ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645
Indian Journal of Science and Technology, *Vol 8(S9), 260–272, May 2015* DOI: 10.17485/ijst/2015/v8iS9/55320

Spam E-mail Filtering using ECOS Algorithms

Ammar Almomani[1*], Atef Obeidat[1], Karim Alsaedi[2], M. Al-Hazaimeh Obaida[1] and Mohammed Al-Betar[1]

[1]Department of Information Technology, Al-Huson University College, Al-Balqa Applied University, P.O. Box 50, Irbid, Jordan; ammarnav6@gmail.com, atefob@gmail.com, dr_obaida82@yahoo.com, moh_betar@yahoo.com
[2]Department of Computer Science, College of Science, Al-Mustainsiriyah University, P.O. Box 46173, Baghdad, Iraq; karimnav6@gmail.com

Abstract

Spam known as Unsolicited Bulk E-mail (UBE) including undesirable electronic correspondence that is sent in bulk to massive mailing lists, sometimes with some business nature sent get into bulk. One of the main problems of spam E-mail detection is an attack from an unknown source known as 'zero-day' so named due to continued changes in timing. Zero day attacks are problematic for having the ability to escape spamming detection as the hosts do not show up in blacklists. This adds to the problem of False positives and OCR computational cost especially in dealing with a huge corpus of texts with images that run through server-side filters. Spammers are busy orchestrating various representation techniques thus making their 'zero-day' spam E-mail that infiltrate the defenses of detection. Our proposed is a novel system called Spamming Dynamic Evolving Neural Fuzzy System (SDENFS), which adapts the Evolving Connectionist System (ECoS) based on a hybrid (supervised/unsupervised) learning approach. SDENFS adaptive online is enhanced by offline learning to detect dynamically the spamming E-mail included unknown zero-day spamming E-mails before it get to user account. SDENFS is suggested to work for high-speed "life-long" learning with low memory footprint with few number of rules creation for E-mail classification. Two datasets composed of 6612 samples of spam and legitimate E-mails were used to assess the proposed system. The proposed system showed a high level of performance in detecting spam E-mail attacks. The average of the accuracy and F-measure of the classification process was 99%.

Keywords: (EcoS) Evolving Connectionist System, Filtering, Spamming E-mail, Zero_Day Attack

1. Introduction

E mail is he 'Achilles heel' for electronic correspondence which gives continuous headache to governments, organizations and businesses that heavily depend on electronic correspondence for a variety of purposes.

Spam (Unsolicited business E-mails) have grown from around 10% in 1998 to more than 80% of overall E-mail volume nowadays[1,2]. The explosive growth of uninvited E-mails has prompted the development of varied spam filtering techniques. However, current spam filters are often simply poisoned by avoiding spam keywords and adding several innocuous keywords within the E-mails. Additionally, they have a significant quantity of your time to adapt to a brand new spam supported user feedback. Moreover, few current spam filters exploit social networks to help spam detection.

E-mail spam or junk E-mail (unwanted E-mail "usually of an advertisement nature sent out in bulk") is one of the major issues of the today's net, transportation money harm to corporations and annoying individual users. Among the approaches developed to prevent spam, filtering is a very important and fashionable one. Common uses for mail filters embody organizing incoming E-mail and removal of spam and PC viruses. A less common use is to examine outgoing E-mail at some corporations to guarantee that staff abides by with acceptable laws. Users may also use a mail filter to order messages, and to type

*Author for correspondence

them into folders supported subject material or alternative criteria. Mail filters are often put in by the user, either as separate programs, or as a part of their E-mail program (E-mail client). In E-mail programs, users will build personal, "manual" filters that then mechanically filter mail in line with the chosen criteria. However, Saudi Arabia is top 5 activity for Spam destination by geography in 2013[3] Symantec's Monthly Intelligence Reports, that monitor spam, virus and spam rates round the world, have recorded Saudi Arabia in concert of the highest recipients of in-bound spam-amounting to 82.7% in January-2013[4].

A huge quantity of spam is being generated each day and waste important web resources also as users time. It has been projected that E-mail traffic would reach 419 billion E-mails per day, out of that 83% area unit going to be spam, that translated into 347 billion spam E-mails every day. Spam attacks each the PC and its users. Spam E-mail will contain viruses, key loggers, spam attacks and more. These sorts of malware will compromise a user's sensitive personal information by capturing bank account information such as username and passwords[5].

Spam E-mails these days are particularly dangerous in what is known as "zero-days" which refer to attacks that spammers launch using techniques to evade black lists of spam detection[6–9].

spam E-mail is so complex that it cannot be detected by many of current techniques because the spammer can use new vulnerabilities which are never seen before10. There are a number of possible solutions to spam but not effective yet[10]. These ranges from communication-oriented approaches like authentication protocols over blacklisting to content-based filtering approaches which usually depend on some of Artificial Intelligence (AI) techniques[11]. Current AI algorithms are able to detect spamming E-mail based on fixed features and rules while a few number of machine learning algorithms design to work in online mode[12]. It is expected that errors will multiply with time particularly when handling zero-day spams in the classification process[13].

Many studies these days heavily focus on spam E-mails detection.

We propose Spamming Dynamic Evolving Neural Fuzzy System (SDENFS) which changes and moves continuously and dynamically. By means of such system E-mail spamming or ham (legitimate) can be determined. When implementing the system, it adapts the evolving clustering method (ECM) which is integral to the Dynamic Evolving Neural Fuzzy Inference System (DENFIS) in an online mode[14,15] along the Dynamic Neural Fuzzy Inference System (DyNFIS) to improve the rule creation in an offline mode[16]. The proposed system has the capacity to detect spamming E-mail by evolving stream data mining that leads to improve classification performance with a high degree of efficiency and it is characterized by life-long learning with low memory footprint[17].

The recent paper will be organized as follows. Related works are presented in Section 2. Section 3 provides discussions to the proposed system. The experimental design and test results are given in Section 4 and finally, Section 5 presents the conclusions and future work.

