

Article

E-mail Spam Filtering By a New Hybrid Feature Selection Method Using Chi2 as Filter and Random Tree as Wrapper

Seyed Mostafa Pourhashemi

Department of Computer, Dezful Branch, Islamic Azad University, Dezful, Iran
E-mail: s.mostafa.pourhashemi@gmail.com

Abstract. The purpose of this research is presenting a machine learning approach for enhancing the accuracy of automatic spam detecting and filtering and separating them from legitimate messages. In this regard, for reducing the error rate and increasing the efficiency, the hybrid architecture on feature selection has been used. Features used in these systems, are the body of text messages. Proposed system of this research has used the combination of two filtering models, Filter and Wrapper, with *Chi Squared* (Chi2) filter and *Random Tree* wrapper as feature selectors. In addition, *Multinomial Naïve Bayes* (MNB) classifier, *Discriminative Multinomial Naïve Bayes* (DMNB) classifier, *Support Vector Machine* (SVM) classifier and *Random Forest* classifier are used for classification. Finally, the output results of this classifiers and feature selection methods are examined and the best design is selected and it is compared with another similar works by considering different parameters. The optimal accuracy of the proposed system is evaluated equal to 99%.

Keywords: Feature Extraction, feature selection, classification, spam filtering, machine learning.

ENGINEERING JOURNAL Volume 18 Issue 3

Received 8 November 2013

Accepted 29 January 2014

Published 10 July 2014

Online at <http://www.engj.org/>

DOI:10.4186/ej.2014.18.3.123

1. Introduction

In Recent years, the massive rises in the Internet and low cost of E-mail have attracted a lot of attention of the most of advertisers of markets. As a result, receiving a high volume of unwanted messages which are increasing day by day, have become commonplace for users. These unwanted messages are called Spam [1]. Spams, in most cases are advertisements for advertising suspicious, plans for getting rich fast and seemingly legitimate services [2].

Spams are annoying for most of users, because not only beginning to diminish the reliability of e-mails, even users are affected by Spam due to the network bandwidth wasted receiving these messages and the time spent by users distinguishing between Spam and normal (legitimate) messages and damaging to the recipient system via malwares and viruses carried By spams [1].

Nowadays, There are many ways which designed to remove spam. This methods use different techniques for analyzing of E-mail and to specify that whether it is spam or legitimate mail.

Among all spam filtering approaches, Machine Learning technique has the best and high performance in spam classification. This method does not require any special rules. Instead, it needs many messages that nature of them (spam or legitimate) is identified, as training instances for the system. A special algorithm is used for training the system for finding the rules of message classification [3].

Ultimately, what we want to achieve is a spam filter which it can be represented as a f function which it specifies that received message m is spam or legitimate.

If we show all the received messages by M , Then we can say that we are looking for a function f defined by Eq. (1).

$$f: M \rightarrow \{S, L\} \quad (1)$$

Figure 1 shows an overview of a spam filter that is used in most modern filters which acts based on machine learning.

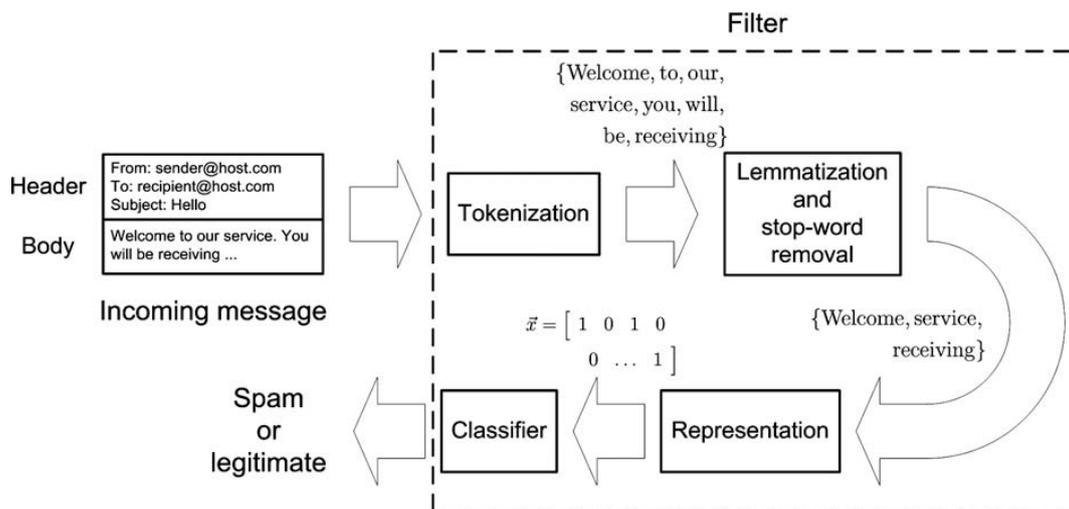


Fig. 1. An illustration of some of the main steps involved in a spam filter.

