Filtering Spam E-Mail from Mixed Arabic and English Messages: A Comparison of Machine Learning Techniques

Alaa El-Halees
Faculty of information Technology, Islamic University of Gaza, Palestine

Abstract: *Spam is one of the main problems in emails communications. As the volume of non-english language spam increases, little work is done in this area. For example, in Arab world users receive spam written mostly in arabic, english or mixed Arabic and english. To filter this kind of messages, this research applied several machine learning techniques. Many researchers have used machine learning techniques to filter spam email messages. This study compared six supervised machine learning classifiers which are maximum entropy, decision trees, artificial neural nets, naïve bayes, support system machines and k-nearest neighbor. The experiments suggested that words in Arabic messages should be stemmed before applying classifier. In addition, in most cases, experiments showed that classifiers using feature selection techniques can achieve comparable or better performance than filters do not used them.*

Keywords: *Anti-spam filtering, machine learning techniques, text data mining.*

Received July 21, 2007; accepted October 23, 2007

1. Introduction

Spam email, also called unsolicited bulk email, or junk email, is an email that containing information that has been sent to a recipient who has not requested it [35] [2]. The problem of spam e-mail has been increasing for years. In recent statistics by [16], 40% of all emails are spam which about 12.4 billion email per day and that cost internet users about $255 million per year. To rule out spam automatically from user's email, the task of spam filter is used. Many works have been used to filter spam. These works used one of two approaches: the rule based method which generates a set of rules that classify an email as spam or legitimate such as work of [8]. The other approach is to use text mining or machine learning methods which consider spam filtering as two-class text classification that classifies a message as spam or legitimate. Many machine learning methods have been used in this approach such as Naïve Bayes (NB) [26, 17]; Decision Trees (DT) [25]; Maximum Entropy (ME) [36]; Neural Networks (NN) [6]; memory based approach [13] and Support System Machines (SSM) [12].

Spam filtering problem can be viewed as text classification. However, the difference between spam filtering and text classification is that email messages contain some form of identifiable textual content as meta-level features such as from, message date, to, and Subject.

Most of the work in anti-spam filters used English or european languages corpora with some exception such as work of [39] in Japanese; [29] in Chinese; [22] in turkish and some others. As the volume of non English language spam increases, little work is done in this area, for example in Arab word users receive spam written mostly in Arabic, English or mixed Arabic and English. In this paper we experimented well known machine learning methods to treat Arabic, English and mixed. These methods are ME, DT, Artificial Neural Nets (ANN), NB, System Vector Machines (SVM) and k-Nearest Neighbor (kNN).

The rest of the paper is organized as follows section 2 summaries related works in anti-spam filtering and classifying Arabic documents. Section 3 gives a general theoretical description on the six machine learning classifiers we used in this study. Section 4 proposes system that implements our approach. Section 5 reports our experiments of the proposed method and compares the results of the different classifiers. Finally we close this paper with a summary and an outlook for future work.

2. Related Work

There are some works in research compare different machine learning methods that filter anti-spam English email messages. For example, work of [11] who presented an empirical evaluation of four machine learning methods which are NB, Term Frequency-Inverse Document Frequency (TF-IDF), kNN and SVM. They found that NB and TF-IDF yield better performance than kNN. Another work is [21] who studied the applicability of some of the most popular

machine learning methods, which are NB, kNN, ANNs, SVMs in spam filtering. He found that NB is the best classifier. Also, he found that kNN has poor performance. In addition, [18] compared the performances of memory based learning, NB and SVM. They implemented the methods in a case-sensitive manner. Results of the experiments showed that SVM has significantly better performance for no-cost and high-cost cases. However, NB performed better in extremely cost cases. Also, [33] investigated the performance of two machine learning methods which are naïve bayesian and memory-based approach. They found that both methods achieved very high classification accuracy and outperformed word based methods. Also they found that if the mechanism of spam messages is flagged or informing the senders of blocked messages not available, memory-based approach appears to be more viable. Finally, [26] evaluated experimentally in spam filtering context five different versions of naïve bayes. They found that the flexible bayes and multinomial naïve bayes with boolean attributes obtained the best results.