2. Related Works

Spamming E-mails filtering methods depends on classification techniques which can be managed by several ways, such as features extraction, machine learning technique and clustering methods for detecting spamming E-mails. Many approaches have been proposed including the features extraction technique supposed by[13]. His approach is a system and methodology for removing ineffective options from a spam feature set. Above all, associate entropy worth is calculated for the feature set supported the effectiveness of the feature set at differentiating between ham and spam. Despite its advantages, this technique is still lacking as to the ability of detecting zero-day spamming E-mail as it relies on supervised learning algorithm.

Resent researcher depends on machine learning technique for detecting spamming E-mails[13]. Machine learning technique is of three types used in the field of spamming E-mail. These include supervised learning, unsupervised learning and some of them used hybrid learning based on classifiers. The basics of the classifiers rely on learning several inputs or aspects to anticipate a desirable output. A quick summary of the algorithms will be dealt with in this paper in addition to a discussion of the Naive theorem classifier, the k-NN classifier, the neural network classifier, the support vector machine classifier and k-means algorithms.

The Naive Bayes classifier could be a straightforward applied mathematics formula with a historical record of giving interestingly good results. Thus, it has been applied in many spam classification studies[16,17]. For the purpose

of classifying Associate in Nursing instance of unknown class, the "naive" version of Bayes's rule is utilized to initially work out the chance of the instance happiness to the spam category, and accordingly the chance of happiness to the not-spam category. Then it normalizes the primary to the sum of both to provide a spam confidence score between 0.0 and 1.0; However, as a result of such a big amount of little possibilities area unit being increased with one another. This may become a drag for finite exactitude floating purpose numbers and increase the error rate with future[18].

K-Nearest Neighbor (k-NN) is supervised learning approach[19]. Via this classifier, the decision is made as follows: Considering the k-nearest training input, samples are chosen using a pre-defined similarity function; Later, the E-mail x is tagged as being related to the same class as the bulk among this set of k samples, the k-NN can be a simple yet effective method, but not effective with zero-day attack[19].

Neural Networks (NNet) classifiers20, which are made up of three layers (input layer, hidden layer, and output layer), obtain the requisite knowledge through training the system with both the input and output of the problem in question. Refining the network goes on for having results with acceptable accuracy; the power of NNet is derived from the nonlinearity of the hidden neuron layers. Nonlinearity has high importance for the network learning of complex mappings. Sigmoid function is the commonly-used function in neural networks[21]. Nonetheless, NNe has a shortcoming realized in the problem of retaining useful learning information for the future[12,19,22], and this will increase the error rate with zero-day attack.

Support Vector Machines (SVMs) are especially effective and have gained momentum in a wide variety of applications pertaining to machine learning issues. These algorithms divide the n-dimensional area illustration of the information into 2 regions exploitation a hyperplane. This hyperplane always maximizes the margin between the 2 regions or categories, instead of linear hyperplanes. Several implementations of those algorithms use alleged kernel functions which lead to non-linear classification surfaces, like polynomial, radial or sigmoid surfaces[20], but this formula style to Figure with supervised learning approach thus it's weak to resolve zero-day attack and overwhelming memory.

k-means clustering[23] is one of the most used clustering techniques. It is an offline and unsupervised algorithm sets out to determine the cluster k as the assumed center of this cluster. Selecting any random E-mail object or features vector can be carried out as initial center then the process continues: determine the centre coordinate; determine the distance of each E-mail object (vector) to the centre group of E-mail objects based on a minimum distance[24]. K-means algorithm is not enough most especially since a k-means algorithm works offline only.

3. Proposed System-SDENFS

SDENFS collects and filters E-mails separately and sequentially. In the proposed system, ECOS[25] is adjusted considering similarity among more than 56 of spam E-mail features, as presented in Figure 1.

Figure 1 shows a block diagram of SDENFS. The proposed methodology is divided into three stages. The first stage is pre-processing; whereby 56 sub-features from E-mails are extracted for the most part similar to Spam Assassin 4 (SA)[26]. The second stage comprises the Evolving Clustering Method (ECM) and its offline extension (ECMc), which generates the basis of rules[27]. Finally, the Dynamic Evolving Neural Fuzzy Inference System (DENFIS)[14] is utilized in online mode as a fuzzy inference system to create, update, or delete a fuzzy rule while the system is running. The Dynamic Neural Fuzzy Inference System (DyNFIS)[16] is also used to enhance the rules in offline mode, improve the level of classification accuracy, and decrease the error rate in the prediction process based on Gaussian Membership Function (MF) and Back-Propagation Algorithm (BP). Yet it is suggested that the system will establish order in the relationship between ECM and ECMc and in the relationship between DENFIS and DyNFIS by employing the enhanced rule created from DyNFIS to be used with the rule generated by DENFIS[17]. The first SDENFS stage implemented by pre-processing is explained as follows.

3.1 Pre-Processing

Pre-processing involves two steps: first, selecting spam E-mail features; and secondly, parsing and stemming of E-mails. While parsing does extraction to the features of spam E-mails, stemming serves to clean the text data merged with the features of a spam E-mail.

A total of 56 features are selected to obtain better accuracy in terms of the classification process, and to represent the most effective characteristics of spam E-mails.