A brief description of the various parts of Fig. 1 is as follows:

- **Preprocessing:** At this phase, first all the words in the message are separated, then based on an preliminary analysis, Stop words like a-are-is-of... which do not help in classification, are separated among them and the remaining words use to determine that whether it can be a appropriate feature in classification or not, and these are sent to the next stage if these have the right conditions.
- **Feature Extraction and Selection:** In this section, Preprocessing phase output words, are examined based on some primary filter and the rules and conditions which designer Considers. Finally, specified number of words are selected as the main features. The selected features which are used in training the system and message classification, have important roles in the final performance of filter.

- Training the system: After selecting optimal features, we need to train the system. In this phase, from training instances, a database will be created based on optimal features, which the system is trained by it.
- Classification: In this phase, system decides whether or not it's spam, by checking the input message and based on the training that the system has been.
- Spam / Legitimate: Based on the final result of filter, Message is placed in the appropriate folder [4].

2. Theory

In the general case, the problem of spam filtering can be displayed as Eq. (2).

$$F(m, \theta) = \begin{cases} C_{spam}; & \text{if } m \text{ is spam} \\ C_{leg}; & \text{if } m \text{ is legitimate} \end{cases} \quad (2)$$

while m is the message should be classified, θ is vector of parameters, C_{spam} and C_{leg} are labels which are assigned to message. In most of spam filters which act based on machine learning, θ is the result of the training of classifier on pre-collected data set. Specifications of the whole system is introduced by Eq. (3).

$$M = \{(m_1, y_1), \dots, (m_n, y_n)\}, \quad y_i \in \{C_{spam}, C_{leg}\} \quad (3)$$

while m_1, m_2, \dots, m_n are marked as spam or legitimate by y_1, y_2, \dots, y_n labels, and θ is training function [5].

A filter which acts based on machine learning, uses a set of labeled data for training and analyzing (a set which previously collected and the judgment has been performed about them, whether they are spam or legitimate).

2.1. Related works

In reference [6], by using Sliding Window and appropriate method in counting of word frequencies on spam and legitimate messages, and using variance of event frequencies for feature selection and by using SVM & Naive Bayes classifiers, the performance reached to 96.8 %.

In reference [7], by using appropriate preprocessing based on clustering and using KNN (K-Nearest Neighbours) classifier, good results are obtained after classification.

In reference [8], the authors have developed a system called Filtron, in which by appropriate using of n-gram method and Information Gain (IG) and Black-White Lists and using by Flexible Bayes, good results are obtained with uni-word terms.

In reference [9], by using a hybrid feature selection system based on document frequency and IG method, and using Adaboost for classification has very good results, and the performance reached to 98.3 %.

In reference [10], by using a good translator for words which tokenizes them and removing the 100 more common words and by using the hybrid Naïve Bayes classifiers, good results are obtained.

In reference [11], by using hybrid semantic similarity and Adaptive Back Propagation Neural Network (ABPNN), very good results are obtained and performance reached to 98.6 %.

In reference [12], by using Logistic Regression (LR), new strategy presented which structure of any message stored in a Huffman tree, and for one message, one tree has been used. This method reached to 97.17 % for accuracy.

3. Methods

In this section, by considering mentioned topics in sections 1 & 2, we will describe proposed methods (included Preprocessing, Feature Extraction and Feature Selection by different algorithms, and used classifiers), and more we will review how to create and operation of spam filter which acts based on machine learning.

3.1. Preprocessing

The first phase should be done in order to create a filtering system, is Preprocessing. In this paper, we use the body of message which includes the main text of the message, for analysing messages.

The method which we have used to display features, is N -Gram with values $N=1,2,3$ which uni-word and dual-word and trey-word terms should be extracted among the body of text messages, to achieve this goal.

