THEMATIC AND SELF-LEARNING METHOD FOR PERSIAN, ENGLISH AND PENGLISH SPAMS IDENTIFICATION

Seyyed Yasser Hashemi and Khalil Monfaredi

[1, 2] Department of Computer Engineering, Miyandoab Branch, Islamic Azad University, Miyandoab, Iran.

Abstract

Spams are unwanted and also undesirable emails which are mass sent to the numerous victims. Further penetration of spams into electronic processors and communication equipments such as computers and mobiles as well as lack of control on the information shared on the internet and other communication networks and also inefficiency of the spam detecting methods developed for Persian contexts are among the main challenging issues of the Persian subscribers. This paper presents a novel and efficient method for thematic identification of Persian spams. The proposed method is capable of identifying the Persian, spams and also "Penglish" spams. "Penglish" is made up of two words Persian and English and demonstrates a Persian text which is written by English alphabetic letters. Based on the experimental analysis of the 10000 spams of different type the efficiency of the proposed method is evaluated to be more than 98%. The presented method is also capable of updating its databases taking the advantage of the feedbacks received from the users.

Keywords

Persian spams, Penglish spams, black list, neural list, trainable.

1.INTRODUCTION

Nowadays electronic email inbox of nearly all of the users are full of unwanted and also undesirable emails and advertisements which may include immoral or false claims. Spam filtering has become a very important issue in the last few years as unsolicited bulk email imposes large problems in terms of both the amount of time spent on and the resources needed to automatically filter those messages [1]. Email communication has come up as the most effective and popular way of communication today. People are sending and receiving many messages per day, communicating with partners and friends, or exchanging files and information. Email data are now becoming the dominant form of inter and intra-organizational written communication for many companies and government departments. Emails are the essential part of life now just likes mobile phones & i-pods [1]. Based on statistical reports revealed by "Pingdow" site which are derived from reputable resources such as "Net craft" and "Web hosting", about 90 trillion emails were sent in 2009 which 81 percent of them were spams and 24 percent increment are deduced from this number regarding the year 2008 statistics. This becomes much worse with increasing the penetration coefficient of the internet among the users in recent years.

The spams not only spread expeditiously into internet resources but also they are further penetrated into other electronic processors and communication equipments such as computers and

mobiles. This essentially requires development of much more efficient spam detecting methods to identify and prevent the received spams. Lack of control on the information spread on the internet and insufficient legal rules for punishment of the internet convicts in some countries make it relatively difficult to introduce a comprehensive solution for this challenging issue, thus the users rely mostly on the special services provided by the companies for spam filtering or they must do it manually.

In order to exclude the spams, all of the received emails must undergo a procedure which labels the emails as spam or non-spam. The words included in the emails could be used as the elements for labeling. Also with a common method which occurs in structure of initial genetic algorithm and biological sequence, a special morphological structure must be introduced to allow every user identify the correctness of received emails.

Although hidden Markov model (HMM) classification is favorably applicable, but some improvements are also applied to further enhance the capability of the proposed method which will be extensively described in following sections. For doing this, the automatic feedback received from the users would be more helpful. This model could be also used for designing spell check and correction toolboxes to be used within text processors or as a new method in search engines when a spelling mistake occurs in the category of search. All of the mentioned cases provide a situation for classifying words and sentences based on detailed information and improper template of the words. In order to overcome this issue the proposed model will be partially based on HMM [2] and the further improvements on the design are performed using the advantage of user received feedbacks. The program is developed based on words initially filled in black list and neural list which is modeled as an improved HMM. Then the email will be analyzed versus the available black lists to extract the number of occurrence of the forbidden words, statistically. Based on the threshold value which is also determined by the algorithm regarding the area of context the email will be labeled as spam or non-spam.

In section 2 the proposed improved HMM is modeled. Section 3 described the method adopted for observed sequence statistical calculation. Model learning is included in section 4. Spam classification is explained in section 5. Experimental results achieved from the proposed method are analyzed in section 6 and finally the paper is concluded in section 7.

2. PROPOSED IMPROVED HIDDEN MARKOV MODELING

2.1. Overview

Spams are usually used as an advertisement of a product which is not required by the user or include a wrong or immoral context. One of the most common methods which are widely used for spam identification is the methods operate based on an initially prepared black list. The black list includes the words which are occasionally repeated in spams. The message will be labeled as spam by means of spam filters if forbidden words repetition exceeds an allowable extreme.

Spam senders try to change the format of produced spams to make their identification difficult or even impossible. Spelling errors and symbol inclusion between word's characters are some of these tricks. For example, if the black list includes the word "درآمد", spam senders change it to "دررآممد, "در..آمد" or other formats in the email context to hide it from spam filters. Including all possible instances of the forbidden words in the spam black list will enlarge the black list extensively which will make the spam identification extremely slow, inefficient and unfeasible. Therefore there is an urgent need for a method capable of identifying all possible changes made to the initial correct words included in the black list.