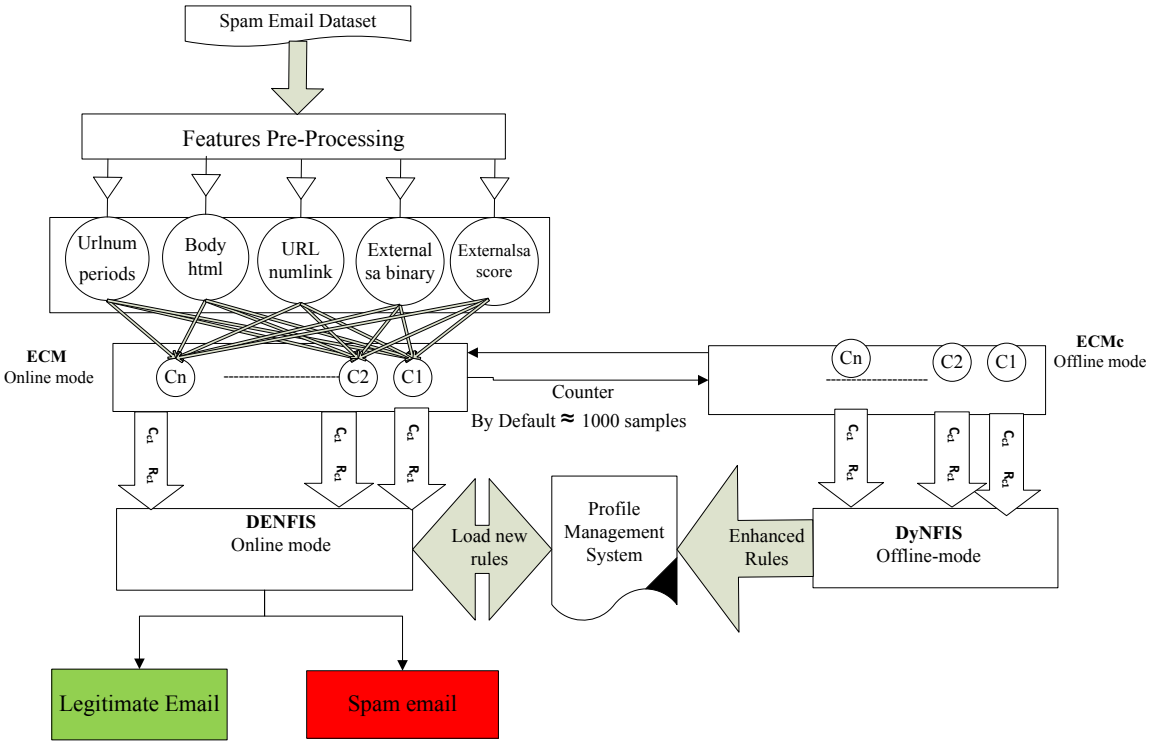

Figure 1. SDENFS.

The features selected in the proposed system are discussed in the next section.

3.1.1 Selected Features of Spam E-mails

SDENFS uses the features of spam E-mails in the first learning process as seed to detect spam E-mails. The proposed system adopts unknown values of the features vector from the evolving clustering nature of E-mails to build the evolving rules. Hence, in contrast to the method in other approaches, the feature behavior in the proposed system is not a fixed factor. Evolving the rules can include a new value of the features vector related to new spam E-mail attacks. Therefore, the system can work with this new attack without prior knowledge of the features vector value itself this is because the second stage of the proposed system implemented by ECM and ECMc algorithms is designed to work with unsupervised learning approach[17,28].

The most effective 56 extracted features are adopted and extracted by analyzing data of parts C and D, which represent the content part of the message and are often used as they represent the best side to work with classifier algorithms for detecting spam attacks before these reach the victim account, as shown in Figure 2. A common

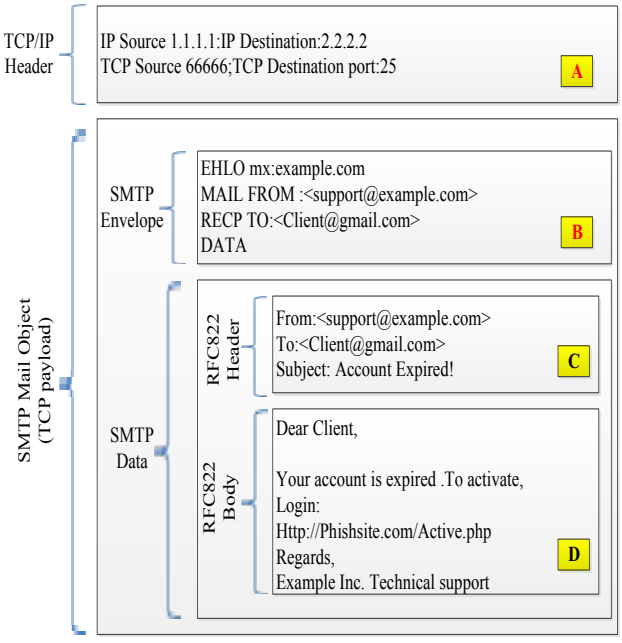

Figure 2. Overview of e-mail data parts[9].

approach for extracting features obtained in data parts A and B by the use of blacklists is given. However, blacklists are purposely avoided as these perform poorly against zero-day spam attacks[9].

Where, x is data part ID, Simple Mail Transfer Protocol (SMTP) is an Internet standard for (E-mail) transmission across Internet Protocol (IP), RFC822 is message sample format accepted by archive mail server.

The features used in the proposed system selected by studying the newest 56 attributes in an area of spam E-mail detection, based on information gain algorithm these features collected from many publication included[9,11,29–31]. Details of selected features shown in Table 1[32], shows that these features represented by parts C and D of the message[9]. The idea behind selecting sub-features from the sets of features (as against using all of them) is that ineffective features will increase the time and space complexity of classifiers, which affect the accuracy of the classifier[9].

Table 1 shows five main of features included more than 54 sub-features from spam assassin tool that represent all parts of the content message. In the next stage, we will discuss the rule creation.

3.2 Rule-Creation Based on Fuzzy Inference System

In this phase, an evolving rule is generated for the classification. Three parts are suggested to build an unlimited "life-long" training system that will enhance the rules while the system is working in online mode. The first part depends on the DENFIS-online mode. The second depends on the DyNFIS-offline mode to enhance the rules. The profile management system is used to arrange the relationship between the first and second parts. Rules for each part in the proposed system are explained below:

3.2.1 Rule-Creation Based on DENFIS-Online Mode

The proposed system uses DENFIS in the online mode. DENFIS is a dynamic inference system that can create or upgrade a fuzzy rule as the system is in running order. The output depends on the most active fuzzy rules at any given time. DENFIS depends on the fuzzy rule set that is chosen automatically (the system dynamically detects the border of input using Takagi-Sugeno fuzzy inference engine). A robust feature pertaining to DENFIS is that capability of generating new rules before or during the learning process. Moreover, DENFIS can extract rules during or after the learning process.