To determine that which features are useful for the system, the first thing to be done is preprocessing, that stop words which are not effective removed, and the words are tokenized (for example, elimination of *ing* & *ed* from end of verbs); as a result, the computational load of these features into the system, and the volume of preliminary information are reduced.

After the above preprocessing steps, we need a way to initialize the features. To do this, Term Frequency technique has been used. In this method, for each document, first, frequencies of each features are calculated and finally for the document, a vector is formed which included features with their frequencies [13]. The continuation of data mining is done by processing of these vectors.

3.2. Feature Selection

Feature selection is the most important phase in data mining and machine learning. Feature selection is used to reduce the main extracting data, to be improved both in terms of computational load and achieves the highest performance.

3.3. The Used Feature Selection Method in this Paper

We are dealing with a very large number of features, so for achieving the best result, we use hybrid feature selection method, which includes the methods which are handled in "Filter" approach. On the other hand, since all operations such as feature extraction and feature selection and finally classification not to be performed in parallel, we need to use of "Wrapper" method, while advantages of wrapper method also cannot be ignored.

Filter model selects features based on separate specifications of features and well-being of a feature. Wrapper model performs feature selection by using a classification algorithm (like Decision Tree), and it uses the high degree of efficiency as a metric to select the features. Hybrid model is a new method which uses advantages of Filter model and Wrapper model simultaneously. Independent tests are implemented on information and also function evaluation selects output subset [14]. The proposed process is shown in Fig. 2.



Fig. 2. Process of implementation and filtering in proposed method.

As it can be seen in Fig. 2, first primary data enters into filter 1. In this filter, Stop words, worthless words and Tokens are removed, it makes the original data size is somewhat reduced.

In filter 2, we use a filter which acts based on wrapper method, and it has more precision than filter 1. This filter is used to decrease the features, to find the optimal subset and to increase the performance of classifier. In classification phase, four classifiers (DMNB, MNB, SVM and Random Forst) are used that the output results of this filter and the results of reviewed classifiers will be presented in section 4.

3.3.1. Filter 1

The overall messages placed at filter1 as text documents and this filter uses Bag of Words (BoW) to show words per document. Term Frequency method is used to extract the words and to recognize the usefulness of them, that the frequency of each word per document is calculated and the features which are repeated lower that a threshold, will be removed.

Then, we should separate more useful features by special techniques. First we should calculate and analyze frequencies of each word in spam class and legitimate class separately. So we change the method of calculating the number of occurrences by defining two new parameters (according to Eq. (4)) for each feature.

$$C_{s,x} = \frac{N_{s,x}}{N_s} \quad , \quad C_{h,x} = \frac{N_{h,x}}{N_h} \quad (4)$$

The $C_{h,x}$ and $C_{s,x}$ parameters are calculated for each features. In above-mentioned equation, N_h and N_s represent the total number of legitimate (ham) messages and spam respectively. $N_{h,x}$ is equal by total number of documents which contain x , and that message are one of legitimate messages. $N_{s,x}$ is equal by total number of documents which contain x , and that message are one of spam messages. After calculating the above values, and by considering a threshold, we can check the features.

For a feature, if $C_{h,x}$ and $C_{s,x}$ parameters are very close together, then it represents that feature is distributed in spam & legitimate messages equally, thus it cannot be a good feature for separating both spam and legitimate classes. If $C_{h,x}$ and $C_{s,x}$ parameters have an appropriate difference (threshold) together, so the feature is repeated in one of classes more, and recognition of two classes can be done by the feature.

Then we have used Chi Squared technique as feature selector in filter 1. Chi Squared acts based on a statistical value called χ^2 [15].

3.3.2. Filter 2 (Wrapper)

Wrapper has important role in identifying spams using proposed methods, due to the high performance of classifier and selecting optimal subset. In this paper, we have used *Random Tree* [16] which performs based on wrapper.

3.4. Performance Evaluation of Classifier

For evaluating the performance of a classifier, there are two categories of indicators, Information Retrieval and Decision Theory. But another problem that should be noted in evaluating a classifier is the costs for messages are being incorrectly classified. Accordingly, accuracy parameter cannot be suitable for evaluating classifier solely.