The most important step in the proposed method is the way HMM is created for every word available in the black lists which are extensively described in the following part. In proposed method we use from multi-level blacklists. Critical spam words puts in major black list, common spam word puts in common blacklist and other spam words classified in no-significant black list. If the effective coefficients of any word be less than the corresponding section threshold, it word is moved to lower group and conversely. For spam detection the first step after creating the HMM, is to recognize the language of the message. This is easily done using the letters composing the email context. The results obtained from the language detection phase are analyzed in two distinct, identical and parallel phases, simultaneously. These two phases are business and moral ones. In each one of those execution phases, all words of the message undergo a processing procedure to determine the possession coefficient of them to every black list. Then, based on the obtained results, the sum of effective coefficient of discovered forbidden words is calculated to be used as a criterion for deciding whether the email is spam or not. The email could be recognized as spam in one, both, or none of the business and moral categories depending on its content. The figure 1 shows the overall steps of the proposed method.

Three-stage Black list Is Increased rate and efficiency improvement because firstly a high speed processing on limit words is performed that may Spam e-mail to confirm. If no definitive diagnosis, the Processing is done on the common blacklist and the results will be analyzed and if necessary, the Processing is done on a no-significant blacklist.

2.2. English, and Penglish spam classifications

Based on an investigation [3], Persian spams are mainly divided in two major categories: First, spams intending to advertise or sell a product or economically seduce the users and second, the spams include immoral contents. Thus the emails are analyzed distinctly in two different phases to identify the advertisement and immoral spams. In this approach every phase uses separate black lists and will result a much more optimum results regarding the conventional unique and comprehensive method.

One of the most challenging issues related to Persian spam identification is the variety of writing types which are as follows: Persian, English, and “Penglish”. As mentioned before the “Penglish” is a Persian text written with Latin alphabets. Thus three black lists are prepared for each group of immoral and advertisement categories. First black list is created for Persian spams. Second black list is created for English spams and finally, the last black list is created for “Penglish” spams. With the proposed method parallel execution of different analyses are possible and thus the speed and efficiency of the algorithm is expeditiously increased. Different forbidden words are not added to the black list identically; in fact each word owns an effective coefficient which is extracted experimentally investigating numerous spam messages. The words included in the black lists as well as their effective coefficients are updated taking into account the received feedbacks from the users. This is considered as the most dominant improvements of the proposed method comparing with the other artworks. After black list identification, for each word of the black lists, HMM must be created.

The initial black lists and the effective coefficients of included words are updated using the feedbacks received from the users. In order to do this, a neural list including all of pronouns, prepositions and also frequently used words is created and beside that a list of words capable of being in that neural list is established. On the other hand, for each one of the black lists a peripheral list of words which have the capability to be included in that black list under certain circumstances are also created. For updating purpose the emails which are labeled by the machine as spam and decided by the user as non-spam and vice versa are utilized which is depicted in

detail in figure 2. This figure shows the neural and black lists update procedure based on the feedback received from mass users.

