Hindawi Publishing Corporation
Applied Computational Intelligence and Soft Computing
Volume 2011, Article ID 416308, 8 pages
doi:10.1155/2011/416308

Research Article

Classification of Textual E-Mail Spam Using Data Mining Techniques

Rasim M. Alguliev, Ramiz M. Aliguliyev, and Saadat A. Nazirova

Institute of Information Technology of Azerbaijan National Academy of Sciences, 9 F. Agayev Street, Baku 1141, Azerbaijan

Correspondence should be addressed to Saadat A. Nazirova, sbunyadova@gmail.com

Received 19 May 2011; Revised 23 August 2011; Accepted 5 September 2011

Academic Editor: Sebastian Ventura

A new method for clustering of spam messages collected in bases of antispam system is offered. The genetic algorithm is developed for solving clustering problems. The objective function is a maximization of similarity between messages in clusters, which is defined by k-nearest neighbor algorithm. Application of genetic algorithm for solving constrained problems faces the problem of constant support of chromosomes which reduces convergence process. Therefore, for acceleration of convergence of genetic algorithm, a penalty function that prevents occurrence of infeasible chromosomes at ranging of values of function of fitness is used. After classification, knowledge extraction is applied in order to get information about classes. Multidocument summarization method is used to get the information portrait of each cluster of spam messages. Classifying and parametrizing spam templates, it will be also possible to define the thematic dependence from geographical dependence (e.g., what subjects prevail in spam messages sent from certain countries). Thus, the offered system will be capable to reveal purposeful information attacks if those occur. Analyzing origins of the spam messages from collection, it is possible to define and solve the organized social networks of spammers.

1. Introduction

E-mail is an effective, fast and cheap communication way. Therefore spammers prefer to send spam through such kind of communication. Nowadays almost every second user has an E-mail, and consequently they are faced with spam problem. E-mail Spam is nonrequested information sent to the E-mail boxes. Spam is a big problem both for users and for ISPs. The causes are growth of value of electronic communications on the one hand and improvement of spam sending technology on the other hand. By spam reports of Symantec in 2010, the average global spam rate for the year was 89.1%, with an increase of 1.4% compared with 2009. The proportion of spam sent from botnets was much higher for 2010, accounting for approximately 88.2% of all spam. Despite many attempts to disrupt botnet activities throughout 2010, by the end of the year the total number of active bots returned to roughly the same number as at the end of 2009, with approximately five million spam-sending botnets in use worldwide [1].

Spam messages cause lower productivity; occupy space in mail boxes; extend viruses, trojans, and materials containing potentially harmful information for a certain category of users; destroy stability of mail servers, and as a result users spend a lot of time for sorting incoming mail and deleting undesirable correspondence. According to a report from Ferris Research, the global sum of losses from spam made about 130 billion dollars, and in the USA, 42 billion in 2009 [2]. Besides expenses for acquisition, installation, and service of protective means, users are compelled to defray the additional expenses connected with an overload of the post traffic, failures of servers, and productivity loss. So we can do such conclusion: spam is not only an irritating factor, but also a direct threat to the business. Considering the stunning quantity of spam messages coming to E-mail boxes, it is possible to assume that spammers do not operate alone; it is global, organized, creating the virtual social networks. They attack mails of users, whole corporations, and even states.

Every day E-mail users receive hundreds of spam messages with a new content, from new addresses which are automatically generated by robot software. To filter spam with traditional methods as black-white lists (domains, IP addresses, mailing addresses) is almost impossible. Application of text mining methods to an E-mail can raise efficiency of a filtration of spam. Also classifying spam messages will be possible to establish thematic dependence from geographical (e.g., what subjects prevail in the spam-messages sent from the certain countries). Methods of text clustering and classifying were successfully applied to spam problem from last decade. A filtration of E-mail onto legitimate and spam with the help of clustering analysis is considered in the papers [3–9].

This paper focuses on the classification of textual spam E-mails using data mining techniques. Our purpose is not only to filter messages into spam and not spam, but still to divide spam messages into thematically similar groups and to analyze them, in order to define the social networks of spammers [10].

The rest of the paper is organized as follows. Section 2 presents related works. Section 3 describes the representation of spam messages in databases for analyzing them and defines similarity measure between spam messages. The proposed clustering method is presented in Section 4. A genetic algorithm for solving clustering problem is offered in Section 5. Classification of the collected spam messages using the *k*NN method is described in Section 6. To receive the information about clusters, the document summarization method is applied in Section 7. Conclusion and future work are given in Section 8.

2. Related Works

Spam messages are one of weapons of information war. Since 2003 in scientific literature the notions spam and war appear in one context [11, 12]. But problems of spammers' social networks are considered in articles begining from 2009. In [13], the clustering of spammers considering them in groups is offered. In [14, 15], spectral clustering method is applied to the set of spam messages collected by Project Honey Pot for defining and tracing of social networks of spammers. They represent a social network of spammers as a graph, nods of which correspond to spammers, and a corner between two junctions of graph as social relations between spammers.