In the field of decision theory, if we consider spam class as Positive class, then *TP* and *TN* parameters based on Eq. (5) and (6) can be defined.

$$\eta_p = \frac{n_{s,s}}{n_s} \quad (5)$$

$$\eta_m = \frac{n_{L,L}}{n_L} \quad (6)$$

While n_s is the total number of spams in data set, and n_L is the total number of legitimates in data set. $n_{s,s}$ is total number of spams which are correctly diagnosed, and $n_{L,L}$ is total number of legitimates which are correctly diagnosed.

In the field of information retrieval, classification be tested based on Precision & Recall parameters. Precision parameter represents the total number of positive class instances that are correctly classified to the total number of instances which have been diagnosed as positive. Recall parameter represents the total number of positive class instances that are correctly classified to the total number of instances. Precision & Recall parameters are shown in Eq. (7) and (8) for spam class.

$$P_s = \frac{n_{s,s}}{n_{s,s} + n_{L,s}} \quad (7)$$

$$r_s = \frac{n_{s,s}}{n_{s,s} + n_{s,L}} \quad (8)$$

By combining Precision & Recall parameters, another parameter is defined, called F_β which β is determined for exactitude. The value of β has been equal to 1 for most of the previous works. How to calculate the F_β parameter is shown in Eq. (9).

$$F_\beta = (1 + \beta^2) \frac{r_s P_s}{\beta^2 p_s + r_s} \quad (9)$$

In the proposed method, the value of β has been selected equal to 1.

4. Results and Discussions

In this section, how to implement the tools has been described, and then the output of the proposed method is presented, and finally, the results are compared with some of similar previous works.

To implement different parts of the designed system, we have used MATLAB version 7.14 for feature extracting and above-mentioned preprocessings and we have used updated version of Weka (version 3.7.9) for used filters and classifications.

4.1. Used Data Set

Each machine learning system requires a training set to train the system. In this paper, we have used LingSpam [17], as standard data set, including 2893 text messages which 2412 messages (about 83.37 %) are legitimate and 481 messages (about 16.63 %) are spam. In this data set, all of HTML tags and headers except Subject have been removed. We have used the third version of this data set. In test phase, we have used this data set (LingSpam data set) again in 10-folds cross validation mode. So we have used the LingSpam data set on both of training and testing phases.

In following, two messages of this dataset, are given as examples:

Message 1: Legitimate Message {Subject: job - university of utah.
the linguistics program at the university of utah invites applications for a one-year visiting assistant professor position to begin september , 1992 . minimum degree requirement is an a . b . d . candidates will be expected to teach an introductory undergraduate linguistics course and a course in american english for english teaching majors . they will also propose other undergraduate or m . a . level courses in general linguistics and sociolinguistics . send letter of application , curriculum vitae , sample publications , and three letters of reference to mauricio mixco , director , linguistics program , stewart building 213 , university of utah , salt lake city , ut 84112 . for further information you may telephone : 801-581 - 7432 or email dipaolo @ anthro . utah . edu .}

Message 2: Spam Message {Subject: please read .
payless typing service 1624 n . robberson springfield , mo 65803 (417) 865-7096 moongoddes @ wwn . net to whom it may concern , friday , may 1 , 1998 greetings ! my name is vania . i have recently started a home typing service . i am writing this letter not only to introduce myself , but to offer my services in the above area as well . included in this area of business are the following : typing / mailing , data entry , records / bookkeeping , billing , and various other services depending on you ' re individual needs . i am anxious to get started so please do n't hesitate to contact me . sincerely , vania juarez }

4.2. Separation the Words and Features

Features are the most important part of each machine learning problem. In this paper, features are terms within text messages which should be extracted from body of text messages. To extract desired words, space character has been used as separator. In Table 1, the number of extracted features for words of length 1, 2, 3 are shown.

Table 1. Extracted features.

Length of terms	The number of extracted features
Uni-word	62089
Dual-word	125396
Trey-word	170341

For accurate study and better test of the proposed method, we have lengths of terms in this research between uni-word and trey-word.

4.3. Feature Selection Based on Filter

Based on the got features in Table 1, it is necessary to eliminate redundant features. To do this, first the features have been studied by filter 1 described in the previous section. Results of the filter, are reduced set of features, which are reported in Table 2. Then the output of the filter will be given to filter 2.

Table 2. The output results of filter 1.

Length of terms	The number of features after applying the Filter 1
Uni-word	1540
Dual-word	1942
Trey-word	2209

Feature reduction is done in filter 1, as mentioned in the previous section also, by considering a threshold and how to repeat the features in spam and legitimate messages. Because, with increasing length of the term, frequency of features will be changed in data set also, in this phase we have used different thresholds for different lengths of terms.