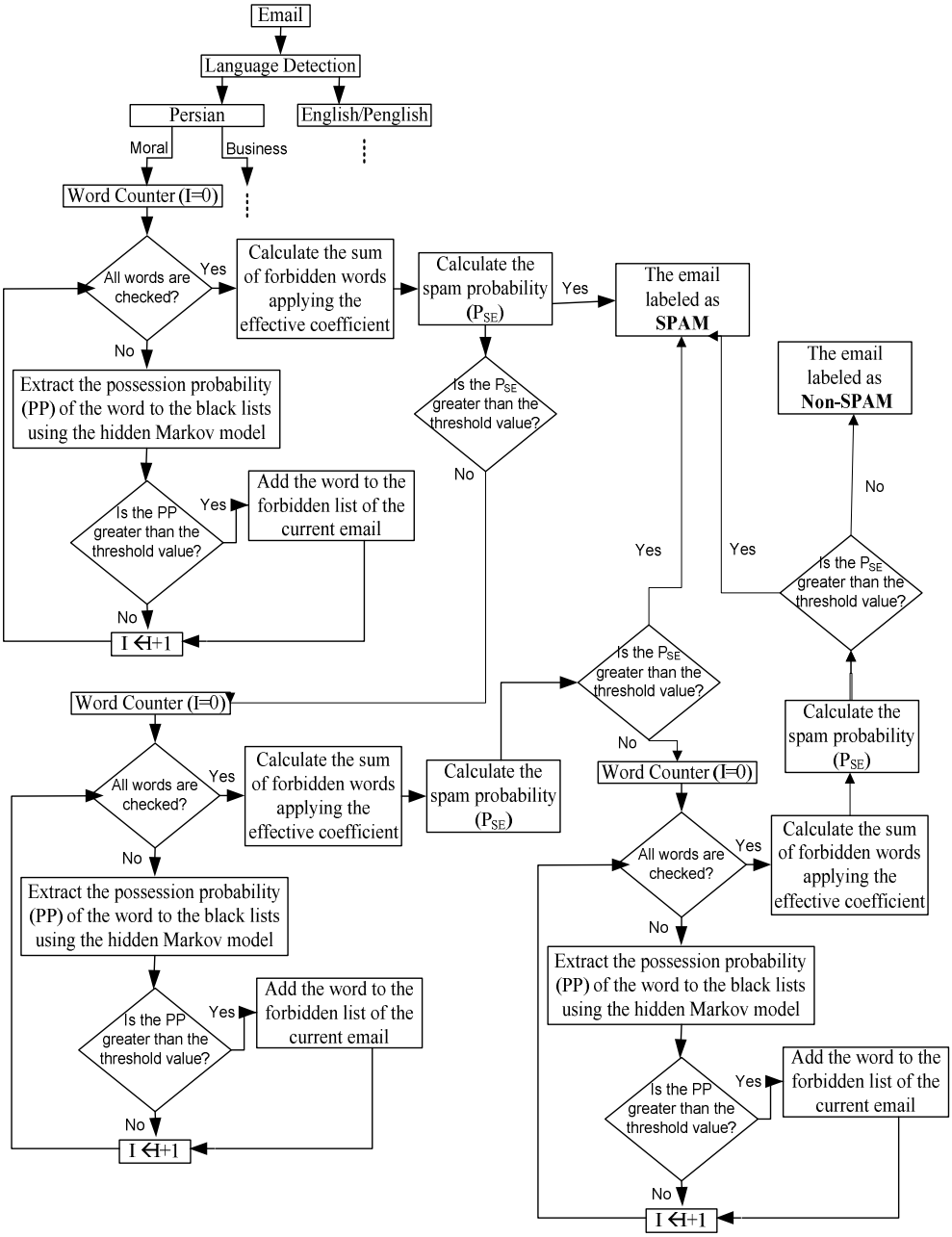

Figure 1. The proposed method overview

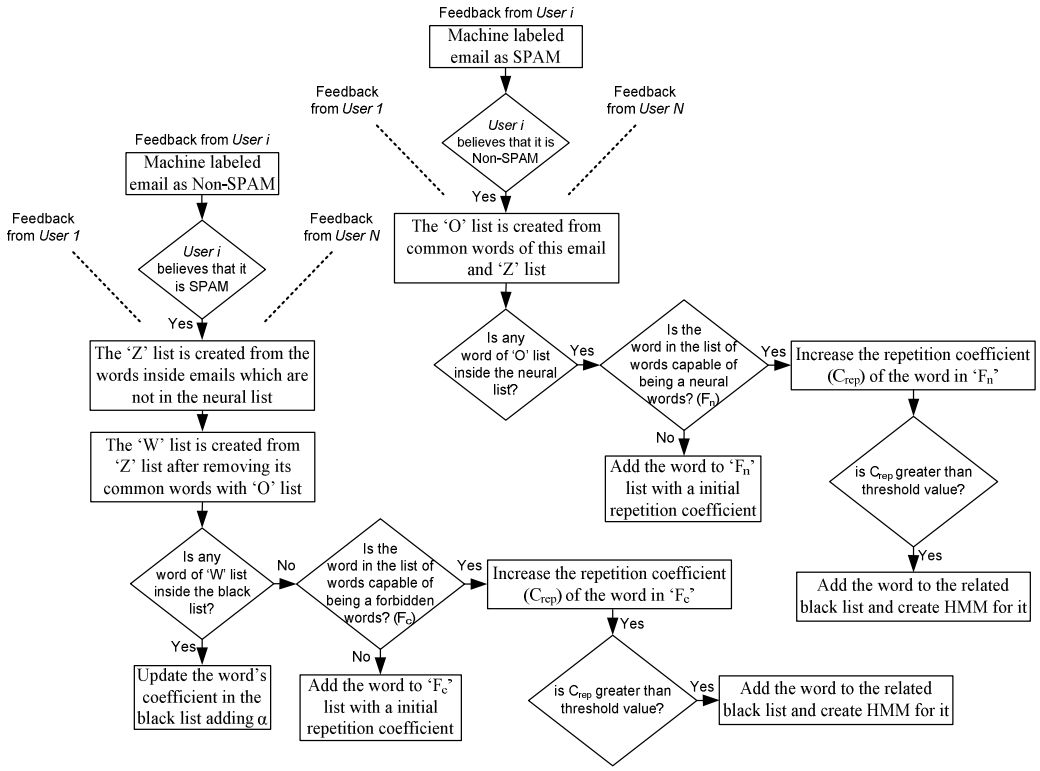

Figure 2. The neural and black lists update procedure based on the feedback received from mass users

2.3. Hidden Markov model

The most important part of the proposed method is creation of hidden Markov model $(\underline{X},\underline{Y}) = \{(X_t, Y_t)_{t\in \mathrm{N}}\}$ and the parameter collections dependent to it $(A, B, \lambda, \underline{\pi})$ for each word available in the black list. In this model $\underline{Y}$ is visible state set of the model, $\underline{X}$ is hidden state set of the model, A is the transmission matrix, B is emission matrix, and $\underline{\pi}$ is initial probability vector. In following subsections all of these parameters and the way they are used for creating the HMM will be discussed by a typical example for "درآمد" which is one of the black list words.