The most important part of the DENFIS algorithm is ECM, as the antecedent of the fuzzy rules based on the cluster centers. A fuzzy inference system is activated when one of the clusters is fully optimized[17].

DENFIS uses the Takagi-Sugeno fuzzy inference engine with triangular membership functions (MF) made up of m fuzzy rules[14]. However, the new version of the DENFIS-online mode works based on the Gaussian MF, and the rules generated from the new version are more accurate in the classification process[33]. By adopting the new version of the DENFIS-online mode based

Table 1. Spam E-mail features used in SDENFS

No	IG	Features	Meaning
1	0.897882	externals score	SpamAssassin4 (SA) features that returns the scoreof a given email as returned by SA (i.e. A contentious feature).
2	0.812893	externals binary	SpamAssassin4 (SA) and a binary feature that returns1 if a given email is labeled as a spam message by SA,and 0 if otherwise (i.e. a binary feature).
3	0.726431	url num link	A feature that returns the total number of links found in the body of a given email.
4	0.692546	Body html	A binary feature that returns 1 if the email message has HTML content, and 0 otherwise
5	0.637373	url num periods	A feature that returns the total number of periods in the body of a given email.

on Gaussian function because, an extension can be made to the DyNFIS-offline mode because the Gaussian MF coverage of the problem space expands a long way as the degree of membership decreases steadily[34].

Gaussian function as a bell curve from the peak at different rates depends on the parameter of the function, and does not reach zero. Basically, there is considerable noise in environmental datasets. Such noise is often non-Gaussian, and handling noisy data may not be suitable via common techniques. Gaussian function is designed to filter and remove the noise[35], thereby increasing the level of accuracy from the rule generated. In DENFIS, the final shapes of rules are defined by the Gaussian-type MF, which has two parameters, as expressed in Equation 1. The fuzzy rule sets defined by the following Gaussian-type MF is as follows:

$$\textbf{Gaussian MF} = \alpha Exp\left(\frac{-(X-c)^2}{2\sigma^2}\right) \qquad (1)$$

Where, x is the input vector, c is the center of the Gaussian function, α is the height of the curve peak, and σ is the width (cluster radius) of the Gaussian function, listed in the order of vector [c, σ]. As shown in Figure 3, part of ECM clusters will generate the rule based on the Gaussian function implementation in 2D, which also includes visualization between the cluster and Gaussian MFs.

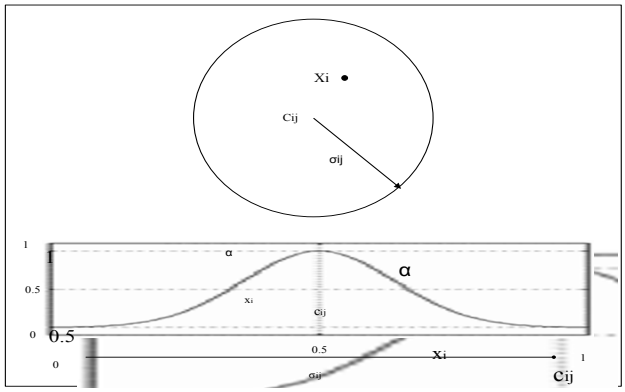

Figure 3. Gaussian MF implementation in 2D.

As data enter the system, there is room for generating a new fuzzy rule together with updating several rules through a new input vector of data.

A new fuzzy rule is created if a new cluster is generated in ECM. Otherwise, one or more fuzzy rules are updated.

When the system is given an input-output pair (Xi,Yi), DENFIS is used to distinguish a spam E-mail from a legitimate E-mail. However, in enhancing the rules generated by DENFIS, the DyNFIS is suggested to be in the offline mode while the system is working in the online mode. Through this procedure, the rules based on ECMc are enhanced to make the generated rules more fitting and accurate for the classification input samples without stopping the system. This process should be done as DENFIS and DyNFIS have the same format or rules based on the Gaussian MF rules, with difference only in the level of accuracy for the rules generated from both algorithms.

3.2.2 *Enhancement Rules Based on DyNFIS-Offline Mode*

The proposed system uses the DyNFIS-based on ECMc to enhance the rule in the offline mode while the system is working in the online mode, build a life-long learning system, and enhance the performance of the learning algorithm. Data derived from the offline part of ECM are fed to DyNFIS. However, DyNFIS works based on the Gaussian MF in the offline mode as the latter is more accurate and more suitable for real-world applications. DyNFIS will allow the antecedents and consequences to be optimized using BP algorithm and then minimize the error in the active rules based on minimizing the objective function[34].

3.2.2.1 *DyNFIS Algorithm Description*

We outlined The DyNFIS offline learning process as follows:

1. Re-cluster the input vectors to detect cluster centers using offline ECMc.
2. For each cluster system will create a fuzzy rule.
3. Antecedent fuzzy rule is created based on each cluster center position.
4. The consequence of the fuzzy rule is a linear function trained by the vector of that inputs.
5. System will extract the output from the most m activated rule for all training input vectors, then adjust the fuzzy membership function and the consequence of the most activated rule, based on BP algorithm to reduce the level of error.
6. Until the desired accuracy system will repeat step three for many epochs or is obtained[34].

The BP algorithm is used to optimize parameters α m, σ, and β to decrease the error rate, with the full equation

discussed in[34], whereas the objective function defined by the summation of the distance among all features vectors, the cluster center, and the system will become more effective if the objective function is decreased, and vice versa. The objective function J_j is calculated according to Equation 3.5[36].

$$F_f = \sum_{f=1}^{n} \left(\sum_{k,X_{K=C_f}} \left[|X]_{K-C_{Cf}} \right] \right| \quad (3.5)$$

Within cluster C_f, for each f=1, 2, 3…n., k is an input number.