In this research, we have used *Chi Squared* (Chi2) as filter 1. The output of the filter is given to four classifiers *Multinomial Naïve Bayes* (MNB) [18, 19], *Discriminative Multinomial Naïve Bayes* (DMNB) [20, 21], *Support Vector Machine* (SVM) [22, 23] with normalized poly kernel and *Random Forest* [16, 24] with 100 random trees. The feature set which has higher accuracy for the most of classifiers, is sent to filter 2 and in this filter, number of features is reduced and the decreased feature set is sent to classifiers. It should be noted that all of classifications are done in 10-fold cross validation.

Figures 3, 4 and 5 show the accuracy of classifiers considering the Chi2 filter for uni-word, dual word and trey-word features.

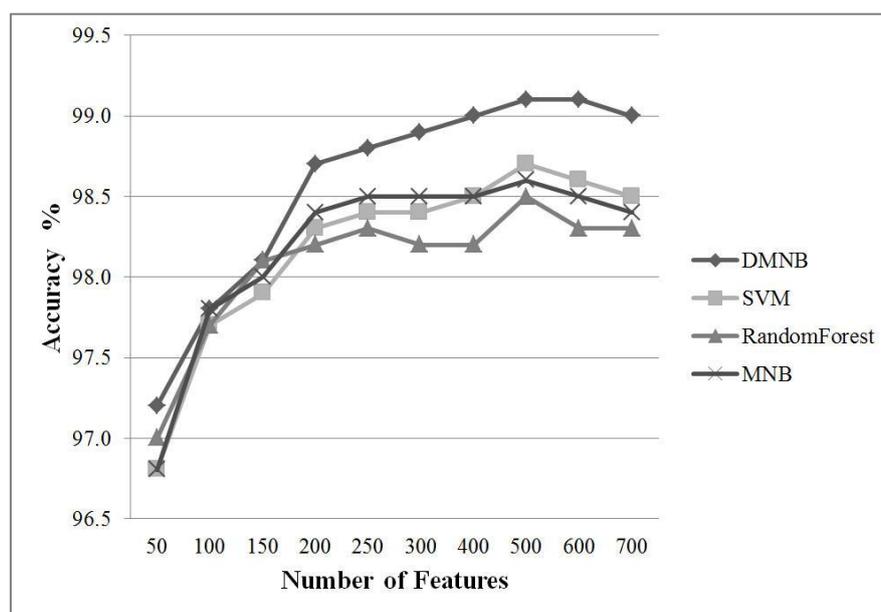


Fig. 3. The results for Chi2 filter and uni-word terms.

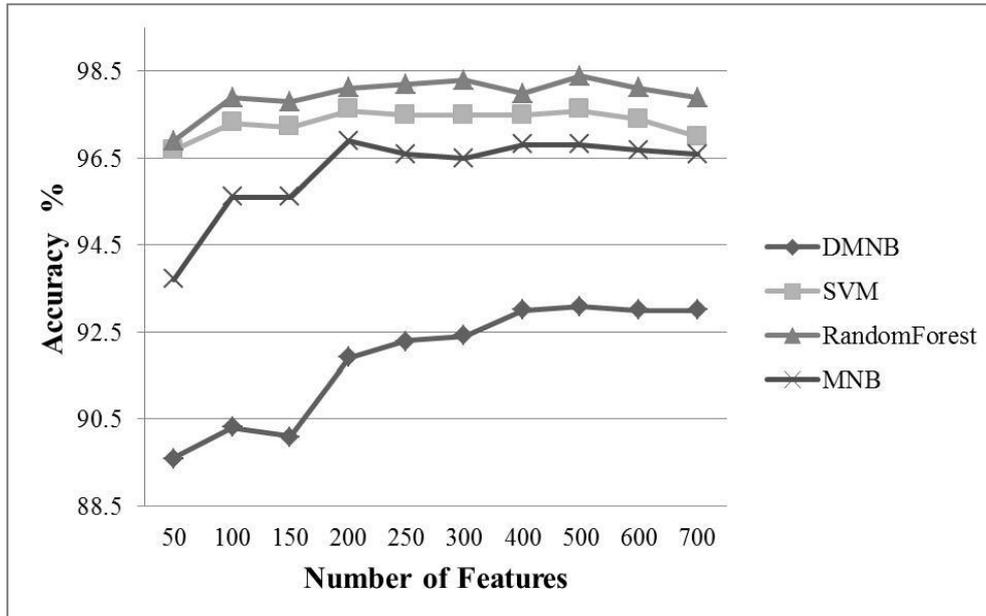


Fig. 4. The results for Chi2 filter and dual-word terms.