2.3.1. $\underline{Y}$: observed states

Observed states for every word is made up of letters, numbers and also symbols available on the keyboard. It must be noticed that these states are different for each one of aforementioned three types of spams. The observed states for the words available in the English and "Penglish" black list types include English alphabetical letters, numbers and also the special symbols available on the keyboard. But there is a major difference related to Persian black lists in comparison to English and "Penglish" types because the word structure in Persian language is different. The Persian language is made up of 32 letters which their appearance depends strongly on the position (beginning letter, middle letter or last letter) that they occupy in the word. According to the mentioned point, the total number of 114 different cases exists for the letters. Therefore the total observed states for the words available in the Persian black list include 114 different cases exist for Persian alphabetical letters, Persian numbers and also the special symbols available on the

keyboard for Persian language. For simplifying the emission matrix, symbols are put in one group. Therefore, taking the word "درآمد" into account for explanation, $\underline{L}$ is the set of all of the letters not available in the word "درآمد", and $\underline{Q}$ includes all symbols such as (!@#$ً ٌ ٍ ّ ْ...). For example if the word is "دآمد", the letter 'ر' is omitted and the observed state for the model will be as follows:

$$\underline{Y} = \{ \text{'د'}, \text{'م'}, \text{'ا'}, \text{'ر'}, \text{'د'}, L, Q, * \}$$

The symbol '*' is used to show the letter removal from the word.

2.3.2. $\underline{X}$: hidden states

Hidden states are described as the logical results of three different operations applied on the characters of the original word. These three operations are as follows:

Delete: shows the positions which the letters are removed. For the sequence related to the word "دآمد" the letter 'ر' is removed.

Insertion: shows the positions that the letters are added. For example for the sequence created for the word "دررآمد" the letter 'ر' is added to the original word.

Match: shows the positions that the letters are substituted by other letters. For example, for the sequence created for the word "دزآمد" the letter 'ز' is substituted in place of the letter 'ر'.

Suppose that R is the length of the model which determines the number of characters of the original word (For the word "درآمد" R is equal to 5). In order to explain all possible predicates between hidden states which are categorized below, the graph of figure 3 is utilized. Figure 3 shows the HMM states diagram when the length of model is equal to five.

- **Delete cases** which are depicted by circles: $d_1, d_2, ..., d_R$
- **Insertion cases** which is depicted by lozenges: $i_0, i_1, i_2, ..., i_R$
- **Match cases** which is depicted by squares: $m_1, m_2, ..., m_R$

The i_0 is predicted to show the insertions occurs before the first letter of the original word. Two fictitious states of 'bb' and 'ee' are created for showing the beginning and termination to make the graph clearer. The state 'bb' will be used for simplicity of initial vector $\underline{\pi}$ and the state 'ee' is used for well determination of the transmission matrix which will be explained more in following sections.

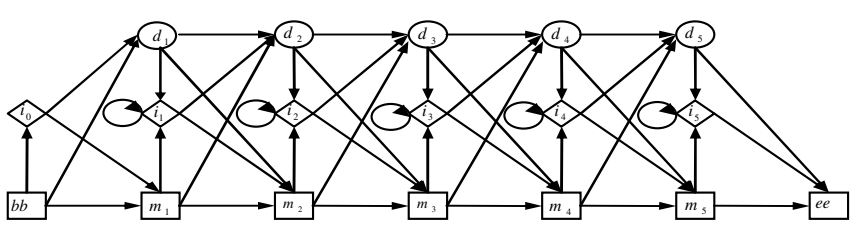

Figure 3. Hidden Markov model states diagram for the length of 5.

As can be seen from the graph of Figure 3 only possible references for the match state of m_r or the delete state of d_r are next insertion state, match state, and delete state (i_r, m_{r+1}, and d_{r+1}).

On the other hand possible references from one insertion state are next match and delete state (m_{r+1} and d_{r+1}) or the insertion state itself (i_r) for investigation of the probability of multiple repetition of several characters between two character of the original word. Since the length of the word is not constant, thus it must be considered that every path that starts from 'bb' and terminates to 'ee', depends directly to the edition form of the word made by the spam sender. Every observed sequence can be compatible with several paths of this graph. Two examples are explained for the word "درآمد" as follows:

Example 2-1: If the observed sequence is "دزآم..د", then the propagation sequence with the highest probability is:

bb	m_1	m_2	m_3	m_4	i_4	i_4	m_5	ee
*	د	Q	ا	Q	Q	م	د	*

Which the values of 'ز' and '.' are selected from Q .

The other sequence of hidden states can be as follows:

bb	m_1	d_2	i_2	m_3	m_4	d_5	i_5	i_5	i_5	ee
*	د	Q	Q	ا	م	*	Q	Q	د	*

Consequently the hidden states are $X = \{bb, i_0, d_1, i_1, m_1, ..., d_R, i_R, m_R, ee\}$. Therefore the number of hidden states is equal to $3(R+1)$ where R determines the model length.

2.3.3. λ parameters

In order to determine the HMM, determination of λ parameters which are provided in transmission matrix A , emission matrix B and initial probability vector are essential. The transmission matrix A is a spars matrix introduced in Figure 4 with dimension of $3(R+1)\times 3(R+1)$. The transmission matrix structure is as follows:

The matrix A possesses different values for its elements in a way that $\sum_{j\in X} a_{ij} = 1, \forall i \in X$ in which a_{ij} are the elements of the matrix A . Considering the matrix structure and assuming different matrix values, all elements which must be calculated are $6+9(R+1)+6 = 9R+3$.

Transmission matrix information could be combined by means of the following limitations related to patterns. For example one can give a_{bbm_1} higher probability versus other elements of the row, because the probability of sequence occurrence of letters starting with the first letter of the original word is higher than others. Also the probability value of $a_{m_j m_{j+1}}$ for every j , is more than other elements of the row $\left(a_{m_j m_{j+1}} \geq a_{m_j i_j} \text{ and } a_{m_j m_{j+1}} \geq a_{m_j d_{j+1}}\right)$ because the match state will occur naturally more than delete or insertion states.