In this paper, the document clustering method is applied for clustering and analyses of spam messages. In our case, the text documents are textual E-mails. In spite of the fact that there are many approaches to representation of text documents, the most widespread of them is the vector model. The vector model for representation of texts has been offered in Salton's works [16, 17]. In the elementary case, the vector model assumes comparison to each document of a frequency spectrum of words and accordingly a vector in lexical space. In more advanced vector models, the dimension of space is reduced by rejection of the most widespread or infrequently words, increasing thereby percent of the importance of the basic words. The main advantage of vector model is the possibility of ranging of documents according to similarity in vector space [18].

Clustering is one of the most useful approaches in data mining for detection of natural groups in a data set. For the solution of clustering problem the traditional algorithms, such as *k*-means algorithm [19, 20], hierarchical clustering, differential evaluation algorithm, particle swarm optimization algorithm, artificial bee colony optimization, ant colony algorithm, and neural network algorithm GEM (Gaussian expectation-maximization), are usually used [21–26]. The up-to-date survey of evolutionary algorithms for clustering, especially the partition algorithms, are described in detail in [27]. The comparison of advanced topics like multiobjective and ensemble-based evolutionary clustering; and the overlapping clustering as soft, fuzzy clustering are also mentioned in that paper. Each of the surveyed algorithms is described with respect to fixed or variable number of clusters; cluster-oriented or nonoriented operators; context-sensitive or context-insensitive operators; guided or unguided operators; binary, integer, or real encodings; and centroid-based, medoid-based, label-based, tree-based, or graph-based representations.

Clustering of spam messages means automatic grouping of thematically close spam messages. In case of information streams as E-mails, this problem becomes complicated necessity to carry out this process in real-time mode. There are some complications connected with plurality of a choice of algorithms for clustering of spam messages. Different methodologies use different similarity algorithms for electronic documents in case of a considerable quantity of signs. As soon as classes are defined by clustering method, there is a necessity of their support as spam constantly changes, and spam messages collection replenishes. In considered work, the new algorithm for definition of criterion function of spam messages clustering problem is offered. The clustering problem itself is solved by genetic algorithm [28]. Genetic algorithms are the subjects of many scientific works. For example, in [29] a survey of genetic algorithms designed for clustering ensembles, the genotypes, fitness functions, and genetic operations is presented and concludes that using genetic algorithms in clustering ensemble improves the clustering accuracy.

In this work, for the classification of spam messages the *k*-nearest neighbor method is applied, and for the determination subjects of spam messages, clusters will be applied to a multidocument summarization method offered in papers [30–32].

3. Problem Statement

Assume there is a collection of spam messages collected on servers of hierarchical system of spam filtration, described in paper [33]. Before applying any clustering method for clustering these spam messages and analyzing them, one should specify the input data. There are some approaches to information representation in databases for maintenance of the subsequent analysis of this information. We will consider the most popular approaches to representation of the text information dynamically arriving in databases of information systems. Let us consider the collection of spam messages in vector space. Assume $S = \{s_1, \ldots, s_n\}$ is a collection of spam messages, and $T = \{t_1, \ldots, t_m\}$ is a set of terms (spam keywords) in spam messages collection. In vector model, any message can be represented as a point in m dimensional space, where m is the number of terms. Each spam message, identified with the weighted vector:

$$s_i = [w_{i1}, \ldots, w_{im}], \tag{1}$$

where w_{ij} is a weight of term t_j in spam message i, $i = 1, \ldots, n$, $j = 1, \ldots, m$, and n is the number of spam messages in collection. There are different ways to calculate the weights of terms [34]. Let us consider the most popular TF∗IDF weighting (term frequency—inverse document frequency) method for determination the weight of term w_{ij}. This method considers frequency of occurrence of a term in all messages of sample and its discriminating ability. By TF∗IDF weighting scheme the weight of a term t_j in the message s_i is calculated by the following formula:

$$w_{ij} = n_{ij} \log\left(\frac{n}{n_j}\right), \tag{2}$$

where n_{ij} is a frequency of appearance of term t_j in spam message s_i and n_j is the number of spam messages containing the term t_j.

After representation of spam messages, one should define similarity measure between them. Similarity measure can be defined by one of the metrics: cosine measure, Euclidian distance, and Jaccard measure. In this paper, the similarity measure between spam messages s_i and s_j will be defined by cosine measure. Cosine measure defines similarity by calculation of a cosine of the angle between vectors s_i and s_j [31]:

$$\operatorname{sim}\left(s_i, s_j\right) = \frac{\sum_{l=1}^{m} w_{il} w_{jl}}{\sqrt{\sum_{l=1}^{m} w_{il}^2 \cdot \sum_{l=1}^{m} w_{jl}^2}}, \quad i, j = 1, \ldots, n. \tag{3}$$

Considering spam messages in vector model and choosing metrics for similarity measure between spam messages, the offered algorithm of content analyses can be applied. The offered algorithm consists of the following steps:

(1) clustering method,

(2) algorithm for solving the clustering problem,

(3) classification of collected spam messages,

(4) knowledge extraction from classes.

Below the detailed description of each step is given.

