it can also be used to remove any RBF when its associated singular value had a small value or if the approximation error can't affect the result.

4.1 Dataset

In this paper, we use SPAMBASE dataset to classify email as spam or non-spam. It is downloaded from UCI Machine Learning Repository site [35]. SPAMBASE was proposed by Mark Hopkinsand in his colleagues. SPAMBASE dataset is multivariate dataset contains data from a single email account. SPAMBASE contains 4601 record previously identified – 1813 classified as non-spam (39.4%) and 2788 classified as spam (60.6%).SPAMBASE is containing fifty-seven data attributes and one classification attribute to determine the type email (value 0 for non-spam and value 1 for spam). Most of the attributes (1-54) express particular words or characters were frequently in anemail. The attributes (55-57) measure the length of sequences of consecutive capital letters.

Here we will introduce definitions of the attributes:

- Attributes (1-48) 48 continuous real attributes of type word_freq_WORD = percentage of words in the e-mail that match WORD.
- Attributes (49-54) 6 continuous real attributes of type char_freq_CHAR = percentage of characters in the e-mail that match CHAR.
- Attribute (55) 1 continuous real attribute of type capital_run_length_average = average length of uninterrupted sequences of capital letters.
- Attribute (56) 1 continuous integer attribute of type capital_run_length_longest = length of longest uninterrupted sequence of capital letters.
- Attribute (57) 1 continuous integer attribute of type capital_run_length_total = sum of thelength of uninterrupted sequences of capital letters = a total number of capital letters in the e-mail.

4.2 Preprocessing

The available data in the SPAMBASE dataset is in numeric form. All thefifty-seven attributes in the SPAMBASE dataset mostly represent frequencies of various words and characters in emails; we wish to normalize this data before running HC-RBFPSO approach. Normalization processing is an important stage due to speeding up model, convergence and reducing the effect of imbalance in data to the classifier. In the training and testing stages for this paper, normalize the data in the range [0, 1].

5. EXPERIMENTS AND RESULTS

In the file SPAMBASE.data from UCI repository site, each column contains attribute value for each email, in each record, the data is delimited by commas. In this research, we use the SPAMBASE dataset after converted to CSV (Comma Separated Values) file compatible with Matlab environment. We will split the data into two phases, training set (70% data) and testing set (30% data). Then the performance was measured by evaluating the accuracy for each phase. To eliminate any data particular behavior, the data is selected randomly from thedataset for training and testing sets.In our approach HC-RBFPSO, we use the most common approach to finding the performance of spam filtering is the classification accuracy [36].Classification Accuracyis the proportion of instances which are correctly classified. We compute Classification Accuracy as shown in equation 12.

$$Accuracy = \frac{(TP + TN)}{N} \tag{12}$$

Where *TP* is the True Positive, *TN* is the True Negative, and N is the size of samples data.To compare the classification accuracy of our approach, we conduct multiple experiments of this work including run the proposed approach (HC-RBFPSO) on the training data and test data for a different number of neurons, after that we compare our results with other previous works.

The proposed approach HC-RBFPSO is experimented using MATLAB 2012 under Windows 7 with i5-3210M CPU 2.5GHz, 4GB RAM memory.

The parameters of the PSO algorithm that are used in this paper were set as inertia weight ω in calculated based on the equation 6, the rests of parameters are illustrated in table 1.

Table 1. Parameters for PSO used in HC-RBFPSO

Parameter	**Value**
Iteration	100
c1	1.4
c2	1.4

The particle size is an important factor in HC-RBFPSO when the algorithm uses small particle numbers it produce poor performance. But, large particle number produces a very residual improvement compared to the improvement that occurs when using Median particle size but increases the computational cost of the algorithms. From the results obtained you can see that HC-RBFPSO approach has a good performance in the classification accuracy. As shown the Table 2, the proposed approach used little number of neurons, therefore, asmall amount of the computational cost.Now, we present the results for training and testing of HC-RBFPSO approach. The experiments are a number of conducted by using adifferent number of neurons in RBFNNhidden layer as shown in table 2. We recorded the results when using 10, 20, 30, 40, and 50 neurons for each class.