As mentioned before, none of the above work tested in Arabic corpus, however, there are some works in classifying Arabic documents such as [10] that used naïve bayes algorithm to automatic Arabic document classification. Another system is called Siraj from Sakhr. The system is available at (siraj.sakhr.com) but it has no technical documentation to explain the method used in the system. The third work proposed by [9] who used statistical classification methods such as maximum entropy to classify and clusters News articles. In addition, [20] described a method based on maximum entropy to classify Arabic documents.

3. Machine Learning Classifiers

The problem of anti-spam filtering can be defined as follows [31] [33]: let set $D = \{d_1, d_2, \ldots. d_j\}$ is a set of email messages. In training phase , we train a Boolean function $\Phi_{spam}(d_j)$: $D \rightarrow \{True, False\}$ where $\Phi_{spam}(d_j)$ is true if the message d_j is spam, and false if d_j is legitimate, given the assumption that a document belongs to exactly one class (i.e., spam or legitimate). During testing phase, the function $\Phi_{spam}(d_j)$ is applied to new document d to predict where d is spam or not. The training and testing is done using one machine learning classifier. This study will compare mixed Arabic and English messages using the following machine learning classifiers:

3.1. Maximum Entropy

ME model estimates probabilities based on the principle of making as few assumptions as possible, other than the constrained imposed [30]. The constraints are derived from training process which expresses a relationship between the binary features and the outcome [15] [27]. Maximum entropy is a model which assigns a class c (i.e., spam or legitimate) of each word w based on its document d in the training data D. Conditional distributed $p(c|d)$ is computed as follows [27]:

$$p(c \mid d) = \frac{1}{Z(d)} \exp\left(\sum_i \alpha_i f_i(d, c)\right) \quad (1)$$

where $Z(d)$ is a normalization function which is computed as:

$$Z(d) = \sum_c \exp\left(\sum_i \alpha_i f_i(d, c)\right) \quad (2)$$

And the parameter α_i must be learned by estimation. It can be estimated by an iterative way using algorithms such as generalized Iterative Scaling (GIS) [4], Improved Iterative Scaling (IIS) [23], or L-BFGS algorithm [14]. In the equation, $f_i(d,c)$ is a binary valued feature which makes prediction about the outcome. Feature presented by each instance that will be classified. The type of feature could be either Boolean that presents if the word is in the text, or integer which presents frequency of the word in the text. In this work integer type is used because it gives more information than boolean. More precisely the feature can be formulated as [5]:

$$f_{(w,c')}(d,c) = \begin{cases} 0 & \text{if } c \neq c' \\ \dfrac{N(d,w)}{N(d)} & \text{otherwise} \end{cases} \quad (3)$$

where $N(d,w)$ is the number of times word w occurs in document d, and $N(d)$ is the number of words in d. [36] used ME to filter spam.

3.2. Naïve Bays

Naïve Bays (NB) is a probability-based approach. NB is used by [17] [26] [39] in anti-spam filtering. In NB, given a vector of words w_k, T the target classes which is either *spam* or *legitimate*. NB classifier defined as:

$$C_{NB} = \arg\max_{c_i \in T} p(c_i) \prod_k p(w_k \mid c_i)$$

(4) And $p(w_k \mid c_i)$ is the probability that a word w_k occurs in the email message which can be estimated as

$$p(w_k \mid c_i) = \frac{\text{Numbr of times word } w_k \text{ occurs in email mess}}{\text{Number of words in email message wit}}$$

(5)

3.3. Support Vector Machine

SVM is a learning algorithm proposed by [34]. It is 2-class classification method. As described by [15], *y* which classifies email message as spam or legitimate according to following dot product:

$$y=w.x-b \quad (6)$$

where *x* is a feature vector of email message composed of words. *w* is the weight of corresponding *x*. *b* is a bias parameter determined by training process.