4. Clustering Method

Data clustering is the process of dividing data elements into classes or clusters so that items in the same class are as similar as possible, and items in different classes are as dissimilar as possible. Depending on the nature of the data and the purpose for which clustering is being used, different measures of similarity may be used to place items into classes, where the similarity measure controls how the clusters are formed. Two types of clustering are defined: hard or fuzzy clustering. In fuzzy clustering (also referred to as soft clustering), data elements can belong to more than one cluster, and associated with each element is a set of membership levels. In hard clustering, data is divided into distinct clusters, where each data element belongs to exactly one cluster [35].

The goal of clustering is to split the set $S = \{s_1, \ldots, s_m\}$ into nonoverlapping clusters $C = \{C_1, C_2, \ldots, C_q\}$, $q > 1$, with the purpose of maintenance of the maximum similarity between messages of one cluster corresponding to certain semantic subjects, and the maximum distinction between clusters. That is, the following conditions of hard clustering should take place

$$\begin{gathered} C_p \neq \varnothing \quad \text{for } p = 1, \ldots, q, \\ C_p \cap C_z = \varnothing \quad \text{for } p \neq z,\ p, z = 1, \ldots, q, \\ \bigcup_{p=1}^{k} C_p = S. \end{gathered} \tag{4}$$

Let us introduce the following designations:

$$O_{k\mathrm{NN}}(s_i) = \left\{s_j \mid \operatorname{sim}\left(s_i, s_j\right) \geq \operatorname{sim}\left(s_i^k, s_j\right)\right\} \tag{5}$$

is a set of k nearest neighbors of spam message s_i, where s_i^k is a kth nearest neighbor of spam message s_i.

$$\begin{gathered} u_{ij} = \begin{cases} 1 & \text{if } s_j \in O_{k\mathrm{NN}}(s_i), \\ 0, & \text{otherwise,} \end{cases} \\ v_{ij} = \begin{cases} 1 & \text{if } s_i \in O_{k\mathrm{NN}}(s_i), \\ 0, & \text{otherwise.} \end{cases} \end{gathered} \tag{6}$$

If $u_{ij} = 1$ and $v_{ij} = 1$, then s_i and s_j will be mutual nearest neighbors. If $u_{ij} = 1$ and $v_{ij} = 0$ or $u_{ij} = 0$ and $v_{ij} = 1$, then s_i and s_j will be nearest neighbors. If $u_{ij} = 0$ and $v_{ij} = 0$, then s_i and s_j will not be nearest neighbors.

Let x_{ip} be a Boolean variable, which is equal to 1, if the spam message s_i belongs to cluster C_p; otherwise, it is equal to 0:

$$x_{ip} = \begin{cases} 1 & \text{if } s_i \in C_p, \\ 0 & \text{if } s_i \notin C_p, \end{cases} \qquad i = 1, \ldots, n; \ p = 1, \ldots, q. \tag{7}$$

Taking into account the above designations, the criterion function of clustering can be defined as follows:

$$f(x) = \sum_{p=1}^{q} \sum_{i=1}^{n} \sum_{j=1}^{n} \left(u_{ij} + v_{ij}\right) \operatorname{sim}\left(s_i, s_j\right) x_{ip} x_{jp} \longrightarrow \max. \tag{8}$$

As clusters are nonoverlapping, that is, each of the n messages belongs to only one of the q clusters, the following condition should be satisfied:

$$\sum_{p=1}^{q} x_{ip} = 1, \quad i = 1, \ldots, n. \tag{9}$$

On the other hand, it is supposed that each cluster contains at least one spam message and does not contain all spam messages.

$$1 < \sum_{i=1}^{n} x_{ip} < n, \quad p = 1,\ldots,q, \tag{10}$$

where

$$x_{ip} \in \{0,1\} \quad \text{for any } i, p. \tag{11}$$

So, the clustering problem of spam messages is formalized as the Boolean quadratic programming (8)–(11), the solution of which provides non-overlapping clusters. This kind of problem is called NP-complete problem, the solution of which requires feasible time and computing resources. For solving this problem, it is possible to apply such algorithms, as genetic algorithm, differential evaluation algorithm, particle swarm optimization algorithm, artificial bee colony optimization, ant colony algorithm, and neural network. As the number of spam-messages is huge and a collection dynamically replenishes, the solution of such problem demands the big computing expenses; therefore, to solve the problem (8)–(11) the genetic algorithm is offered.

5. Genetic Algorithm for Solving the Clustering Problem

Genetic algorithms are powerful tools for solving large dimension problems. But they do not guarantee an optimality of a found solution. In genetic algorithms, the first step is an encoding of solutions in the form of chromosomes which depends on the character of a solved problem. Therefore, before using genetic algorithm, at first it is necessary to design solutions of a problem in the form of a chromosome. Proceeding from character of a solved problem (8)–(11), a chromosome in populations is represented in such kind: $X = (x_{11},\ldots,x_{1k},x_{21},\ldots,x_{2k},\ldots,x_{n1},\ldots,x_{np})$, where genes (variables) x_{ip} $(i = 1,\ldots,n;\ p = 1,\ldots,q)$ according to (11) accept values 0 or 1. At such encoding, the size of a chromosome equals to $n \cdot q$, where the first q position corresponds to the first spam message, following q position to the second spam-message. For example, for $n = 7$ and $q = 3$, encoding $X = (0,0,1,\ 1,0,0,\ 0,1,0,\ 1,0,0,\ 1,0,0,\ 0,1,0,\ 0,0,1)$ describes that clustering in which spam messages s_1 and s_7 belong to cluster $C_3(x_{13} = x_{73} = 1)$; spam messages s_2, s_4, and s_5 belong to cluster $C_1(x_{21} = x_{41} = x_{51} = 1)$; and spam messages s_3 и s_6 belong to cluster $C_2(x_{32} = x_{62} = 1)$.