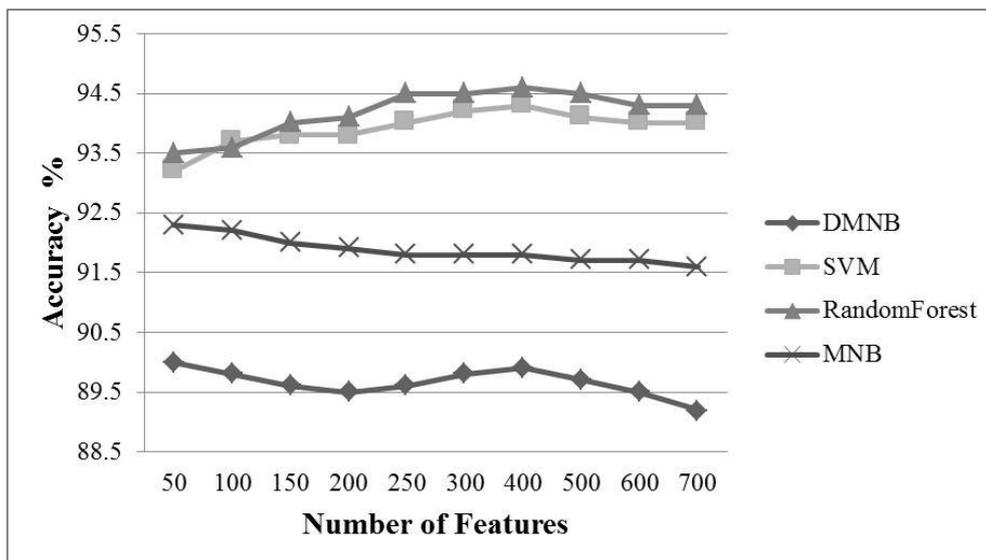


Fig. 5. The results for Chi2 filter and trey-word terms.

After applying the Chi2 filter, the set of best features that are capable to produce the highest accuracy is shown in Table 3.

Table 3. The number of optimal features.

Length of terms	Chi2
Uni-word	500
Dual-word	500
Trey-word	400

The number of optimal features which are shown in Table 3, is based on best output results for the feature selection algorithm and the specific terms. When we use uni-word features, classifiers show higher accuracy; this represents that uni-word terms have higher power for classification. By identifying the appropriate feature set at this phase, the new feature set is sent to filter 2 for finding the final optimal feature set.

4.4. Feature Selection by Applying Wrapper

In filter 2 and by applying wrapper model, we find the final feature set. In this phase, we have used Random Tree for wrapper and the results are compared. Table 4 represents the number of final features.

Table 4. The number of final selected features by applying Random Tree for wrapper.

Length of terms	Chi2
Uni-word	28
Dual-word	32
Trey-word	21

According to the above table, after applying filter 2, the number of final optimal selected features in all of cases is different by the case that only filter 1 was applied.

4.5. The Output Results of the Proposed Method

In this system, uni-word features produced better results. The accuracy of this hybrid feature selection method for all of four studied classifiers is shown in Table 5, with time of classification.

Table 5. The accuracy of classifiers with time of classification.

Classifier	Accuracy (%)	Time of Classification (s)
DMNB	98.24	1.02
SVM	98.69	9.93
Random Forest	98.62	24.43
MNB	98.44	0.95

We consider the case which has most accuracy and precision on messages diagnosis, as proposed method and the output results are shown in Table 6.

Table 6. The output results of proposed method.

Classifier	Accuracy (%)	Precision (%)	Recall (%)	F1 (%)	FP	Time (s)
SVM	98.69	98.7	98.7	98.7	8	9.93

According to the Table 6, Recall parameter for proposed method is equal to 99.7%, that represents a few number of spams which have been wrongly diagnosed as legitimates, and the Precision parameter is equal to 99.7%, that represents a few number of legitimates which have been wrongly diagnosed as spams. False Positive (FP) parameter is equal to 8, which represents only 8 messages of 2412 legitimate messages have been wrongly diagnosed as spams. By considering output results, it can be seen that proposed method is shown very good performance.