3.4. k-Nearest Neighbor

kNN a very simple method to classify document. In training phase, email messages have to be indexed and convert to vector representation. To classify new message *x*, the similarity of its document vector to each document vector in the training set has be computed. Then its kNN is determent.If the majority of mssages among these neighbors are spam then the message is classified as spam. Otherwise it is classified as .[legitimate [32

3.5. Artificial Neural Networks

ANN is classification methods. Many researches used it for email filtering such as [28] and [19]. There are two kinds of ANNs; the perceptron and multilayer perceptron. In this research we used perceptron. In perceptron for anti-spam filtering, given a message *x*, find a liner function of the feature vector:

$$f(x)=w^T x+b \quad (7)$$

Such that $f(x)>0$ for vectors of one class (i.e., spam), and $f(x<0$ for vectors of other class (i.e., legitimate). Here $w=(w_1,w_2,...w_m)$ is a vector of coefficients of the function which represent the weights used in the ANN, and *b* is a bias.

3.6. Decision Trees

DT is classification and prediction technique used widely in data mining. C4.5 is a typical and effective decision tree method and it was used in some works to filter e-mail messages such as in [12] and [24]. In anti-spam email, a decision tree is a tree whose internal nodes are words of the email message and the leaf nodes are either spam or legitimate.

4. System Description

In this work, a system has been constructed to test the classifiers that classify email messages. The structure of the system is depicted in Figure 4. The system consists of two subsystems, one for training and the other for testing. The system has the following parts.

4.1. Corpus

All the classifiers we used in this study are supervised learning techniques which need training corpus. There are many ways to get datasets to train anti-spam filtering. The most common one is to use a public benchmark datatsets such as Ling-spam [13]. However, this study can not use any of these datasets because none of them contain Arabic corpus. So we have to look for another approach to get one which contains Arabic. To overcome this problem, we used personal emails with 1047 messages which contain 41883 tokens collected in 6 month period. Summary of collected corpora used in the experiment presented in Table 4.

The corpus is not as large as public datasets; however it is enough for our experiments using cross-validation as stated by [38]. We used English corpus which are messages contain mostly English text, and Arabic corpus which mostly contain Arabic text. The mixed corpus is datasets of the Arabic and English corpus. In training phase, the corpus manually classifieds as spam or legitimate.

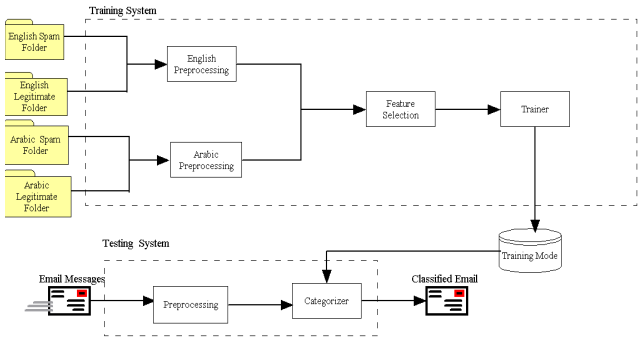


Figure 4. The system structure.

Table 4. Summary of Arabic and English corpora used in the experiments.

	Spam		
	No. Messages	**Vocabulary Size**	**Percentage**
Arabic	263	8923	49.1%
English	298	14582	61.5%
Total	561	23505	56.12%
	Legitimate		
	No. Messages	**Vocabulary Size**	**Percentage**
Arabic	268	9248	50.9%
English	218	9130	38.5%
Total	486	18378	43.88%

Considering privacy issue, we removed all data that identify the users, senders and recipients, as well as any persons, addresses or emails addresses available within the email body.

4.2. Preprocessor

Before applying machine learning methods, for both training and testing datasets, some preprocessing in the text are performed in both Arabic and English text. All the experiments are performed after preprocessing the text. In the preprocessing, the text is converted to UTF-8 encoded and punctuations and non- letters are removed. Then, the HTML and XML tags are striped out. Then text is parsed by a parser and all stopwords, in both Arabic and English, are removed. Stopwords are terms that are too frequent in the text. These terms are insignificant. So, removing them reduces the space of the items significantly. Then, for Arabic text, some Arabic letters are normalized such as آ ,إ ,أ, is converted to ا, and ى replaced by ي and ة to ه. Then, the roots are extracted from the text using a stemmer from AraMorph package, which is Arabic morphology Analysis package from http://www.qamus.org /morphology.htm. In English text all letters converted to lower case.