It is necessary to note, that genetic algorithms are easily applied to unconstrained optimization problems, but when solving constrained problems genetic algorithms are faced by a problem of occurrence of infeasible solutions. Infeasible solutions are solutions which break restrictions (in our case conditions (9) and (10)). When solving problem with genetic algorithm, the most important is to continuously support a feasibility of solutions during algorithm work, that is, maintenance of chromosomes such that they did not break restrictions (conditions) of a problem. There are different approaches for prevention of occurrence of infeasible chromosomes [36–38]. In this paper, in order to avoid maintenance of infeasible chromosomes, the penalty functions method is applied. The penalty functions method is very effective in constrained optimization problems [39, 40].

The idea of the penalty functions method is that it decreases the value of fitness function when infeasible chromosome occurs. So if the minimization problem is considered, then the introduced penalty function should sharply increase the fitness value of an infeasible chromosome. And on the contrary, if the maximization problem is considered, then the penalty function should be constructed so that it sharply reduced fitness value of an infeasible chromosome. Before constructing penalty function, we introduce the following designations:

$$\begin{aligned} u(x_{i\bullet}) &= \sum_{p=1}^{q} x_{ip}, \quad i = 1,\ldots,n, \\ v\left(x_{\bullet p}\right) &= \sum_{i=1}^{n} x_{ip}, \quad p = 1,\ldots,q. \end{aligned} \tag{12}$$

As the problem (8)–(11) is a maximization problem, when infeasible chromosome occurs the penalty function should sharply reduce the fitness value.

Taking into account the last statement, the penalty functions should be defined as follows:

$$U(x) = \prod_{i=1}^{n} e^{-\alpha|u(x_{i\bullet})-1|}, \tag{13}$$

$$V(x) = \prod_{p=1}^{q} e^{-\alpha h(v(x_{\bullet p}))}, \tag{14}$$

where $\alpha \geq 1$ is a deterioration coefficient and

$$h(t) = \begin{cases} 0 & \text{if } 1 < t < n, \\ 2 - t & \text{if } t \leq 1, \\ 1 & \text{if } t = n. \end{cases} \tag{15}$$

The function $U(x)$ (13) prevents occurrence of the infeasible chromosomes violating a condition (9), and function $V(x)$ (14) prevents occurrence of the infeasible chromosomes violating a condition (10).

It is easy to show that the functions (13) and (14) have the following conditions:

(i) if the condition (9) is satisfied, then $U(x) = 1$;

(ii) if for any i the condition (9) is not satisfied, then $U(x) \leq e^{-\alpha}$;

(iii) if the condition (10) is satisfied, then $V(x) = 1$;

(iv) if for some p the condition (10) is not satisfied, then $V(x) \leq e^{-\alpha}$.

Hence, if both conditions (9) and (10) are satisfied, that is, the solution is feasible, then $U(x)V(x) = 1$, and on the contrary if at least one of these conditions is not satisfied, that is, the solution is infeasible, then $U(x)V(x) \leq e^{-\alpha}$.

Considering the properties of penalty functions, multiplying the criterion function (8) to $U(x)V(x)$, the problem (8)–(11) with restrictions can be reduced to the following problem without restriction:

$$E(x) = f(x)U(x)V(x) \longrightarrow \max. \quad (16)$$

In another way, the chromosome can be designed in the form of a raw $X = (y_1, y_2, \ldots, y_n)$ with length n where alleles y_i define the number of clusters and accept the value from the set $\{1, 2, \ldots, p\}$, and loci (positions of genes) correspond to numbers of spam messages. On the basis of such representation, the solution of the problem is defined thus:

$$x_{iy_j} = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{if } i \neq j, \end{cases} \qquad i = 1, \ldots, n, \; j = 1, \ldots, m. \quad (17)$$

For example, $X = (2, 3, 1, 3, 4, 2, 3)$ corresponds to that division, that the spam messages s_1 and s_6 belong to cluster $C_2(x_{12} = x_{62} = 1)$, the spam messages s_2, s_4, and s_7 belong to cluster $C_3(x_{23} = x_{43} = x_{73} = 1)$, the spam message s_3 belongs to cluster $C_1(x_{31} = 1)$, and the spam message s_5 belongs to cluster $C_4(x_{54} = 1)$.