After transmission matrix, the emission matrix B is evaluated. Recall that total $|Y|$ is observable and the characters of the original word which is not observed are classified in letter and symbol parts. The symbol '*' is used to model the not happening the delete state and also the occurred states are used to refer to the 'ee' and 'bb' states, thus these states do not have any output. Thus the emission matrix is achieved as is shown in Figure 4.

$$B = \begin{array}{c} \\ bb \\ i_0 \\ m_1 \\ d_1 \\ i_1 \\ m_2 \\ d_2 \\ i_2 \\ m_3 \\ d_3 \\ i_3 \\ m_4 \\ d_4 \\ i_4 \\ m_5 \\ d_5 \\ i_5 \\ ee \end{array} \begin{array}{c} \begin{array}{cccccccc} \text{د} & \text{ر} & \text{ا} & \text{م} & \text{د} & L & Q & * \end{array} \\ \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0.05 & 0.05 & 0 & 0 & 0 & 0.1 & 0.8 & 0 \\ 0.9 & 0.05 & 0 & 0 & 0 & 0 & 0.05 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0.2 & 0.1 & 0 & 0 & 0 & 0.1 & 0.6 & 0 \\ 0.05 & 0.6 & 0.05 & 0 & 0 & 0.1 & 0.2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0.05 & 0.15 & 0.05 & 0 & 0 & 0.3 & 0.45 & 0 \\ 0 & 0.05 & 0.75 & 0.05 & 0 & 0.1 & 0.05 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0.05 & 0.05 & 0 & 0.2 & 0.7 & 0 \\ 0 & 0 & 0.1 & 0.55 & 0.05 & 0.05 & 0.25 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0.05 & 0.05 & 0.1 & 0.8 & 0 \\ 0 & 0.05 & 0 & 0.05 & 0.75 & 0.05 & 0.1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0.15 & 0.05 & 0.3 & 0.5 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \end{array}$$

Figure 4. Emission Matrix of the word "درآمد"

Some other restrictions can be applied to the matrix B parameters. For example, the probability value of $b_{m_{1د}}$ and $b_{i_{kQ}}$ must be higher than other elements of the row. Consequently, the initial probability vector of $\underline{\pi}$ is created. Fictitious state 'bb' existence, simplifies the probability vector definition, extremely, because all of the hidden states start from this state, thus all probabilities are concentrated on this state. Initial probability vector is given as follows:

$$\underline{\pi}^T = \begin{array}{c} \begin{array}{cccccc} bb & i_0 & d_1 & m_1 & & ee \end{array} \\ (1 \quad 0 \quad 0 \quad 0 \quad \cdots \quad 0) \end{array} \tag{1}$$

By defining the parameters of the transmission and emission matrix and initial probability vector the model is created for each word. For example, nonzero blocks of matrix A for model of "درآمد" whose emission matrix are shown in Figure 4 are given as follows:

$$A_1 = \begin{array}{c} \\ bb \\ i_0 \end{array}\begin{array}{c} \begin{array}{ccc} i_0 & m_1 & d_1 \end{array} \\ \begin{bmatrix} 0.05 & 0.9 & 0.05 \\ 0.2 & 0.75 & 0.05 \end{bmatrix} \end{array} \qquad A_2 = \begin{array}{c} \\ m_2 \\ d_2 \\ i_2 \end{array}\begin{array}{c} \begin{array}{ccc} i_2 & m_3 & d_3 \end{array} \\ \begin{bmatrix} 0.2 & 0.5 & 0.3 \\ 0.05 & 0.9 & 0.05 \\ 0.1 & 0.85 & 0.05 \end{bmatrix} \end{array}$$

$$A_3 = \begin{array}{c} \\ m_2 \\ d_2 \\ i_2 \end{array}\begin{array}{c} \begin{array}{ccc} i_2 & m_3 & d_3 \end{array} \\ \begin{bmatrix} 0.15 & 0.8 & 0.05 \\ 0.3 & 0.6 & 0.1 \\ 0.4 & 0.5 & 0.1 \end{bmatrix} \end{array} \qquad A_4 = \begin{array}{c} \\ m_3 \\ d_3 \\ i_3 \end{array}\begin{array}{c} \begin{array}{ccc} i_3 & m_4 & d_4 \end{array} \\ \begin{bmatrix} 0.15 & 0.6 & 0.25 \\ 0.25 & 0.7 & 0.05 \\ 0.4 & 0.4 & 0.2 \end{bmatrix} \end{array}$$

$$A_5 = \begin{array}{c} \\ m_4 \\ d_4 \\ i_4 \end{array}\begin{array}{c} \begin{array}{ccc} i_4 & m_5 & d_5 \end{array} \\ \begin{bmatrix} 0.35 & 0.55 & 0.1 \\ 0.1 & 0.8 & 0.1 \\ 0.3 & 0.6 & 0.1 \end{bmatrix} \end{array} \qquad A_6 = \begin{array}{c} \\ m_5 \\ d_5 \\ i_5 \end{array}\begin{array}{c} \begin{array}{cc} i_5 & ee \end{array} \\ \begin{bmatrix} 0.2 & 0.8 \\ 0.05 & 0.95 \\ 0.1 & 0.9 \end{bmatrix} \end{array}$$