4.3. Feature Selection

In general, the size of the training corpus is very large. To reduce the high dimensionality of the words, feature selection was performed. In this case the features are the words to be trained in email messages. Feature selection usually used to reduce the size of the training corpus to an acceptable level. The benefit of feature selection also includes a small improvement in predication accuracy in some cases [36]. To select the most appropriate feature in the text message, the Mutual Information (MI) is computed for each word in the email message. MI is commonly used in language modeling such as spam filtering. The following formula is used to compute MI [36]:

$$MI(X,C) = \sum_{x\in\{0,1\},c\in spamlegitem} P(X=x,C=c)\times\log\frac{P(X=x,}{P(X=x)\times} \quad (8)$$

In this formula *MI* is computed for words *W* and class *C* where *C* denoted the class (spam or legitimate). *P (W=w,C=c)* is the probability that the word *W* occurs (*W=1*) or does not occur (*W= 0*) in Spam *(C= spam*) or legitimate (*C=legitimate*) email message and *P(W=w)* is the probability that the word *W* occurs or not in all emails messages, *P(C=c)* is the probability that an email is spam or legitimate. Then, the features with the *n* highest MI. score are selected [36]. We tested our experiments with *n* varies from 50 to 8000.

5. Experiments and Results

This section explains the set of experiments carried out to evaluate and compare the different machine learning classifiers discussed in section 3 that we used to classify spam messages. In all experiments, 10-fold cross-validation was employed. Each collection of messages in folder is divided almost equally to 10 parts; nine of them are used for training and one for testing. This process performed ten times for each experiment. The final result is the average of the ten iterations. This process produces more reliable results and used the entire corpus for both training and testing phases [38].

5.1. Performance Measures

To evaluate the spam filtering system, we computed the *recall* (the proportion of spam messages that are correctly classified as spam) and *precision* (the proportion of messages that are classified as spam that are, in fact, spam) which are generally accepted ways of measuring system's performance in this field [37]. The F1-measure is an average parameter based on precision and recall. These measurements computed as:

$$Recall = \frac{n_{S\to S}}{n_{S\to S}+n_{S\to L}} \quad (9)$$

$$Precision = \frac{n_{S\to S}}{n_{S\to S}+n_{L\to S}} \quad (10)$$

$$F1 = \frac{2*\mathrm{Re}\,call*\mathrm{Pr}\,ecision}{\mathrm{Re}\,call+\mathrm{Pr}\,ecision} \quad (11)$$

In these equations, *S* stands for spam message and *L* stands for legitimate message. We denote $n_{L\to L}$ as number of messages that correctly classified as legitimate messages, $n_{S\to S}$ number of messages correctly classified as spam messages. Also, $n_{L\to S}$ represents the number of legitimate messages which are misclassified as spam messages which also called false positive and $nS_{\to L}$ represents the number of spam messages which are misclassified as legitimate messages, this called false negative. In the previous measurements false positive and false negative are weighted equally. However, in reality misclassifying a legitimate email may be more harmful than

classification a spam email. For this reason, [2] introduced Weighted Accuracy (W_{Acc}) as follows:

$$W_{Acc} = \frac{\lambda . n_{L \to L} + n_{S \to S}}{\lambda . N_L + N_S} \quad (12)$$

In this equation N_L represented the number of legitimate messages and N_S is the number of spam messages. In this measurement misclassification of legitimate messages as spam messages is λ times more costly than misclassification of spam message as legitimate messages. In our experiments λ was 0.9.

5.2. Experiment 1

The first experiment was designed to compare the effectiveness of using machine learning classifiers in English, Arabic and mixed. The datasets used in these experiments are preprocessed except that no stemming or feature selection was used. Tables 5, 6 and 7 depicted the results.

Table 5. Appling machine learning classifiers to English corpus.