The proposed system takes advantage of DENFIS and DyNFIS to minimize the error in the active rules. It is most suited to a higher level of noise data and has the ability to detect and predict zero-day spam E-mails rapidly and with high accuracy in the classification and prediction processes. However, the profile management system is suggested to arrange the relationship between the two algorithms (DENFIS and DyNFIS) as follows:

3.3 Profile Management System

Profile management systems plays two main rules in second and third phase respectively in SDENFS. The first contribution of Profile management system occour in the second level which implement by capturing input samples form ECM-Online then enhance the place of input vector by enhance the postion of clusters centers based on ECMc-offline as shown in Figure 4 and Figure 5 respectively. This process to optimize the clusters based on decrease the objective function, this condition happens by default for every 1000 samples of E-mails because the lenght of time the mail server will receive a new message is not known. Therefore, this process is controlled based

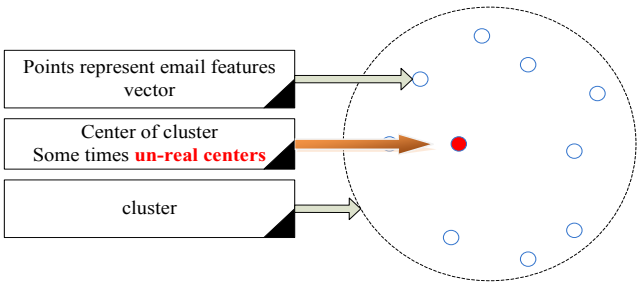

Figure 4. The basic cluster shape in evolving clustering method (ECM).

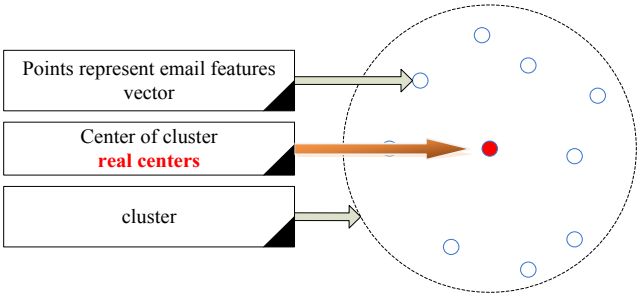

Figure 5. The basic cluster shape in evolving clustering method with constrain optimization (ECMc).

on the number of E-mails and not based on Time of pick up E-mail.

ECM is a strong algorithm design to work with noise data[14]. Therefore, it is used to allow SDENFS to deal with noisy datasets and this will support to work with unknown attack. ECM with ECMc works in DENFIS and DyNFIS, respectively, to build the base of evolving rules. The cluster center need not be in the center of gravity of the "real center" in ECM online, this will make the rule generated by ECM without full accurate in classification process[16]. On the other hand, the offline version of ECM is proposed to deal with such problem of cluster centers whenever they are not positioned at the center of gravity, which mean it can generate more accurate rules for classification process. Thus, a dynamic system between ECM and ECMc in SDENFS was suggested. ECMc optimizes the final result in offline mode by capturing stream of E-mails as vectors of features while the system is working in the online mode.

Below is explanation of the algorithm steps of combination between DENFIS and DyNFIS based on ECM and ECMc respectively. With note that steps from one to seven related to the Online mode of ECM and the steps from eight to twelve related to the offline mode of ECMc. Figure 6 shows the flowchart of ECM with its extension ECMc as it works in SDENFS.

The ECM algorithm inside DENFIS algorithm is described in algorithm 1-Figure 7 as follow:

3.3.1 Algorithm 1: (ECM): One-Pass and Fast Algorithm

Algorithm 1

Input: Input vectors $x_e \in X$

Output: n clusters' centers Cc f and corresponding radius R_{uf}

Where f=1, 2, 3…, n

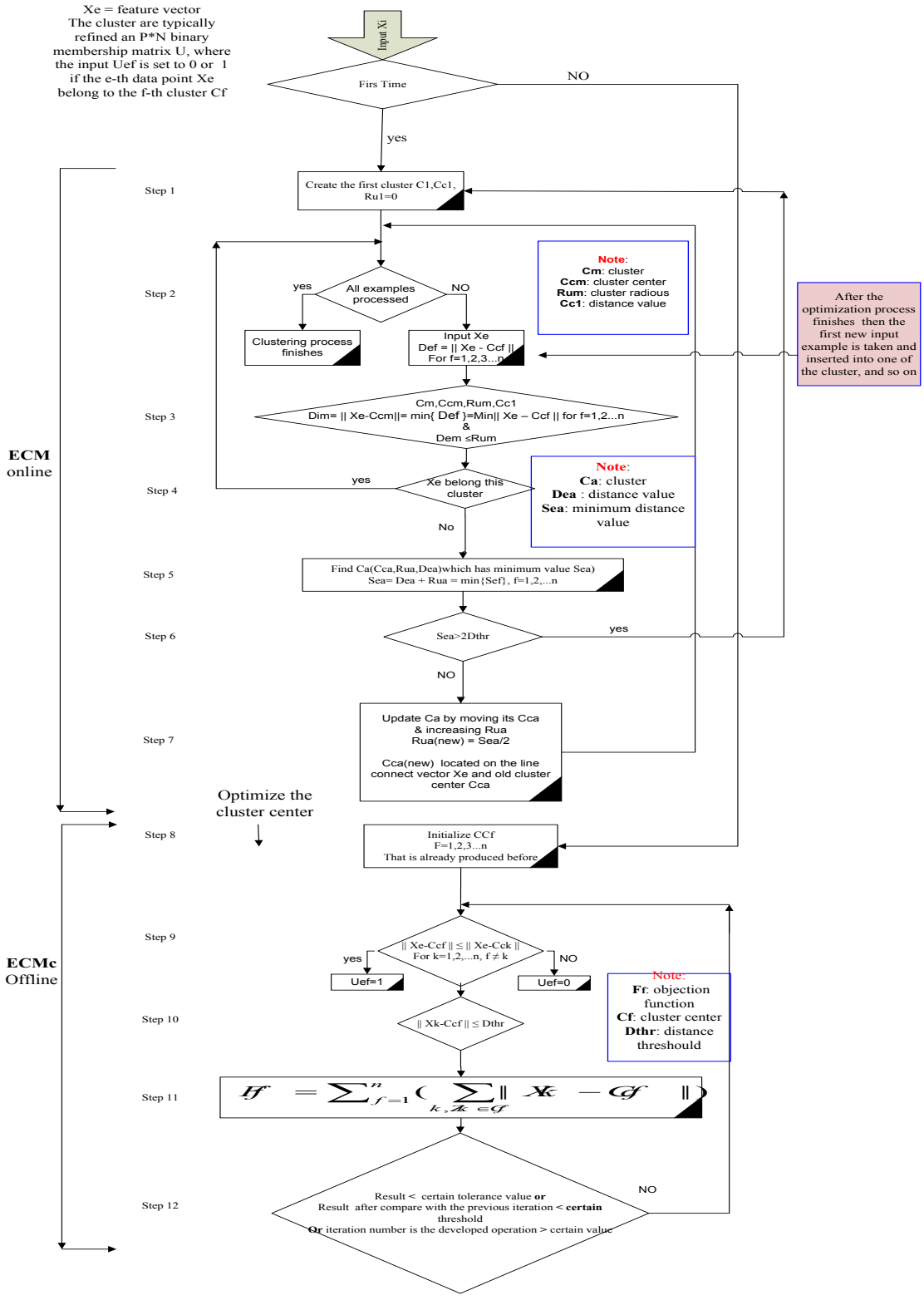

Figure 6. ECM and its extension ECMc to connect between DENFIS and DyNFIS.

Step 1: Create the initial cluster C1 by suggesting the place of the first learning data sample as a first cluster center C_{c1}, and the cluster Radius R_{u1}=0.

Step 2: Then,

If $\forall xe \in C = \{C_1, C_2, ..., C_n\}$ **Then**

Stop the algorithm.

Else

Step 3: distance Calculated between the progress training input vector X_e and the cluster center C_{cf}.

as follows $D_{ef} = ||x_e - C_{cf}||$, f=1,2,…,n.

End if

Step 4: $\exists ||X_e - C_{cm}|| \leq$ Rum **Then**

X_e is assumed to belong in cluster C_m and no new cluster is created and no cluster is updated.

Go back to step 2.

End if

Step 5: Discover a cluster C_a from all created cluster centers and $S_{ef} = D_{ef} + R_{uf}$, j=1,2,…n, and select the cluster centers C_{ca} with the smallest

$S_{ea} = D_{ea} + R_{ua} = \min \{S_{ef}\}$, f=1, 2,..,, n.

Step 6: If the shortest distance $S_{ea} > 2 \times$ Distance threshold (D_{thr}) **Then**

Sample X_e will not related to any existing cluster, and a new cluster is created as described in step1

Go back to step 2.

End if

Step 7: $S_{ea} \leq 2 \times D_{thr}$ **Then**

Cluster C_a is updated by moving C_ca and enlarging the cluster

Radius $R_{ua} = S_{ea} / 2$ and the new cluster center C_{ca} is set as follows:

$$\mathbf{Cca}^{new} = X_e - \left((C_{ca} - X_e) \times \frac{\left(\frac{S_{ea}}{2}\right)}{D_{ea}} \right)$$

End if

End while

3.3.2 *Algorithm 2: Steps of Combination between ECM and ECMc*

Algorithm 2:

Step 8: If entering input for the first time, go to step 1 to create a new cluster. Else re-create the cluster center C_{cf}, f = 1, 2, 3…n that is already created before.

Step 9: the membership matrix U Determined,where the element U_{ef} is 1 if the e^{th} data point X_e belongs to C_f, and 0 otherwise. Once the cluster centers C_{cf} are defined, the values U_{ef} are derived as

IF $|| X_e - C_{cf} || \leq || X_e - C_{ck} ||$, For k=1,2,...n, f ≠ k;

Then U_{ef}=1, else U_{ef}=0

Step 10: Use the condition of minimization method to update the cluster centers, as in the following equation.

$|| X_k - C_{cf} || \leq D_{thr}$

Step 11: Optimize the cluster. Calculate the objective function F_e according to the following equation.

$$F_f = \sum_{f=1}^{n} \left(\sum_{k, X_k \in C_f} | X_k - C_f | \right)$$

Within cluster C_f for each f=1,2,3…n.

Step 12: If the result is shorter than a border tolerance value, or the result after comparing with the earlier iteration is less than the threshold value, or the iteration number for the optimization is greater than a border value, then go to step 1, else go to step 9.

The second contribution of the profile management system occurs in the fourth level, which is implemented by two steps:

1. Capture the rule profiles created in DyNFIS-offline mode.
2. Insert capturing rule profile to DENFIS-online mode while the system is working.

This process appears as a parallel system to enhance the repository of rules based on ECMc. The profile management system will pass the updated rule to DENFIS, which can automatically adapt the rules without duplication and select the best rule in the classification process. The profile management system has the same format of rules, which rely on the same type of fuzzy rule and Gaussian MF.

4. Experimental Design and Test Results

This part will focus on describing the experiments which are designed to analyze the variations in online mode activity and to implement the proposed E-mail detection solution using various methods. The main intention behind these experiments is to find a system which is capable of accurately detect spam E-mail based on two dataset from three resources as follow.

Table 2. Basic dataset statistics

Groups of datasets	Dataset	Size	Start	End
First dataset Public and (benchmark)[26,37]	Legitimate (L)	4,150	Jan 2002	Oct 2002
	spam (P)	4,116	Nov 2005	Aug 2006

4.1 Datasets Used

Two datasets are used in assessing the proposed system, as shown in Table 2.

Table 2 shows basic dataset statistics, a publicly available source of spam and legitimate E-mails, used by many authors in this area. This dataset has been used in a number of recent studies such as[9,21,29,38] publicly available spam and legitimate E-mail sources are discussed below.

The spam dataset is a collection of 4,116 spam messages received from November 2005 to August 2006. The files are as distributed by the Monkey Website[37]. The legitimate dataset is composed of 4,150 E-mails from the Spam Assassin project distributed by[26]. We randomly selected 6612 samples then training 5785 samples and testing 827 based on 8 cross validation and linear normalization. A full description is provided in the experiment section.