Here the sequence shown for “دزآم..د” from example 2-1 is analyzed using two sequences of hidden states.

$$\begin{array}{lccccccccc} X^1 = & bb & m_1 & m_2 & m_3 & m_4 & i_4 & i_4 & m_5 & ee \\ Y^1 = & * & \text{د} & Q & \text{ا} & \text{م} & Q & Q & \text{د} & * \end{array}$$

$$\begin{array}{lccccccccccc} X^2 = & bb & m_1 & d_2 & i_2 & m_3 & m_4 & d_5 & i_5 & i_5 & i_5 & ee \\ Y^2 = & * & \text{د} & * & Q & \text{ا} & \text{م} & * & Q & Q & \text{د} & * \end{array}$$

Here it will be proved that the probability of first sequence is higher than the second one.

For first sequence:

$$p_\lambda\left[\underline{X}^1,\underline{Y}^1\right] = \pi_{bb}.b_{bb*}.a_{bbm_1}.b_{m_1\text{د}}.a_{m_1m_2}.b_{m_2Q}.a_{m_2m_3}.b_{m_3\text{ا}}.a_{m_3m_4}.b_{m_4\text{م}}.a_{m_4i_4}.b_{i_4Q}.a_{i_4i_4}.b_{i_4Q}.a_{i_4m_5}.b_{m_5\text{د}}$$
$$.a_{m_5ee}.b_{ee*} = 1\times1\times0.9\times0.9\times0.5\times0.2\times0.8\times0.75\times0.6\times0.55\times0.35\times0.8\times0.3\times0.8\times0.6$$
$$\times0.75\times0.8\times1 = 3.84\times10^{-4}$$

For second sequence:

$$p_\lambda\left[\underline{X}^2,\underline{Y}^2\right] = \pi_{bb}.b_{bb*}.a_{bbm_1}.b_{m_1\text{د}}.a_{m_1d_2}.b_{d_2*}.a_{d_2i_2}.b_{i_2Q}.a_{i_2m_3}.b_{m_3\text{ا}}.a_{m_3m_4}.b_{m_4\text{م}}.a_{m_4d_5}.b_{d_5*}.a_{d_5i_5}.b_{i_5Q}$$
$$.a_{i_5i_5}.b_{i_5Q}.a_{i_5i_5}.b_{i_5\text{د}}.a_{i_5ee}.b_{ee*} = 1\times1\times.09\times0.9\times0.3\times1\times0.3\times0.45\times0.5\times0.75\times0.6\times0.55\times0.1\times1$$
$$\times0.05\times0.5\times1\times0.5\times0.1\times0.05\times0.9\times1 = 6.8505\times10^{-10}$$

It is clearly seen that $p_\lambda\left[\underline{X}^1,\underline{Y}^1\right] > p_\lambda\left[\underline{X}^2,\underline{Y}^2\right]$.

For calculation whether “دزآم..د” is a modified version of the original word “درآمد” or not, all hidden sequences for “دزآم..د” must be achieved in “درآمد” model and the common probabilities must be added. In fact a model is created and based on the number of repetitions and also the calculated probability of the sequence the spams are identified.

3.PROBABILITY CALCULATION OF THE OBSERVED SEQUENCE

In this section the probability calculation method of the observed sequence $\underline{Y}$ with the aid of λ parameters will be described. The observed sequence $\underline{Y} = (y_1, \cdots, y_T)$ is located in λ model with the length of R. It is assumed that S is the hidden sequence length omitting the fictitious states and also it is supposed that $X_0 = bb$ and $X_{S+1} = ee$. It must be considered that these lengths may differ from case to case. The R, S, and T determine the number of steps of the following types:
Delete states $D = S - T$,

Insertion states $I = S - R$, and

Match states $M = R + T - S$

For next analysis determining the value of S is essential. The value of S is not initially determined, the only available information is:

$S \geq R$ because of the fact that the insertions in the sequence of the original word is probable.

$T \geq S$ because some of the characters may be removed.

$R \geq D = S - T$ since the number of removals cannot exceed the length of the model.

$T \geq I = S - R$ because of the fact that the number of insertions cannot exceed the observed length.

Consequently, $\text{Max}\{R, T\} \geq S \geq R + T$ thus S cannot become greater than $R + T$.

S is considered as a parameter and the problem can be solved for different values of that parameter and its minimum values of probability can be achieved. On the other hand since the boundaries of S values are small, this procedure is feasible while it is impossible for biological cases in which the boundaries are not so small. The other probable method which could be applied is using the Baysian paradigm [4-6]. Using a posteriori distribution for the variable $S : P[S = s]$ with $\text{Max}\{R, T\} \geq S \geq R + T$, one can achieve the probability of the observed sequence $\underline{Y}$ for different values of S. In other words, $P\left[S = s \,\middle|\, \underline{Y}\right]$ can be achieved by the posteriori distribution of the equation (2).

$$\mathrm{P}[\mathrm{S} = \mathrm{s}^* | \underline{\mathrm{Y}}] = \frac{\left[P[\underline{\mathrm{Y}} | \mathrm{S} = \mathrm{s}^*].P[S = s^*]\right]}{\sum_{s=\max\{R,T\}}^{R+T} P[\underline{\mathrm{Y}} | \mathrm{S} = \mathrm{s}].P[S = s]} \tag{2}$$

Therefore a value is achieved for S which the maximum probability is extracted based on it. The sequence $\underline{Y} = (y_1, \cdots, y_T)$ and the λ parameters are available and the probability $P_\lambda[\underline{Y}]$ can

be calculated. In fact $P_\lambda[\underline{Y}]$ is the probability value of achieving the observed sequence $\underline{Y}$ from the model.