	W_{Acc} %	Recall %	Precision %	F1 %
ME	88.61	91.80	93.14	92.06
NB	92.12	78.15	98.75	86.56
DT	89.00	94.04	90.50	92.04
ANN	87.11	96.07	88.43	92.16
SVM	99.03	98.72	98.34	98.32
kNN	86.60	97.78	88.78	92.95

Table 5 represents the results of applying machine learning classifiers on only English datasets. It showed that SVM classifier gave the best W_{Acc} and F1-Measure. SVM gave the best recall. But, in precision naïve NB gave slightly better performance than SVM. Overall, SVM got the best performance in English Corpus. These results also showed that classifying English corpus has very good results, for example SVM has *WAcc* (99.03%) which is considered as a very good result. [18] also found that SVM has better performance in English corpus comparing with memory based learning and naïve bays.

Table 6. Appling machine learning classifiers to Arabic corpus.

	W_{Acc} %	Recall %	Precision %	F1 %
ME	82.55	79.04	68.78	73.43
NB	80.77	74.04	66.83	71.84
DT	78.02	78.52	74.25	68.00
ANN	89.77	89.21	74.69	76.25
SVM	78.83	87.03	77.59	76.51
kNN	72.253	90.09	71.55	75.54

Table 6 represents the results of using the classifiers to filter spam in Arabic Messages. ANN got the best *WAcc* results (i.e., 89.77%). The best recall (i.e., 90.09%) recorded by using kNN and the best precision (i.e., 77.59%) and F1-measure (i.e., 76.51%) recorded by SVM. The overall performance using all used matrices is much less accurate than using the same classifiers in English corpus. This is because Arabic is highly inflected language [3]. Therefore, in experiment 2 we will use stemming to stem the words before classifying them.

Table 7. Appling machine learning classifiers to Arabic and English corpora.

	W_{Acc} %	Recall %	Precision %	F1 %
ME	89.42	73.63	81.41	73.59
NB	78.54	81.75	63.43	76.57
DT	77.66	81.23	68.91	72.01
ANN	87.18	88.08	83.29	86.24
SVM	83.93	81.47	71.34	72.96
kNN	80.72	82.13	84.95	81.88

In Table 7 we measured the performance of the classifiers using mixed (Arabic and English) datasets. It appeared from the results that ME recorded the best W_{Acc} with 89.42%. In addition, ANN classifier has the best recall and precision. Hence it has the best F1-measure with 86.24%. From these results we can notice that the overall performance for all classifiers is better than using only Arabic Corpus and less for English.

5.3. Experiment 2

In experiment 1, we concluded that the overall performances of all classifiers in English datasets are much better than in Arabic datasets. To improve the performance, we stemmed the messages by converting each Arabic word in the message body to its roots. Table 8 depicted the results on mixed corpus.

Table 8. Applying machine learning classifiers to Arabic and English corpora with stemmed Arabic messages.

	W_{acc} %	Recall %	Precision %	F1 %
ME	92.92	90.65	90.04	90.01
NB	90.35	96.78	88.57	92.42
DT	82.78	95.33	80.75	80.36
AN	87.07	94.52	85.85	89.70
SVM	88.97	92.81	89.10	90.86
kNN	80.59	95.24	78.77	85.98

From the result we found that stemming significantly improved the results in most of the classifiers. ME still has the best W_{Acc} which increased from 89.42% with no stemming to 92.92% with stemming. Also, ME increased in precision from 81.41% to 90.04% which became the best precision in all used classifiers. NB has the best recall with 96.78% and F1-measure with 92.42%.

We also noticed that in ANN which has best recall, precision and F1-measure without stemming is not the case with stemming where the improvement is not as much as NB and ME.

5.4. Experiment 3

In this experiment, we used Mutual Information (MI)

Classifier	N	W_{Acc} %
ME	5000	94.31
NB	1000	90.35
DT	7000	82.78
ANN	1000	85.61
SVM	200	88.99
kNN	500	90.65

to select the most appropriate feature (i.e., words) in the text message.

Classifier	N	F1-Meaure %
ME	2000	91.01%
NB	1000	90.74
DT	3000	85.71
ANN	1000	91.74
SVM	2000	92.59
kNN	2000	89.01

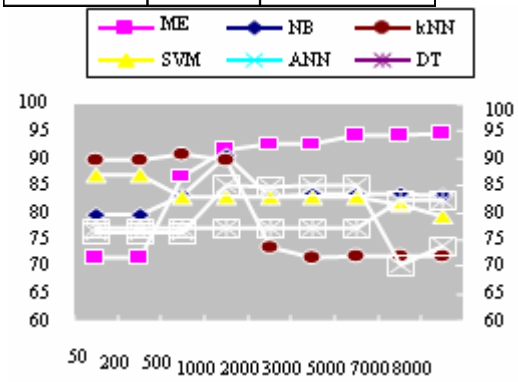


Figure 5. WAcc for machine learning classifiers using different feature sizes.