4.2 Performance Evaluation Mechanism

Eight measurements are used to evaluate the quality of SDENFS commonly used in other studies to clarify the experiments. These measurements depend on True Positive (TP) referring to the number of spam E-mails correctly classified as spam; True Negative (TN) is the number of legitimate E-mails correctly classified as legitimate; False Positive (FP) is the number of legitimate E-mails wrongly classified as legitimate; and false negative (FN) is the number of spam E-mails wrongly classified as legitimate. Table 3 presents measurements that compare the proposed system (SDENFS) with the most powerful algorithms currently used to detect spam E-mail attack. However, we will focus on two measurements: accuracy and F-measure, as they represent most of the measurements. Full details of each experiment are shown below.

4.3 Training and Testing

TThe experiments are based on tenfold cross-validations and use random method with min-max (linear normalization) for all input, to confine the input within the range of [0,1] and preserve all relationships of the data values. Cross-validation is the process of estimating the rate of errors in an efficient and unbiased manner21. In each run of the experiments, nearly 80% of the dataset is trained, whereas 20% of the dataset is tested. The procedure is as follows: the datasets are classified into k sub-samples (in the experiments, k = 10). Two sub-samples are chosen as testing data and the remaining k-2 sub-samples as training data. The process is repeated k times, in which each of the k sub-samples is used exactly once as the testing data. Finally, an average is worked out for all results with unified evaluation to clarify the number of experiments in a simple method.

Table 3. Measures used for classification of spam and legitimate E-mails[39,40]

Measure	Formula	Meaning
Accuracy	$= \frac{\|TP\| + \|TN\|}{\|TP\| + \|TN\| + \|FP\| + \|FN\|}$	The percentage of predictions that is correct.
F-Measure	$= 2.\frac{precision.recall}{precision + recall}$	It is a measure of a test's accuracy. bothprecision and the recall of the test are utilized to compute the score

4.4 Performance Evaluation

This part discusses four main experiments, including many sub-experiments, to show the objective and contribution of our study based on performance results generated from each experiment and compared with other approaches.

4.4.1 Performance of SDENFS

The main objective of this experiment is to show the performance of SDENFS which depends on the distance threshold (D_{thr}) in ECM; many random D_{thr} values are selected to find the optimal D_{thr}. Five D_{thr} are selected within the range of [0.1, 0.5], as they show the best results in the proposed system.

Table 4 shows the parameters of this experiment, and Table 5 shows the average results of many sub-experiments distributed. Each sub-experiment depends on training 80% and testing 20% of data, and this design based on a eight fold cross-validation.

Table 4 shows that the prediction level of SDENFS generally yields highly satisfactory results. The optimal threshold is D_{thr} = 0.2, yielding the best result compared with all other distance thresholds (D_{thr}). Overall accuracy and F-Measure is shown to reach up to 99%, thus, this threshold is used in Experiments and the result shown in Table 5 below and Table 6 as shown below.

Table 6 shows the results of comparison of SDENFS with other approaches. The performance metrics which are compared accuracy (for binary classification and 2-class classification). Table 6 shows the comparison of SDENFS with other methods and the study on the accuracy metric shows that SDENFS gave a better accuracy level of classification (99%) in comparison with the best result obtained by SVM and NNet (MLP) algorithms. Similarly, Finally SDENFS indicates an improvement based on DENFIS and DyNFIS in detecting unknown spam E-mail attacks with capacity to work with noise data, and the ability of life-long learning using classification data in the online mode while it enhanced by offline mode.

5. Conclusions and Future Work

The proposed system improved the level of performance in detecting and predicting unknown spam E-mails. SDENFS distinguished spam E-mails from legitimate E-mails in an online mode based on new rules, classes, or features to enhance learning using ECOS. Therefore, the

Table 4. SDENFS prediction average based on distance threshold

D_{thr}	SDENFS				
	Training phase			Testing phase	
	NDEI	RMSE	Rule Number	NDEI	RMSE
0.1	0.2	0.1	34	0.18	0.09
0.2	0.2	0.1	16	0.17	0.08
0.3	0.2	0.1	9	0.17	0.09
0.4	0.2	0.1	6	0.18	0.09
0.5	0.2	0.1	3	0.2	0.1

Table 5. SDENFS performance average based on D_{thr} =0.2

TP	TN	FP	FN	Accuracy	F-measure
99%	99%	0%	0%	99%	99%

Table 6. Performance comparison-based on (Accuracy)

Classifiers	Accuracy	Can handle unlabeled "Zero-day" attack?	Online/Offline
SDENFS	99%	yes	Online enhanced by Offline
SVM	99%	No	Online or offline
NNet (MLP)	99%	No	Online or offline
MLR	98%	No	
Bayesian (Naïve-base)	98%	No	Online-only
Random forest	98%	No	Online or offline

present work is an important step in the use of ECOS for spam E-mail detection.

A new technique was used for the extraction of features based on the assumption that all features have a binary or continuous features. The proposed approach used a new incremental clustering algorithm tailored for this purpose, which depends on the Maximum Distance (MaxDist) between the input data and the cluster center for classification and for developing new rules in DENFIS *online mode* and enhancing the rules based on *DyNFIS offline mode while the system is working*. Our system depended on the Takagi-Sugeno fuzzy model generation and Gaussian membership function.

The experiments proved that the proposed system has better performance, including enhancing the overall accuracy and others measurement compared with other learning algorithms that used existing solution. Therefore, the proposed approach has a great potential for real-world applications. In future, suggestion to extend the functionality of the proposed system to become full dynamic system based on dynamic Distance Threshold (D_{thr}), and without changing the distance threshold in static way as we done in current system, for building a system capable of high-speed work and has a high performance for real-world implementation.

6. Acknowledgement

This work was supported by Al-Balqa Applied University, Al-Huson University College, Department of Information Technology, 50, Irbid, Jordan.