4.6. Performance Comparison of the Proposed Method with Other References

In Table 7, the proposed method is compared with all of above-mentioned methods (told in section 2), using accuracy parameter.

Table 7. Comparison of the proposed method based on accuracy parameter.

Accuracy (%)	
Proposed Method	98.69
Reference [6]	96.80
Reference [7]	94.40
Reference [8]	95.42
Reference [9]	98.30
Reference [10]	97.06
Reference [11]	98.60
Reference [12]	97.17

In Table 8, the proposed method is compared with some other methods, using different parameters. Notice that, the training and testing data set of all of following references, all similar to our dataset. It means that all of them have used LingSpam data set on training and testing phases, same to us.

Table 8. Comparison of the proposed method based on different parameters.

	Accuracy (%)	Precision (%)	Recall (%)
Proposed Method	98.69	98.7	98.7
Reference [6]	96.80	93.73	98.1
Reference [7]	94.40	91.1	97.6
Reference [8]	95.42	94.95	91.43
Reference [9]	98.30	98.3	98.3

Amount of difference between proposed method and other references is compared, and amount of performance improvement is shown in Table 9.

Table 9. Amount of improvement of proposed method in comparison with other references.

	Amount of accuracy improvement (%)	Amount of precision improvement (%)	Amount of recall improvement (%)
Reference [6]	+ 1.89	+ 4.97	+ 0.6
Reference [7]	+ 4.29	+ 7.6	+ 1.1
Reference [8]	+ 3.27	+ 3.75	+ 7.27
Reference [9]	+ 0.39	+ 0.4	+ 0.4

5. Conclusions and future works

The purpose of this paper is designing and presenting a machine learning system to increase the performance for automatic diagnosing and filtering spam messages from legitimate messages.

First, we attempted to separate and to extract uni-word, dual-word and trey-word terms by considering the body of text messages. These terms are the features which messages can be judged by them, at next phases. For Appropriate judgment about a message, we should select the best features among all of extracted features; so, in continue, we enter the next phase called Feature Selection, which is done by two filters. In filter 1, after eliminating the stop words which are not effective and tokenizing the words, we calculated the frequencies of each features in spam and legitimate message categories, then we deleted the features which have repeated lower than a threshold. In filter 2, we selected optimal set among reduced

feature set, using learning algorithms (the combination of filter and wrapper). The performance of used classifiers is one of parameters which help in selecting optimal subset.

Output results of each classifiers and feature selection approaches which used in this paper, was noted in section 4, the performance of designed system was evaluated, the best design was selected and it was compared considering different parameters. Finally, what can be concluded about the designed system, it is that the combination of filter and wrapper methods in feature selection and the use of appropriate classifier can has very good performance in data mining issues.

For future work we will focus on Ontology. The combination of semantic ontologies in feature selection phase can be used to improve classifier performance. In this paper, we used body of messages for decision making; we can use another characteristic like Sender address, Recipient address and Size of message also. And also we can generalize our proposed method on another data sets used for spam filtering (like multi-language datasets), and another data sets used for another topics based on text processing (like web classification) and finally we can test them and observe the results.