The HMM uses forward and backward algorithms for classifying and determining the probability value [7-9]. In this paper using the same algorithm with some improvements the considered probability for observed sequence is extracted. Using the forward algorithm reduces the calculation complexity, extensively. If all possible probabilities for hidden sequences are calculated based on the graph using $\underline{Y} = (y_1, \cdots, y_T)$ and constant value of S, the calculation complexity would be $PR_S^{D,I,M} = \frac{S!}{D!I!M!}$, while using the forward algorithm the calculation complexity is on the order of $O(S \times \min\{T, D\} \times \min\{I, M\})$. Where D, I, and M are the number of delete, insertion, and match states in the hidden sequence, respectively. The backward algorithm starts from 'ee' state and terminates to 'bb' state and calculates $P_\lambda[\underline{Y}]$ and also is used for model training. The complexity of this algorithm is also equal to $O(S \times \min\{T, D\} \times \min\{I, M\})$.

4. MODEL TRAINING

For model training the words available in the black list are parameterized by means of several different forms of the original word. Each forbidden word has its own special λ parameters which results the highest probability by means of the HMM.
Determining the highest estimation probability from the λ vector parameters by solving the optimization problem of equation (3) is done for each model.

$$\begin{aligned} \underset{\lambda=(A,B,\underline{\pi})}{MAX} \quad & P_\lambda[Y_1 = y_1, \cdots, Y_T = y_T] \\ s.t. \quad & \sum_{j \in X} a_{ij} = 1, \quad i \in X \\ & \sum_{k \in Y} b_{jk} = 1, \quad j \in X \\ & \sum_{i \in X} \pi_i = 1 \\ & a_{ij}, b_{jk}, \pi_i \geq 0, \quad \forall i, j, k \end{aligned} \tag{3}$$

This problem is not a concave function and hence is considered a global function. The standard method which is adopted by the HMM to solve the problem is to use the local search method which is called "Baum-Welch algorithm" [8-10]. In this paper we also adopted the standard HMM approach with applying some improvements as well.

5. CLASSIFICATION WITH SPECIAL PARAMETERS

After model creation for each word of the black lists, the final goal is to classify the message either as a spam or non-spam which is done based on the occurrence number of the forbidden words inside the message context. With the aid of $\lambda_1, \ldots\ldots, \lambda_L$ models (each one of the models is related to a word available in the black lists) for the observed sequence $\underline{Y} = (y_1, \cdots, y_T)$ (which is either a word or part of a message) and based on the forward and backward algorithms the

value of dependency probability of observed sequence is achieved for each one of the models and this probability value is shown with q_l and is described as follows:

$$q_l = P_{\lambda_l} = [Y_1, \cdots, Y_T] \quad for \quad l = 1, \ldots, L \tag{4}$$

The threshold values $C_1, \ldots\ldots, C_L$ are constantly selected based on λ_L parameters for every model in a way that the selected threshold values are made equal with the minimum probability value required for observed sequence to be a member of the model.

The following rules are used for classification:

$$\text{if } \exists\, \mathrm{L} : \mathrm{q_L} > \mathrm{C_L} \rightarrow \ : \text{Classify } \ \mathrm{Y} \rightarrow \text{Forbiden}$$

If there is a model which presents a higher probability value for the observed sequence in comparison to the model threshold value, then the observed sequence is identified as a forbidden word and is labeled as an element of forbidden set, $\underline{F}$, for this email, otherwise it is identified as an unforbidden word:

$$\text{Otherwise} \ \rightarrow \ : \text{Classify } \ \mathrm{Y} \rightarrow \text{Unforbiden}$$

Finally, as an outstanding improvement versus other spam mail detecting methods using HMM, after evaluating all of the email words and extracting the whole elements of the forbidden set of $\underline{F}$, the spam probability of email, $\underline{P_{SE}}$, will be achieved based on the coefficients of the identified forbidden words, $coef\ (L)$, from equation (5).

$$\underline{P_{SE}} = \frac{\sum coef\,(L)_{\ where\ \ L \in \underline{F}}}{\text{Total number of email words}}, \quad \begin{cases} \underline{P_{SE}} \geq 0.5 \ \rightarrow spam \\ \underline{P_{SE}} < 0.5 \ \rightarrow non-spam \end{cases} \tag{5}$$

6. EXPERIMENTAL RESULTS

The proposed algorithm was realized on a computer with dual core 2.66GHz CPU in 2011 and applied and examined on experimental data including 5000 words which 100 words of them was various written forms of the word "درآمد", 100 words of them was various written forms of the word "سکس", 100 words of them was various written forms of the word "intercourse", 100 words of them was various written forms of the word "porn", 100 words out of them was various written forms of the "Penglish" word "jende", and the other 4500 were unforbidden words. The results are given in Table 1.