7. References

1. MAAWG. Messaging Anti-Abuse Working Group (MAAWG) E-mail Metrics Program; 2011.
2. Almomani A. An online model on evolving phishing E-mail detection and classification method. J Appl Sci. 2011; 11(11):3301–7.
3. Gooking D. The real difference between integers and floating-point values. 2013. Available from: http://www.dummies.com/how-to/content/the-real-difference-between-integers-and-floatingp.html
4. Arab weeks. Saudi Arabia's in-bound spam rates highest worldwide at 82.7 percent. 7 Jun 2013; Available from: http://arabweeks.net/en/?p=239
5. Gupta N, Saikia A. Web application firewall. Indian Institute of Technology, Kanpur; 2007.
6. Cook DL. Phishwish: A simple and stateless phishing filter. Secur Comm Network. 2009; 2(1):29–43.
7. Dunlop M. GoldPhish: Using images for content-based phishing analysis. 5th International Conference on Internet Monitoring and Protection, ICIMP; 2010. p. 123–8.
8. Bimal Parmar F. Protecting against spear-phishing. Computer Fraud and Security. 2012:8–11.
9. Khonji M. Enhancing phishing E-mail classifiers: A lexical url analysis approach. International Journal for Information Security Research (IJISR). 2012; 2.
10. Kasabov N. Evolving connectionist systems with evolutionary self-optimisation. Do Smart Adaptive Systems Exist? 2005; 181–202.
11. Bergholz A. New filtering approaches for phishing mail. Journal of Computer Security. 2010; 18:7–35.
12. Kasabov ZSHCN, Song Q, Greer D. Evolving connectionist systems with evolutionary self-optimisation. 2005; 173.

13. Almomani A. Phishing dynamic evolving neural fuzzy framework for online detection zero-day phishing E-mail. Indian Journal of Science and Technology. 2013; 6(1):2–126.
14. Kasabov N, Song Q. DENFIS: Dynamic Evolving Neural-Fuzzy Inference System and its application for time-series prediction. IEEE Trans Fuzzy Syst. 2002; 10(2):144–154.
15. Snjezana Soltic IP. Bulletin of Applied Computing and Information Technology. Journal of Applied Computing and Information Technology. 2006; 4:1–8.
16. Hwang YC, Song Q. Dynamic neural fuzzy inference system. Advances in Neuro-Information Processing. 2009; 5506:1245–50.
17. Almomani A. An enhanced online phishing E-mail detection framework based on evolving connectionist system. International Journal of Innovative Computing, Information and Control (IJICIC). 2013 Mar. p. 9.
18. Silver DL. Unsupervised and transfer learning workshop. ICML; 2011.
19. Lam HY, Yeung DY. A learning approach to spam detection based on social networks. Hong Kong University of Science and Technology; 2007.
20. Lee SM. Cost-sensitive spam detection using parameters optimization and feature selection. J Univers Comput Sci. 2011; 17(6):944–60.
21. Abu-Nimeh S. A comparison of machine learning techniques for phishing detection. Proceedings of the eCrime Researchers Summit; 2007. p. 60–9.
22. Ammar Almomani TCW, Manasrah A, Altaher A, Almomani E, Alnajjar A, Sureswaran R. A survey of learning-based techniques of phishing E-mail filtering. International Journal of Digital Content Technology and its Application, JDCTA. 2012; 6(18):119–29.
23. Shou-Sheng LHFL, Xue-Ren Z. Clustering-based Improved K-means text feature selection. Computer Science. 2011; 1:048.
24. Ram Basnet SM, Sung AH. Detection of phishing attacks: A machine learning approach. Soft Computing Applications in Industry. 2008; 226:373–83.
25. Kasabov N. Evolving Connectionist System (ECOS). Computational Neurogenetic Modeling. Springer; 2007.
26. Corpus P. Spam assassin. 2010 Jul 22. Available from: http://sourceforge.net/projects/sawin32/
27. Song Q, Kasabov N. ECM-A novel on-line, evolving clustering method and its applications. Proceedings of the 5th Biannual Conference on Artificial Neural Networks and Expert Systems (ANNES 2001); 2001. p. 87–92.
28. Almomani A. A survey of phishing E-mail filtering techniques. IEEE Communications Surveys and Tutorials. 2013; 15:2070–90.
29. Fette I. Learning to detect phishing E-mails. Proceedings of the 16th International World Wide Web Conference (WWW 2007); 2007 May. p. 649–56.
30. Gansterer WN. E-mail classification for phishing defense. Proceedings of the 31st European Conference on IR Research on Advances in Information Retrieval; 2009.
31. Toolan F, Carthy J. Feature selection for Spam and Phishing detection. eCrime Researchers Summit (eCrime); 2010. p. 1–12.
32. Khonji M. A study of feature subset evaluators and feature subset searching methods for phishing classification. Proceedings of the 8th Annual Collaboration, Electronic Messaging, Anti-Abuse and Spam Conference; 2011.
33. Satchidanandan Dehuri CM, Ghosh A, Mall R. A comparative study of clustering algorithms. Information Technology Journal. 2006; 3:551–9.
34. Hwang YC, Song Q. In: Koppen M, editor. Dynamic Neural fuzzy inference system advances in neuro-information processing. Berlin/Heidelberg: Springer; 2009. p. 1245–50.
35. Hwang YCP. Local and personalized models for prediction, classification and knowledge discovery on real world data modelling problems [PhD thesis]. School of Computing and Mathematical Sciences, AUT University; 2009.
36. Ma L. Establishing phishing provenance using orthographic features; 2009. p. 1–10.
37. Nazario J. Phishing corpus; 22 Jul 2006. Available from: http://monkey.org/ jose/wiki/doku.php?id=PhishingCorpus
38. Khonji M. A brief description of 47 Phishing Classification; 2011.
39. Venkatesh Ramanathan HW. phishGILLNET - phishing detection methodologyusing probabilistic latent semantic analysis, AdaBoost, and co-training. EURASIP Journal on Information Security; 2012.
40. Kim J, Kasabov N. HyFIS: Adaptive neuro-fuzzy inference systems and their application to nonlinear dynamical systems. Neural Networks. 1999; 12:1301–19.