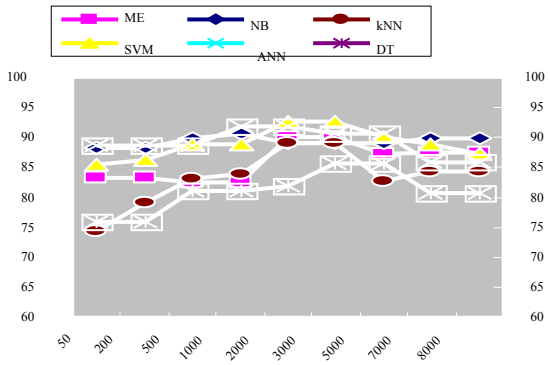
The MI is computed for each word in the email message. Then the features with the *n* highest MI-score were selected. We tested mixed Arabic and English datasets with *n* varying from 50 to 8000. Figure 5 depicts the W_{Acc} for each classifier. From the graph we can get the best *n* for each classifier which represented in Table 9. For example the best *n* for ME is *5000* with W_{Acc} (i.e., 94.31%) which is the best W_{Acc} in all classifiers.

Table 9. Best W_{Acc} and *n* in each classifier.

Comparing the results using feature selection and not using it, as in the previous experiment, we noticed that feature selection, beside reduces the size of the data, it increases the performance in W_{Acc} in all classifies except ANN which decreased from 87.07% without feature selection to 90.74% with feature selection.

Figure 6 depicted the F1-measure for all classifiers. It shows the best performance classifier is SVM which has F1-measure 92.59% with *n* is 2000.

Fig



ure 5. W_{Acc} for machine learning classifiers using different feature sizes.

Table 10. Best F1-meaure and *n* in each classifier.

Table 10 represents the best *n* for each classifier. We noticed the it increases the performance in F1-measure in all classifies except NB which decreased from 92.42%, which was the best F1-measure in all classifiers without feature selection, to 85.61 % with feature selection.

6. Conclusion

In this paper, we presented a system that filters spam from email messages in mixed Arabic and English corpus. To choose the best classifier, the system evaluated commonly used supervised machine learning methods. The methods we tried are: ME, DT, ANN, NB, SSM, kNN. Using personal email messages, in English only email messages, the performance of the system was acceptable where that best method was SVM with W_{Acc} 99.03% and F1-measure is 98.32%. In

Arabic only, the performance was not accurate where the best W_{Acc} was 89.77% by ANN and F1-measure 76.25% by SVM. We suggested that the reason of these results because that Arabic language is highly inflected language. Thus, we stemmed Arabic messages before classification and we got better performance by ME where W_{Acc} was 92.92% and by NB which has F1-measure of 92.42%. To reduce the training corpus we used mutual information feature selection method and we found that feature selection reduces the training data and sometimes increases the performance of the system.

For future work, a way to increase the performance of the filter for Arabic messages could be investigated. One way could be using other kinds of anti-spam methods such as statistical methods or combine more than one learning method. Also, the experiments in this paper used default parameters for the machine learning methods, changing the parameters could be another way to increase the performance of the system.

References

[1] Androutsopoulos I., Koutsias J., Chandrinos K., Paliouras G., and Spyropoulos C., "An Evaluation of Naïve Bayesian Anti-Spam Filtering," *in Proceedings of the Workshop on Machine Learning in the New Information Age 11th European Conference on Machine Learning (ECML 2000)*, pp. 9-17, Spain, 2000.

[2] Androutsopoulos I., Zaragoza H., Gallinari P., and Rajman M., "Learning to Filter Spam E-Mail: A Comparison of A Naïve Bayesian and A Memory Based Approach," *in the Proceedings of the 4th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD-2000)*, pp. 149-162, France, 2000.