Let us note that at such designing of chromosomes the condition (9) will be always satisfied. In [37], it has been shown that implementation of such operators to feasible chromosomes does not lead to occurrence of infeasible chromosomes. It is effective only in that case when initial population is generated at observance of a condition (10). As the number of clustering spam messages is much more than the number of clusters $n \gg q$, in that case for $i_1 \neq i_2$ the equality $y_{i_i} = y_{i_2}$ takes place then for the generation of initial population the probability of occurrence of infeasible chromosomes will be very high. Thus, the time spent for observance of a condition (10), for the generation of initial population, can be compared with time for solving problem. Therefore, it is reasonable to apply a penalty function method. As for such encoding, the condition (9) is always satisfied then the criterion function will be in such form:

$$E(x) = f(x)V(x) \longrightarrow \max. \quad (18)$$

As a selection operator, the proportional selection, where the chromosome from current population $Z = (X_1, X_2, \ldots, X_d)$ is selected according to the probability defined by the formula:

$$z_d = \frac{F(X_d)}{\sum_{d=1}^{D} F(X_d)}, \qquad d = 1, \ldots, D, \quad (19)$$

is used.

Here D is a size of population, and $F(X_d)$ is a fitness value of chromosome X_d. Fitness function depends on a character of a solved problem. As in our case, the problem purpose consists in maximization of the function $E(x)$ then chromosome with bigger value of criterion function $E(x)$ should have every prospect to survive for the following generation. According to the last formula, it means that the chromosome with smaller value of criterion function $E(x)$ should have big fitness value. Taking into account this, fitness value $F(X_d)$ of chromosome X_d is defined as

$$F(X_d) = E(X_d). \quad (20)$$

So, applying the penalty function, method the infeasible solutions generated by operators of genetic algorithm will be eliminated during the process on ranging of fitness values, and feasible decisions will have more chances to survive, that is, the penalty functions method allows to accelerate process of convergence of genetic algorithm. This is because the penalty functions method at occurrence of infeasible chromosomes does not demand performance of additional operations (to make recoil and to return chromosome to the previous state, correction of infeasible chromosomes, etc.). Here it is necessary to note that any type of operators of crossing and a mutation could be used as input of penalty functions.

Now the stop criterion should be defined, which is an important step of genetic algorithm.

The maximization of compactness causes points in each cluster to be very similar to the corresponding center. Therefore, we will define coordinates of the centers of clusters. jth coordinate of the center O_p of the cluster C_p is calculated by the formula:

$$o_{pj} = \frac{1}{n_p} \sum_{d=1}^{n_p} w_{dj}, \qquad p = 1, \ldots, q; \; j = 1, \ldots, m, \quad (21)$$

where n_p is a number of points in cluster C_p. Obviously $\sum_{p=1}^{q} n_p = n$.

The compactness of cluster C_p is calculated by the following formula:

$$r_p = \frac{1}{n_p} \sum_{i=1}^{n} \text{sim}\left(s_i, O_p\right) x_{ip}. \quad (22)$$

The average similarity of the cluster C_p to other clusters we define as follows:

$$R_p = \frac{1}{q-1} \sum_{z=1}^{q} \text{sim}\left(O_p, O_z\right), \qquad p = 1, \ldots, q, \quad (23)$$

where $\text{sim}(O_p, O_z)$ is the similarity between the centers of the clusters C_p and C_z.

If the condition $\max_p(R_p/r_p) < 1$ is satisfied, then stop the genetic algorithm.

6. Classification Using the *k*NN Method

As the collection of spam messages permanently changes, replenishing with new types of spam messages after clustering, it is necessary to accompany clusters. So the collection of spam messages should be classified. For classification, the kNN method is used. The kNN method is used in many problems to determine a class to which the object belongs. This classification method is based on already available set of the classified objects. As in our case, objects are spam messages; then designating each new spam message coming to the collection as s_{n+1}, we will define the k nearest spam messages which already belonged to one of the classes.

The kNN classifier for each cluster C_p calculates the relevance score

$$\begin{aligned}\text{score}\left(s_{n+1}, C_p\right) &= \sum_{s' \in O_p(s_{n+1}) \cap S_p}^{q} \cos(s_{n+1}, s') \\ &= \sum_{i \in I_p(s_{n+1})} \cos(s_{n+1}, s_i) x_{ip},\end{aligned} \tag{24}$$

where $O_p(s_{n+1})$, $I_p(s_{n+1})$ are elements and their indexes of k-nearest neighbors of the spam message s_{n+1}; correspondingly, S_p is a set of spam messages in cluster C_p and

$$e_{ip} = \begin{cases} 1 & \text{if } s_i \in C_p, \\ 0 & \text{if } s_i \notin C_p. \end{cases} \tag{25}$$

The spam message s_{n+1} belongs to that class C_p, for which the value score (s_{n+1}, C_p) is a maximum. If score $(s_{n+1}, C_p) < \theta$, then spam message does not belong to any of the clusters C_p and in this case a new cluster C_{q+1} is created, where θ is a predefined threshold.

7. Knowledge Extraction from Classes Using Summarization Technique

At this stage after clustering of messages and solving the problem (8)–(11), it is necessary to define themes, descriptions of clusters. To receive the information about clusters, the document summarization method is applied. In our case, documents the spam messages which are belonged to the same theme and are in the same cluster. It is necessary to take the content from these sets of spam messages, deleting the unnecessary information and taking into account similar and differing moments in the content, and to present the most important information in a condensed form. Therefore, the multidocument summarization method described in the paper [30] can help to find informative sentences from each cluster. The multidocument summarization is a process of automatic creation of the compressed version of set of the documents giving to the user the helpful information. At the first stage in order to defining thematic sections, the clustering of spam messages is satisfied. And at the second stage in order to define the informativeness value and for extracting the informative sentences, the ranging is made. This will ensure to define representative sentences and their quantity for each thematic section, avoiding redundancy in the summary.