Table 2. Experiment result and comparison

Approach	# Hidden Neuron	Accuracy	
[37] – RBF	---	84.3 %	
[38] – MLP	---	93.28 %	
[39] – ANN	30	91 %	
[40] – MLP	---	91.85 %	
Proposed Approach HC-RBFPSO	10	Training phase	89.5 %
		Testing phase	88.5 %
	20	Training phase	91 %
		Testing phase	90.1 %
	30	Training phase	91.8 %
		Testing phase	90.9 %
	40	Training phase	92.5 %
		Testing phase	91.4 %
	50	Training phase	93.1 %
		Testing phase	90.6 %

From table 2, we note that most previous studies compute the classification accuracy without determines the number of hidden neurons but in our approach for each experiment determines the number of neurons used and the classification accuracy. We note that from experiments on our approach be effective in spam filtering messages using a small number of neurons in a large dataset.

6. Conclusion

Radial Basis Function Neural Networks (RBFNN) is one of the most important types of artificial neural networks (ANNs). It is characterized bybetter approximation, simpler network structures, and faster learning algorithms. In this paper, we proposed a hybrid approach (HC-RBFPSO), that combining RBFNN and PSO in the purpose to classifyEmail spam problems. PSO has been used to improve RBFNN in several sides like network architecture, learning algorithm and network connections. In this paper, we use PSO to find optimal centers of hidden neurons in RBFNN; it is also used traditional Knn algorithm to optimize the width of the RBFNN and SVD technique to optimize the weight of RBFNN. The proposed method applied used large SPAMBASE dataset, which presents a collection of spam and non-spam emails with 57 attributes. The results obtained from the experiments are comparable with other approaches those that use the same dataset show that better performance in terms of accuracy of the proposed approach.The results of the simulations show that HC-RBFPSO is an effective method that is a reliable alternative for classification. The quality of the results improves the convergence. Generally, we can conclude that the main contributions of this paper which have been achieved through the application of the proposed method HC-RBFPSO on the benchmark of spam dataset. This method has enabled build a spam filtering system based on combining RBFNN and PSO.

REFERENCES

[1] De Morais Fonseca, O. H., Fazzion, E., Cunha, I., Las-Casas, P., Guedes, D., Meira, W., ...& Chaves, M. (2016) “Measuring Characterizing, and Avoiding Spam Traffic Costs” IEEE Internet Computing Volume:PP , Issue: 99.

[2] Email Statistical Report, http://www.radicati.com/wp/wp-content/uploads/2010/04/Email-Statistics-Report-2010-2014-Executive-Summary2.pdf.

[3] Gaikwad, B. U., &Halkarnikar, P. P. (2014) “Random Forest Technique for E-mail Classification”, International Journal of Scientific & Engineering Research, 5(3), 145-153.

[4] Gudkova, D. (2013) Kaspersky Security Bulletin, Spam Evolution 2013.

[5] UC Irvine Spam Stat istics (2013) http://www.oit .uci.edu/email/spam/stats/

[6] Bahgat, E. M., Rady, S., & Gad, W. (2016). “ An e-mail filtering approach using classification techniques”. In The 1st International Conference on Advanced Intelligent System and Informatics (AISI2015), November 28-30, 2015, BeniSuef, Egypt (pp. 321-331). Springer International Publishing.

[7] Bouguila, N., &Amayri, O. (2009) “A discrete mixture-based kernel for SVMs: application to spam and image categorization”, Information Processing & Management, 45(6), 631-642.

[8] Cao, Y., Liao, X., & Li, Y. (2004) “An e-mail filtering approach using neural network”, In International Symposium on Neural Networks (pp. 688-694). Springer Berlin Heidelberg.

[9] Fdez-Riverola, F., Iglesias, E. L., Diaz, F., Méndez, J. R., &Corchado, J. M. (2007) “SpamHunting: An instance-based reasoning system for spam labelling and filtering”, Decision Support Systems, 43(3), 722-736.