[3] Apte C., Damerau F., and Weiss S., "Towards Language Independent Automated Learning of Text Categorization Models," *in Proceeding of Research and Development in Information Retrieval*, pp. 23-30, New York, 1994.

[4] Berger A., Pietra D., and Pierta D., "A Maximum Entropy Approach to Natural Language Processing," *Computer Journal Computational Linguistics*, vol. 22, no. 4, pp. 39-71, 1996.

[5] Berger A., The Improved Iterative Scaling Algorithm: A Gentle Introduction, *Technical Report*, 1997.

[6] Carreras X. and Marquez L. "Boosting Trees for Anti-Spam Email Filtering*," in the Proceedings of Recent Advances in NLP (RANLP-2001)*, pp. 58-64, Bulgaria, 2001.

[7] Chinchor N., "Named Entity Task Definition," *in Proceedings of the Seventh Message Understanding Conference*, pp. 137-142, US, 1998.

[8] Clark J., Koprinska I., and Poon J., "A Neural Network Based Approach to Automated E-Mail Classification," *in Proceeding of IEEE International Conference on Web Intelligence (WI'03)*, pp. 450-453, Hong Kong, 2003.

[9] Cohen P., *Empirical Methods for Artificial Intelligence*, MIT Press Cambridge, MA. 1995.

[10] Cohen W., "Learning Rules to Classify Email," *in Processing of the 1996 AAAI Spring Symposium on Machine Learning in Information Acces*s, Stanford, pp. 88-95, 1996.

[11] Cortes C. and Vapnik V., "Support-Vector Networks," *Machine Learning,* 1995.

[12] Darroch J. and Ratcliff D., "Generalized Iterative Scaling for Long_Linear Model," *Annals of Mathematical Statistics*, vol. 43, no. 5, pp. 1470-1480, 1972.

[13] Dasarathy B., *Nearest Neighbor Norms: NN Pattern Classification Techniques*, IEEE Computer Society Press, Uk, 1991.

[14] Diao Y., Lu H., and Wu D., "A Comparative Study of Classification Bases Personal E-Mail Filtering," *in the Proceedings of 4th Pacific-Asia Conference on Knowledge Discovery and Data Mining 2000* (*PAKDD-00*), pp. 62-73, 2000.

[15] Dong J., Cao H., Liu R., and Ren L.**,** "Bayesian Chinese Spam Filter Based on Crossed N-Gram**,"** *in Proceedings of the 6th International Conference on Intelligent Systems Design and Applications (ISDA'06)*, pp. 103-108**,** Shandong, 2006.

[16] Drucker H., Wu D., and Vapnik V., "Support Vector Machines for Spam Categorization," *IEEE Transactions on Neural Networks*, vol. 10, no. 5, pp. 1048-1054, 1999.

[17] El-Halees A., "Arabic Text Classification Using Maximum Entropy," *The Islamic University Journal*, vol. 15, no. 1, pp. 157-167, 2007.

[18] El-Kourdi M., Bensaid A., and Rachidi T., "Automatic Arabic Document Categorization Based on the Naïve Bayes Algorithm," *in the Proceeding of 20th International Conference on Computational Linguistics 28th*, pp. 1043-1050, Geneva, 2004.

[19] Eryigit G. and Tantuğ A., "A Comparison of Support Vector Machines, Memory-Based and Naïve Bayes Techniques on Spam Recognition," *in the Conference on Artificial Intelligence Applications (AIA-2005),* pp. 1-10, Australia, 2005.

[20] Evett D., "Spam Statistics 2006," http://spam-filter -review. toptenreviews.com/ spam-statistics .html, 2006.

[21] Hammo B., Abu-Salem H., Lytinen S., and Evens M., "Workshop on Computational Approaches to Semitic Languages*," in Proceeding of QARAB: A Question Answering*

System to Support the Arabic Language Workshop on Computational Approaches to Semitic Languages, pp. 55-65, Jordan, 2002.

[22] Iwanaga M., Tabata T., and Sakurai K., "Some Fitting of Naïve Bayesian Spam Filtering of Japanese,*" Workshop on Information Security Applications (WISA-2004*), Springer, Korea, 2004.