The representativeness of a sentence is defined by similarity measure between them and corresponding cluster centroid, that is, the less Euclidean distance between the sentence and corresponding cluster centroid means the sentence is more representative. To include sentences into summary, they are ranged in ascending order according to their similarity measures to corresponding cluster centroid. In this paper, each cluster consists of thematically close messages. Some messages contain many sentences and, hence, form the main content of the cluster. Other themes can be shortly mentioned to complete the main subjects. Hence, quantities of sentences in different clusters are different. Such approach allows maximum covering of main content of the cluster and avoids redundancy.

In general, the number of sentences included into summary depends on compression factor. Compression factor α_{comp} is determined by length summary and message:

$$\alpha_{\text{comp}} = \frac{\text{len(summ)}}{\text{len(E-mail)}} \tag{26}$$

and is a main factor influencing the quality of summary, where len(summ), len(E-mail) are the lengths of summary and message, correspondingly. As for minimum value of compression factor, the summary will be shorter, and the main part of the information will be lost. At the same time, at great value of factor of compression, the summary will be plentiful; however, it will contain insignificant sentences.

Considering the above-stated, the quantity of the representative sentences N_p which have been selected from each cluster C_p, calculated by the following formula is defined:

$$N_p = \text{INT}\left[\frac{\text{len}\left(C_p\right) \cdot \alpha_{\text{comp}}}{\text{len}_{\text{aver}}}\right], \quad p = 1, \ldots, q, \tag{27}$$

where $\text{len}(C_p)$ and $\text{len}_{\text{aver}} = \text{len(E-mail)}/m$ are the length of cluster C_p and the average length of sentences in message correspondingly, and $\text{INT}[\cdot]$ means the whole part.

8. Conclusion and Future Work

In this paper, the problem of clustering of spam messages collection is formalized. The criterion function is a maximization of similarity between messages in clusters, which is defined by k-nearest neighbor algorithm. Genetic algorithm including penalty function for solving clustering problem is offered. Classification of new spam messages coming to the bases of antispam system is also given. After classification, the knowledge extraction from divided classes is considered. Multidocument summarization method is applied for knowledge extraction from clusters. The information extracted from clusters and thematic dependence of spam messages from their origin can be also helpful in detection of social networks of spammers if they exist.

Though there are a lot of works offering methods for classification of E-mails into spam or nonspam, there is no one previous scientific work and experimental study showing classification of spam messages into thematically groups. In this context, it is decided to make experiments on this subject in future works, especially to show efficiency of the clustering method and genetic algorithm used in this paper in comparison with others.

Acknowledgment

The authors would like to express their appreciation to the anonymous reviewers for their very useful comments and suggestions.

References

[1] Symantec, "State of spam and phishing. A monthly report 2010," http://symantec.com/content/en/us/enterprise/other_resources/b-state_of_spam_and_phishing_report_09-2010.en-us.pdf.

[2] Ferris Research, "Cost of spam is flattening–our 2009 predictions," http://www.ferris.com/2009/01/28/cost-of-spam-is-flattening-our-2009-predictions/.

[3] C. Ray and H. Hunt, "Tightening the net: a review of current and next generation spam filtering tools," *Computers and Security*, vol. 25, no. 8, pp. 566–578, 2006.

[4] H. Wen-Feng and C. Te-Min, "An incremental cluster-based approach to spam filtering," *Expert Systems with Applications*, vol. 34, no. 3, pp. 1599–1608, 2008.

[5] M. L. Sang, S. K. Dong, and S. P. Jong, "Spam detection using feature selection and parameters optimization," in *Proceedings of the 4th International Conference on Complex, Intelligent and Software Intensive Systems, (CISIS '10)*, pp. 883–888, Krakow , Poland, February 2010.

[6] F. S. Mehrnoush and B. Hamid, "Spam detection using dynamic weighted voting based on clustering," in *Proceedings of the 2nd International Symposium on Intelligent Information Technology Application, (IITA '08)*, pp. 122–126, Shanghai , China, December 2008.

[7] S. Minoru and Sh. Hiroyuki, "Spam detection using text clustering," in *Proceedings of the International Conference on Cyberworlds, (CW '05)*, pp. 316–319, Singapore, November 2005.

[8] C. Paulo, L. Clotilde, S. Pedro et al., "Symbiotic data mining for personalized spam filtering," in *Proceedings of the International Conference on Web Intelligence and Intelligent Agent Technology, (IEEE/WIC/ACM)*, pp. 149–156, 2009.

[9] Kh. Ahmed, "An overview of content-based spam filtering techniques," *Informatica*, vol. 31, no. 3, pp. 269–277, 2007.

[10] S. Nazirova, "Mechanism of classification of text spam messages collected in spam pattern bases," in *Proceedings of the 3rd International Conference on Problems of Cybernetics and Informatics, (PCI '10)*, vol. 2, pp. 206–209, 2010.

[11] W. Lauren, "Spam wars," *Communications of the ACM*, vol. 46, no. 8, p. 136, 2003.