[10] S. Mason. New Law Designed to Limit Amount of Spam in e-mail.

[11] Stuart, I., Cha, S. H., &Tappert, C. (2004) “A neural network classifier for junk e-mail”, In Document Analysis Systems VI (pp. 442-450). Springer Berlin Heidelberg.

[12] Han, J., Kamber, M., & Pei, J. (2011) Data mining: concepts and techniques, Elsevier.

[13] Lee, C. C., Chung, p. C., Tsai, J. R., & Chang, C. I. (1999) “Robust Radial Basis Function Neural Networks”, Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE ,29, 674 – 685.

[14] Qasem, S. N., Shamsuddin, S. M., & Zain, A. M. (2012) “Multi-objective hybrid evolutionary algorithms for radial basis function neural network design”, Knowledge-Based Systems, 27, 475-497

[15] Schaffer, J. D., Whitley, D., &Eshelman, L. (1992) “Combinations of genetic algorithms and neural networks: a survey of the state of the art”, Combinations of Genetic Algorithms and Neural Networks, 1–37.

[16] Elbeltagi, E., Hegazy, T., & Grierson, D. (2005) “Comparison among five evolutionary-based optimization algorithms”, Advanced Engineering Informatics, 19, 43 – 53.

[17] Clark, J., Koprinska, I., & Poon, J. (2003) “A neural network based approach to automated e-mail classification”,In null (p. 702). IEEE.

[18] Özgür, L., Güngör, T., &Gürgen, F. (2004) “Adaptive anti-spam filtering for agglutinative languages: a special case for Turkish”, Pattern Recognition Letters, 25(16), 1819-1831.

[19] Wang, H. B., Yu, Y., & Liu, Z. (2005) “SVM classifier incorporating feature selection using GA for spam detection”, In Embedded and Ubiquitous Computing–EUC 2005 (pp. 1147-1154). Springer Berlin Heidelberg.

[20] Tak, G. K., &Tapaswi, S. (2010) “Query Based approach towards spam attacks using artificial neural network”, International Journal of Artificial Intelligence & Applications (IJAIA), 1(4).

[21] Sabri, A. T., Mohammads, A. H., Al-Shargabi, B., &Hamdeh, M. A. (2010) ”Developing new continuous learning approach for spam detection using artificial neural network (CLA_ANN)”, European Journal of Scientific Research, 42(3), 525-535.

[22] Kufandirimbwa, O., &Gotora, R. (2012) “Spam Detection using Artificial Neural Networks (Perceptron Learning Rule)”, Online J Phys Environ Sci Res, 1(2), 22-29.

[23] Liu, K., Tan, Y., & He, X. (2010) “Particle swarm optimization based learning method for process neural networks”, In Advances in Neural Networks-ISNN 2010 (pp. 280-287). Springer Berlin Heidelberg.

[24] Niu, D., Lu, Y., Xu, X., & Li, B. (2014) “Short-Term Power Load Point Prediction Based on the Sharp Degree and Chaotic RBF Neural Network”, Mathematical Problems in Engineering, 2015, 8.

[25] Sangwan, O. P., Bhatia, P. K., & Singh, Y. (2011) “Radial basis function neural network based approach to test oracle”, ACM SIGSOFT Software Engineering Notes, 36(5), 1- 5.

[26] Han, H. G., Chen, Q. L., &Qiao, J. F. (2011) “An efficient self-organizing RBF neural network for water quality prediction”, Neural Networks, 24(7), 717-725.

[27] Mashor, M. Y. (1999) “Some Properties of RBF Network with Applications to System Identification”, International Journal of Computer and Engineering Management, 7.

[28] Ou, Y. Y., Gromiha, M. M., Chen, S. A., &Suwa, M. (2008)“TMBETADISC-RBF: Discrimination of β-barrel membrane proteins using RBF networks and PSSM profiles”, Computational Biology and Chemistry, 32(3), 227–231.