[23] Lai C. and Tsi M., "An Empirical Performance Comparison on Machine Learning Spam E-mail Categorization," *in Proceedings of 4th International Conference on Hybrid Intelligent Systems*, pp.44-48, Australia, 2004.

[24] Malouf R., "A Comparison of Algorithms for Maximum Entropy Parameter Estimation," *in Proceedings of the 6th Conference on Natural Language Learning (CoNLL-2002)*, pp. 49-55, Taiwan, 2002.

[25] Metsis V., Androutsopoulos I., and Paliouras G., "Spam Filtering with Naïve Bayes: Which Naïve Bayes," *in The 3rd Conference on Email and Anti-Spam CEAS 2006 Mountain View,* pp. 1702-1761, California, 2006.

[26] Nigam K., Lafferty J., and McCallum A., *Workshop on Machine Learning for Information Filtering*, pp. 61-67, UK, 1999.

[27] Ozgur L., Gungor T., and Gurgen F., "Adaptive Anti Spam Filtering for Agglutinative Languages: A Special Case for Turkish," *Pattern Recognition Letters*, vol. 25, no. 16, pp. 1819-1831. 2004.

[28] Sakkis G. and Androutsopoulos I., *A Memory: Based Approach to Anti-spam Filtering for Mailing Lists*, Kluwer Academic Publishers, London, 2003.

[29] Sawaf H., Zaplo J., and Ney H., *Arabic Natural Language Processing, Workshop on the ACL'2001*, France, 2001.

[30] Sculley D., Wachman G., and Brodley C., "Spam Filtering Using Inexact String Marching in Explicit Feature Space with on Line Classifiers," *in Proceedings of the 15th Text REtrieval Conference (TREC 2006)*, pp. 191-204, USA, 2006.

[31] Sebastiani F., "Machine Learning in Automated Text Categorization," *Computer Journal ACM Computing Surveys,* vol. 34, no. 1, pp. 1-47, 2002.

[32] Tretyakov K., "Machine Learning Techniques in Spam Filtering," *Technical Report*, 2004.

[33] Uchitmoto K., Ma Q., Murata M., Ozaku H., and Isahara H.,"Named Entity Extraction Based on a Maximum Entropy Model and Transformation Rules," *Journal of Natural Language Processing*, vol. 7, no. 2, pp. 63-90, 2000.

[34] Woitaszek M., Shaaban M., and Czernikowski R., "Identifying Junk Electronic Mail in Microsoft Outlook with a Support Vector Machine," *in the Proceedings of Symposium on Applications and the Internet (SAINT2003)*, pp. 166-171, Florida, 2003.

[35] Yang Y. and Pedersen J., "Comparative Study on Feature Selection in Text Categorization," *in Proceedings of ICML-97 14th International Conference on Machine Learning,* pp. 412-420, US, 1997.

[36] Youn S. and McLeod D., "A Comparative Study for Email Classification," *in Proceedings of International Joint Conferences on Computer Information System Sciences and Engineering (CISSE'06)*, pp. 462-567, USA, 2006.

[37] Zdziarski J, *Ending Spam: Bayesian Content Filtering and the Art of Statistical Language Classification 1st*, No Starch Press, 2005.

[38] Zhang L. and Yao T., "Filtering Junk Mail with a Maximum Entropy Model," in *the Proceedings of 20th International Conference on Computer Processing of Oriental Languages,* pp. 469-475, China, 2003.

[39] Zhang L., Zhu J., and Yao T., "An Evaluation of Statistical Spam Filtering Techniques," *ACM Transactions on Asian Language Information Processing (TALIP)*, vol. 3, no. 4, pp. 243-269, 2004.

Alaa El-Halees is an assistant professor in computing and dean of faculty of Information Technology Department at Islamic University of Gaza, Palestine. He holds PhD degree in data mining in 2004, MSc degree in software development in 1998 from Leeds Metropolitan University, UK. He received his BSc degree in computer engineering in 1989 from University of Arizona, USA. His research activities are in the area of data mining, in particular text mining, machine learning and e-learning.