[12] G. Pawel and M. Jacek, "Fighting the spam wars: a remailer approach with restrictive aliasing," *ACM Transactions on Internet Technology*, vol. 4, no. 1, pp. 1–30, 2004.

[13] L. Fulu, H. Mo-Han, and G. Pawel, "The community behavior of spammers," http://web.media.mit.edu/~fulu/ClusteringSpammers.pdf.

[14] K. S. Xu, M. Kliger, Y. Chen, P. J. Woolf, and A. O. Hero, "Revealing social networks of spammers through spectral clustering," in *Proceedings of the IEEE International Conference on Communications, (ICC '09)*, Dresden, Germany, June 2009.

[15] K. S. Xu, M. Kliger, Y. Chen et al., "Tracking communities of spammers by evolutionary clustering," http://www.eecs.umich.edu/~xukevin/xu_spam_icml_2010_sna.pdf.

[16] G. Salton, *Dynamic Library–Information System*, Mir, Moscow, Russia, 1979.

[17] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," *Information Processing and Management*, vol. 24, no. 5, pp. 513–523, 1988.

[18] S. V. Mochenov, A. M. Blednov, and U. A. Lugovskikh, "Vector representation of the textual information," in *Proceedings of the International Scientific Conference Materials*, pp. 131–139, 2006.

[19] R. M. Alguliev and R. M. Alyguliev, "Automatic text documents summarization through sentences clustering," *Journal of Automation and Information Sciences*, vol. 40, no. 9, pp. 53–63, 2008.

[20] G. Vishal and G. Si Lehal, "A survey of text mining techniques and applications," *Journal of Emerging Technologies in Web Intelligence*, vol. 1, no. 1, pp. 60–76, 2009.

[21] X. Li and N. Ye, "A supervised clustering and classification algorithm for mining data with mixed variables," *IEEE Transactions on Systems, Man, and Cybernetics Part A*, vol. 36, no. 2, pp. 396–406, 2006.

[22] T. Li, "A unified view on clustering binary data," *Machine Learning*, vol. 62, no. 3, pp. 199–215, 2006.

[23] J. Grabmeier and A. Rudolph, "Techniques of cluster algorithms in data mining," *Data Mining and Knowledge Discovery*, vol. 6, no. 4, pp. 303–360, 2002.

[24] M. Halkidi, Y. Batistakis, and M. Vazirgiannis, "On clustering validation techniques," *Journal of Intelligent Information Systems*, vol. 17, no. 2-3, pp. 107–145, 2001.

[25] D. R. Tauritz, J. N. Kok, and I. G. Sprinkhuizen-Kuyper, "Adaptive Information Filtering using evolutionary computation," *Information Sciences*, vol. 122, no. 2, pp. 121–140, 2000.

[26] J. Lijuan and F. Liping, "Text classification based on Ant Colony Optimization," in *Proceedings of the 3rd International Conference on Information and Computing, (ICIC '10)*, pp. 229–232, Jiang Su , China, June 2010.

[27] E. R. Hruschka, R. J. G. B. Campello, A. A. Freitas, and A. C. P. L. F. de Carvalho, "A survey of evolutionary algorithms for clustering," *IEEE Transactions on Systems, Man and Cybernetics Part C*, vol. 39, no. 2, pp. 133–155, 2009.

[28] R. M. Alguliev and R. M. Aliguliyev, "Fast genetic algorithm for solving of the clustering problem of text documents," *Artificial Intelligence*, vol. 3, pp. 698–707, 2005 (Russian).

[29] R. Ghaemi, N. Sulaiman, H. Ibrahim et al., "A review: accuracy optimization in clustering ensembles using genetic algorithms," *Artificial Intelligence Review*, vol. 35, no. 4, pp. 287–318, 2011.

[30] R. M. Alguliev and R. M. Aliguliyev, "A new summarization method of text documents and evaluation of classification result in three aspects," *Telecommunications*, vol. 3, pp. 7–16, 2006 (Russian).

[31] R. M. Alguliev and R. M. Aliguliyev, "Effective summarization method of text documents," in *Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence, (WI '05)*, pp. 264–271, September 2005.

[32] R. M. Alyguliyev, "The two-stage unsupervised approach to multidocument summarization," *Automatic Control and Computer Sciences*, vol. 43, no. 5, pp. 276–284, 2009.

[33] R. M. Alguliev and S. A. Nazirova, "Mechanism of forming and realization of anti-spam policy," *Telecommunications*, vol. 12, pp. 38–43, 2009 (Russian).

[34] L. Kyung-Chan, K. Seung-Shik, and H. Kwang-Soo, "A term weighting approach for text categorization," *Lecture Notes in Computer Science*, vol. 3689, pp. 673–678, 2005.

[35] G. Patanè and M. Russo, "Comparisons between fuzzy and hard clustering techniques," in *Proceedings of the Advances in Fuzzy Systems and Intelligent Technologies, (WILF '99)*, pp. 176–184, 1999.

[36] N. N. Glibovec and S. A. Medvid', "Genetic algorithms used to solve scheduling problems," *Cybernetics and System Analysis*, vol. 39, no. 1, pp. 81–90, 2003.