[29] Modares, H., Alfi, A., & Sistani, M. N. (2010) “Parameter estimation of bilinear systems based on an adaptive particle swarm optimization”, Engineering Applications of Artificial Intelligence, 23(7), 1105-1111.

[30] Shi, Y., &Eberhart, R. (1998) “A Modified Particle Swarm Optimizer”, Evolutionary Computation Proceedings, IEEE World Congress on Computational Intelligence (pp. 1945–1950).

[31] Altman, N. S. (1992) “An introduction to kernel and nearest-neighbor nonparametric regression”, The American Statistician, 46 (3), 175–185.

[32] Song, Y., Huang J., Zhou, D., Zha, H., & Giles, C. L. (2007) “IKNN: Informative K-Nearest Neighbor Pattern Classification”, Knowledge Discovery in Databases, 4702, 248–264.

[33] Weinberger, K. Q., & Saul, L. K. (2009)“Distance Metric Learning for Large Margin Nearest Neighbor Classification”, Journal of Machine Learning Research, 10, 207-244.

[34] Golub, G. H., & Van Loan, C. F. (2013) Matrix Computations, Hopkins University Press.

[35] Spambase Data Set https://archive.ics.uci.edu/ml/datasets/Spambase. 18/4/2016, 22:35.

[36] Sokolova, M., Japkowicz, N., &Szpakowicz, S. (2006) “Beyond accuracy, F-score and ROC: a family of discriminant measures for performance evaluation”, In AI 2006: Advances in Artificial Intelligence (pp. 1015-1021). Springer Berlin Heidelberg.

[37] Costa, K., Ribeiro, P., Camargo, A., Rossi, V., Martins, H., Neves, M., & Papa, J. P. (2014)“Comparison of the Intelligent Techniques for Data Mining in Spam Detection to Computer Networks”.

[38] Sharma, S., & Arora, A. (2013) “Adaptive approach for spam detection”, IJCSI International Journal of Computer Science Issues, 10(4), 23-26.

[39] Jukic, S., Azemovic, J.,Keco, D., &Kevric, J. (2015) “COMPARISON OF MACHINE LEARNING TECHNIQUES IN SPAM E-MAIL CLASSIFICATION”, Southeast Europe Journal of Soft Computing, 4(1).

[40] Faris, H., Jaradat, K., Al-Zewairi, M., &Adwan, O. (2015)“Improving knowledge based spam detection methods: The effect of malicious related features in imbalance data distribution”, International Journal of Communications, Network and System Sciences, 8(5), 118.

[41] Monir Foqaha Mohammed Awad, (2016) “Hybrid Approach to Optimize the Centers of Radial Basis Function Neural Network Using Particle Swarm Optimization”, Journal of Computers (JCP), Volume 12, Number 5, 2016.

AUTHORS

Mohammed Awad received the Ph.D. degrees in Computer Engineering from the Granada University Spain. From 2005 to 2006, he was a contract Researcher at Granada University. Since Feb. 2006, he has been an Assistant Professor in Computer Engineering Department, College of Engineering and Information Technology at Arab American University. At 2010 he has been an associate Professor in Computer Engineering. He worked for more than 10 years at the Arab American University in academic Position, in parallel with various Academic administrative positions. (Departments Chairman and Dean Assistant, and Dean of Scientific Research/Editor-In-Chief, Journal of AAUJ). Through the research and educational experience, he has developed strong research record. His research interests include: Artificial Intelligence, Neural Networks, Function Approximation of Complex Systems, Clustering Algorithms, Input Variable Selection, Fuzzy Expert Systems ,Genetic Algorithms and Particle Swarm Optimization.

Monir Foqaha is Master Student in Computer Science programin the Faculty of Engineering and Information Technology at Arab American University of Jenin, Palestine. He received his Bachelor's degree in Computer Science from An-Najah National University of Nablus, Palestine in 2007. His research interests include Neural Networks and optimization algorithms.