References

- [1] I. Androustopoulos, J. Koutsias, K. V. Chandrinou, and C. D. Spyropoulos, "An experimental comparison of naive Bayesian and keyword-based anti-spam filtering with personal e-mail messages," in *Proceedings of the 23rd annual international ACM SIGIR conference on Research and Development in Information Retrieval*, New York, NY, USA, 2000, pp. 160-167.
- [2] Spam Abuse Corporation. [Online]. Available: <http://spam.abuse.net/overview/whatisspam.shtml> [Accessed: 2013].
- [3] K. Tretyakov, "Machine learning techniques in spam filtering," in *Data Mining Problem-Oriented Seminar*, 2004, pp. 62-79.
- [4] T. S. Guzella and W. M. Cominhas, "A review of machine learning approaches to spam filtering," *Expert Systems with Applications*, vol. 36, pp. 10206-10222, 2009.
- [5] E. Blanzieri and A. Bryl, "A survey of learning-based techniques of email spam filtering," *Artificial Intelligence Review*, 2008 March, vol. 29, no. 1, pp. 63-92, Mar. 2008.
- [6] Y. Zhu and Y. Tan, "A local-concentration-based feature extraction approach for spam filtering," *IEEE Transactions on Information Forencics and Security*, vol. 6, pp. 486-497, Jun. 2011.
- [7] M. Besavaraju and R. Prabhakar, "A novel method of spam mail detection using text based clustering approach," *International Journal of Computer Applications (IJCA)*, vol. 5, pp. 15-25, Aug. 2010.
- [8] E. Michelakis, I. Androustopoulos, G. Paliouras, G. Sakkis, and P. Stamatopoulos, "Filtron: A learning-based anti-spam filter," in *Proceedings on First Conference on Email and Anti-Spam (CEAS)*, California, USA, Jul. 2004.
- [9] A. Beiranvand, A. Osareh, and B. Shadgar, "Spam filtering by using a compound method of feature selection," *Journal of Academic and Applied Studies (JAAS)*, vol. 2, pp. 25-31, Mar. 2012.
- [10] I. Androustopoulos, J. Koutsias, K. V. Chandrinou, G. Paliouras, and C. D. Spyropoulos, "An evaluation of naive bayesian anti-spam filtering," in *Proceedings of the Workshop on Machine Learning in the New Information Age, 11th European Conference on Machine Learning*, Barcelona, Spain, Jun. 2000, pp. 9-17.
- [11] C. H. Li and J. X. Huang, "Spam filtering using semantic similarity approach and adaptive BPNN," *Neurocomputing*, vol. 92, pp. 88-97, Sept. 2012.
- [12] Y. Zhou, M. S. Mulekar, and P. Nerellapalli, "Adaptive spam filtering using dynamic feature spaces," in *Proceedings of the 17th IEEE International Conference on Tools with Artificial Intelligence*, Hong Kong, Washington, DC, USA, 2005, pp. 302-309.
- [13] M. Chang and C. K. Poon, "Using phrases as features in email classification," *The Journal of Systems and Software*, vol. 82, pp. 1036-1045, Jun. 2009.
- [14] X. Geng, T. Y. Liu, T. Qin, and H. Li, "Feature selection for ranking," in *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, New York, NY, USA, 2007, pp. 407-414.
- [15] H. Liu and R. Setiono, "Chi2: Feature selection and discretization of numeric attributes," in *Proceedings of the IEEE 7th International Conference on Tools with Artificial Intelligence*, Washington, DC, USA, 5-8 Nov. 1995, pp. 388-391.
- [16] L. Breiman, "Random forests," *Journal of Machine Learning*, vol. 45, pp. 5-32, Oct. 2001.
- [17] LingSpam Public Corpus. [Online]. Available: <http://www.aueb.gr/users/ion/publications.html> [Accessed: 2013].

- [18] A. M. Kibriya, E. Frank, B. Pfahringer, and G. Holmes, "Multinomial naive Bayes for text categorization revisited," in *Proceedings of 17th Australian Joint Conference on Artificial Intelligence*, Cairns, Australia, Dec. 2004, vol. 3339, pp. 488-499.
- [19] K. M. Schneider, "A comparison of event models for naive Bayes anti-spam e-mail filtering," in *EACL'03 Proceedings of the Tenth Conference on European Chapter of the Association for Computational Linguistics*, PA, USA, 2003, vol. 1, pp. 307-314.
- [20] M. Hall, "Discriminative multinomial naive Bayes for text classification," Community Contribution: Pentaho Data Mining-Weka/DATAMINING-125, Jun. 2008.
- [21] J. Su, H. Zhang, C. X. Ling, and S. Matwin, "Discriminative parameter learning for Bayesian networks," in *Proceedings of the 25th International Conference on Machine learning (ICML)*, New York, NY, USA, 2008, pp. 1016-1023.
- [22] E. Alpaydin, *Introduction to Machine Learning*, 2nd ed. The MIT Press, Feb. 2010, pp. 350-380.
- [23] S. S. Keerthi, S. K. Shevade, C. Bhattacharyya, and K. R. K. Murthy, "Improvements to Platt's SMO algorithm for SVM classifier design," *Neural Computation*, vol. 13, no. 3, pp. 637-649, 2001.
- [24] G. Biau, L. Devroye, and G. Lugosi, "Consistency of random forests and other averaging classifiers," *Journal of Machine Learning Research (JMLR)*, vol. 9, pp. 2015-2033, 2008.