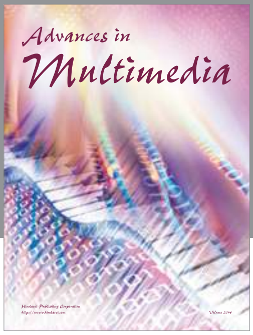
[37] T. Witkovski, S. Elzway, and A. Antchak, "Designing of the main operations of genetic algorithms for production

scheduling," *Journal of Automation and Information Sciences*, vol. 35, no. 12, pp. 50–58, 2003.

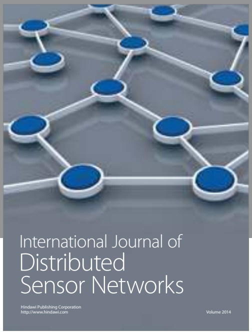
[38] R. M. Alguliev and R. M. Alyguliev, "A genetic approach to quasi-optimal assignment of tasks in the distributed system," *Telecommunications and Radio Engineering*, vol. 64, no. 2, pp. 97–108, 2005.

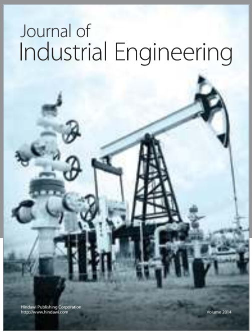
[39] A. L. Olsen, "Penalty functions and the knapsack problem," in *Proceedings of the 1st IEEE Conference on Evolutionary Computation*, pp. 554–558, June 1994.

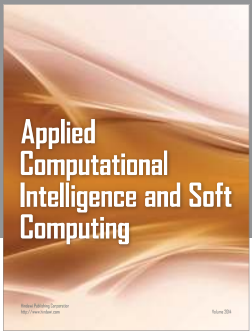
[40] Z.-J. Lee, S.-F. Su, C.-Y. Lee et al., "A heuristic genetic algorithm for solving resource allocation problems," *Knowledge and Information Systems*, vol. 5, no. 4, pp. 503–511, 2003.

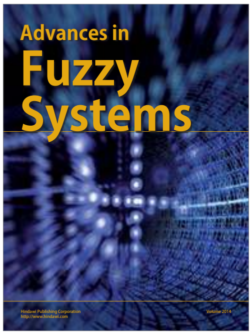

Advances in
Multimedia

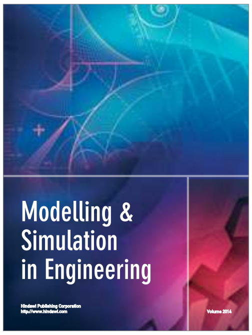
The Scientific
World Journal

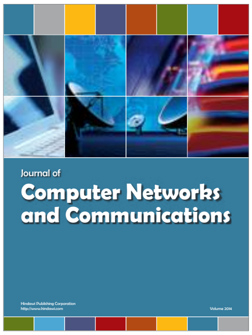

International Journal of
Distributed
Sensor Networks

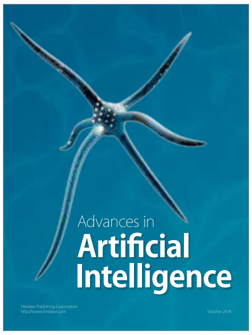

Journal of
Industrial Engineering

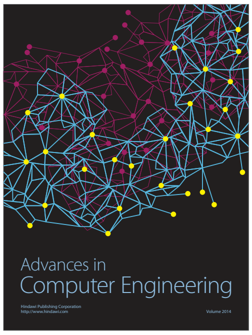

Applied
Computational
Intelligence and Soft
Computing


Advances in
Fuzzy
Systems

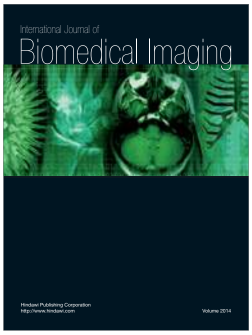

Modelling &
Simulation
in Engineering

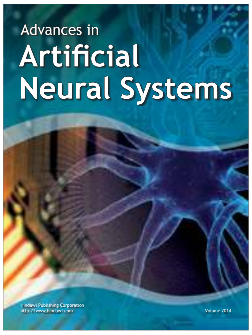

Journal of
Computer Networks
and Communications

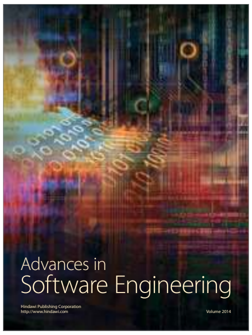

Advances in
Artificial
Intelligence


Advances in
Computer Engineering

International Journal of
Computer Games
Technology

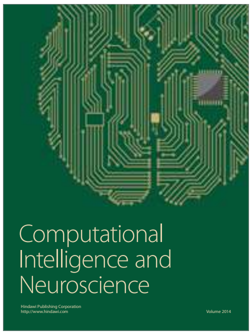

International Journal of
Biomedical Imaging


Advances in
Artificial
Neural Systems


Advances in
Software Engineering

Journal of
Robotics

Advances in
Human-Computer
Interaction


Computational
Intelligence and
Neuroscience

International Journal of
Reconfigurable
Computing

Journal of
Electrical and Computer
Engineering