Table 1. The results obtained from the proposed algorithm.

	T_p	F_{FP} *	F_p	F_n	**Sensitivity value (%)**	**Identification value (%)**	**Precision value (%)**	**Error value (%)**
درآمد	97	1	2	13	88.1	99.95	99.65	0.347

سکس	99	0	1	10	90.8	99.97	99.76	0.239
intercourse	95	2	3	18	84	99.93	99.5	0.5
Porn	97	2	1	17	85	99.97	99.56	0.434
jende	96	3	1	12	81.35	99.97	99.65	0.347

***The number of forbidden words identified as another forbidden words**

The following components are used for evaluating the results.

$$\text{Sensitivity value} = \frac{T_p}{F_p + T_p}$$

$$\text{Identification value} = \frac{T_n}{F_n + T_n}$$

$$\text{Precision value} = \frac{C_c}{T_s}$$

$$\text{Error value} = \frac{M_c}{T_s}$$

where T_p is the number of correctly identified forbidden words, F_p is the number of unforbidden words identified incorrectly as forbidden words, T_n is the number of correctly identified unforbidden words, F_n is the number of forbidden words identified incorrectly as unforbidden words, C_c is the number of all of the words classified correctly, M_c is the number of all of the words classified incorrectly, and finally T_s is the number of all of the words. Based on the executed experiments, sensitivity value is better than 81.35 percent, identification value is better than 99.93 percent, precision value is better than 99.5 percent and error value has the least value of 0.5 percent.

7. CONCLUSION AND DISCUSSION

In this paper an efficient, adaptive, and powerful algorithm was presented for Persian, English and "Penglish" spam identification which is capable of identifying business and also immoral spams as an outstanding improvement of the proposed method. The proposed method was examined on a huge number of Persian, English and "Penglish" spams at research phase to prove its strength and capability. The black list is not created identically for all of the forbidden words, that is, an effective coefficient is defined for each forbidden words. The words included in the black lists as well as their effective coefficients are updated using the feedbacks received from the users which improves the identification precision substantially. According to the experimental results the proposed method achieved more than 99.5 percent precision, identifying the various types of spams.

REFERENCES

[1] Enrico Blanzieri and Anton Bryl, “A Survey of Learning-Based Techniques of Email Spam Filtering,” Conference on Email and Anti-Spam, 2008.

[2] Jose´ Gordillo and Eduardo Conde, "An HMM for detecting spam mail", Expert Systems with Applications, vol. 33, pp. 667–682, 2007.

[3] Seyyed Yasser Hashemi et.al. “Persian Spam Detecting Based on Hidden Markov Model,” Regional Conference on Scientific and Research Achievements in Electrical and Computer Engineering, Miyandoab Sama College, Miyandoab, Iran, pp. 97-103, September 2010.

[4] J. O. Berger, “Statistical decision theory: foundations concepts and methods,” New York: Springer Verlag, 1980.

[5] J. M. Bernardo and A. F. M. Smith, “Bayesian theory,” Chichester: John Wiley and Sons, 1994.

[6] T. Leonard, and J. S. J. Hsu, “Bayesian methods: an analysis for statisticians and interdisciplinary researchers,” Cambridge: Cambridge University Press, 2001.

[7] I. MacDonald and W. Zucchini, “Hidden Markov and other models for discrete-valued time series,” London: Chapman & Hall, 1997.

[8] L. R. Rabiner, “A tutorial on hidden Markov models and selected applications in speech recognition,” Proceedings of the IEEE, vol. 77, no. 2, pp. 257–286, 1989.

[9] L. R. Rabiner, and B. H. Juang, “Fundamentals of speech recognition,” New York: Prentice Hall, 1993.

[10] L. E. Baum, T. Petrie, G. Soules, and N. Weiss, “A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains,” The Annals of Mathematical Statistics, vol. 41, no. 1, pp. 164–171, 1970.

Authors

Seyyed Yasser Hashemi was born in Miyandoab, Azarbayjane Gharbi, Iran, in 1985. He received the B.Sc. and M.Sc. degrees from Islamic Azad University of South Tehran Branch, in Computer Engineering field. He is with Computer Department of Islamic Azad University, Miyandoab Branch since 2008. He is the author or coauthor of more than ten national and international papers and also collaborated in several research projects. His current research interests include voice and image processing, pattern recognition, spam detecting, optical character recognition, and parallel genetic algorithms.

Khalil Monfaredi was born in Miyandoab, Azarbayjane Gharbi, Iran, in 1979. He received the B.Sc. and M.Sc. degrees from Tabriz University in 2001 and Iran University of Science and Technology in 2003, respectively. He is with Electronic Research Center Group, since 2001 and is also an academic staff with Islamic Azad University, Miyandoab Branch since 2006. He serves as the Research and Educational Assistant of Miyandoab Sama College since 2009. He is the author or coauthor of more than twenty national and international papers and also collaborated in several research projects. He was also the chairman of 2010 electronic and computer scientific conference (ECSC2010) held in Islamic Azad University, Miyandoab Branch. He is currently pursuing his Ph.D. degree in Iran University of Science and Technology, Electrical and Electronic Engineering Department. His current research interests include voice and image processing, current mode integrated circuit design, low voltage, low power circuit and systems and analog microelectronics and data converters.