A Novel Approach Toward Spam Detection Based on Iterative Patterns

Mohammad Razmara, Babak Asadi, Masoud Narouei, Mansour Ahmadi
Computer Engineering Department
Islamic Azad University
Arak, Iran
razmara777@gmail.com, babakmz2002@yahoo.com, naroee@cse.shirazu.ac.ir, info@mahmadi.com

Abstract — **Spamming is becoming a major threat that negatively impacts the usability of e-mail. Although lots of techniques have been proposed for detecting and blocking spam messages, Spammers still spread spam e-mails for different purposes such as advertising, phishing, adult and other purposes and there is not any complete solution for this problem. In this work we present a novel solution toward spam filtering by using a new set of features for classification models. These features are the sequential unique and closed patterns which are extracted from the content of messages. After applying a term selection method, we show that these features have good performance in classifying spam messages from legitimate messages. The achieved results on 6 different datasets show the effectiveness of our proposed method compared to close similar methods. We outperform the accuracy near +2% compared to related state of arts. In addition our method is resilient against injecting irrelevant and bothersome words.**

Keywords – Spam Detection; Classification; Iterative Patterns; Text Mining

I. Introduction

Email as a low cost communication tool is broadly used by the direct marketers for exchanging information. Because sending email costs very low, one could send hundreds or even thousands of malicious email messages each day over internet connection. These junk emails, referred to as spam, reduce staff productivity, consume significant network bandwidth etc. In many cases such messages also contain viruses, spyware and inappropriate contents that can create legal/compliance issues, loss of personal information and corporate assets. Therefore it is important to accurately evaluate the effectiveness of countermeasures such as spam filtering tools. Spamming is a big challenge toward organizations, internet consumers, and service providers today. Email spamming, also known as Unsolicited Bulk Email (UBE) or Unsolicited Commercial Email (UCE), is sending unwanted email messages frequently in large quantities to an indiscriminate set of recipients [1]. It is becoming a serious problem for the internet community. A broad array of products is designed to stop or reduce the large amount of spam which come into individuals' emails. These products use techniques, implemented in various ways such as origin-based filters which are based on using network information and IP addresses in order to detect whether a message is spam or not. The most common techniques are filtering techniques. The introduction of technologies, such as Artificial Neural Network (ANN), support vector machines (SVMs), Bayesian filtering, Artificial Immune system (AIS), etc, which Attempt to identify whether a message is spam or not based on the content and other characteristics of the message can improve the accuracy of spam filters. The implementation of these machine learning algorithms is very important in the continuous fight against spam. In spite of the large number of methods and techniques available to combat spam, the volume of spam on the internet is still rising. This is the first work, based on best of our knowledge, to show the effectiveness of iterative patterns as a mean for spam filtering. Iterative patterns are sequences that occur in a sentence more than a predefined support or frequency.

In this paper, we extract the iterative patterns, which are formally defined in next section, from text. The extracted patterns are used as features and the problem is converted to a regular supervised learning. Results of our algorithm on standard labeled datasets shows promising results.

Rest of the paper is as follows: in section two the definitions are presented. Related works is described in section 3. Section 4 presents the proposed method. The experimental results are presented in section 5. Finally, conclusions and future work wraps up the paper.

II. Definitions

Definition (1) - Itemset: Let $\Gamma = \{l_1, l_2, ..., l_m\}$ be a set of items. A subset of Γ is called an itemset. An itemset that contains k items is a k-itemset. The occurrence frequency of an itemset is the number of transactions that contain the itemset. This is also known, simply, as the frequency of the itemset. Let D be a set of database transactions $D = \{T_1, T_2, ..., T_m\}$ where each transaction T_i is a nonempty itemset such that $T \subseteq \Gamma$.

Definition (2) - Frequent Itemset: An itemset T is frequent for a transaction dataset D if $\frac{|D_T|}{|D|} \geq \alpha$, where $\frac{|D_T|}{|D|}$ is called the support of T in D, written s (T), and α is the minimum support threshold, $0 \leq \alpha \leq 1$.

978-1-4673-4476-0/12/$31.00 ©2012 IEEE

Definition (3) - Closed Frequent Itemset: An itemset T is closed in a dataset D if there exists no super-itemset Γ such that Γ has the same support count as X in D. An itemset T is a closed frequent itemset in set D if T is both closed and frequent in D.

Definition (4) - Iterative Pattern Instance: Given a pattern $P\left\langle e_1,e_2,...,e_n \right\rangle$, a consecutive series of words $SB\left(sb_1,sb_2,...,sb_m\right)$ in a sentence (word sequence) S in WSD database (WSDDB) is an instance of P iff it is of the following Quantified regular expression (QRE):

$$e_1;\left[-e_1,...,e_n\right]*;e_2;...;\left[-e_1,...,e_n\right]*;e_n$$

QRE resemble to standard regular expression with ';' as the concatenation operator, '[-]' as the exclusion operator (e.g., [-P, S] means any event except P and S), and '*' as the standard Kleene star.

Definition (5) - Frequent Iterative Pattern: An iterative pattern P is frequent if its instances occur above a certain threshold of minimum support in WSDDB.

Considering the following Example:

SPAM #1:

Department tesl & apply linguistic **university californium** , lo angele announce open tenure-track position , rank determine , discourse analysis (pend final budgetary approval). Appointee participate propose interdisciplinary teach program language , interaction , culture . Candidate display strong research teach record ( ) interface conversation culture, (ius) integration visual verbal resource construction mean , (iius) expertise technology analyze discourse society . Candidate must ph . d . Hand application. **Application** must receive january 15 , 1995 include letter , vita , three letter reference , representative publication . send application : chair , search committee , **department** tesl & apply linguistic , 3300 rolfe hall , ucla , lo angele , ca 90024-1531 . ucla **affirmative** action , equal opportunity employer . Women member under represent minority encourage apply.

SPAM #2:

announcement open rank professorial position **university California** , san diego department linguistics subject availability fund , department linguistic university californium , sandiego , seek fill open rank professorial position (tenure / tenure - track) effective July 1 , 1995 , linguist capable teach formal semantics prove research record formal semantics , include semantics / syntax interface . Salary commensurate rank experience base current university californium salary scale. letter **application** , curriculum vita , representative publication manuscript , name address 3 referee send : university californium , san diego open search committee **department** linguistic , 0108 9500 gilman drive la jollum , ca 92093-0108 application material must receive later february 1 , 1995 . University californium equal opportunity, **affirmative** action employer . Announcement supersede our October lsa bulletin announcement our august departmental notice tenure position formal semantics / syntax.

For support count of two the following patterns can be generated:

University
University California
University California application
University California application department
University California application department affirmative

As is shown previous patterns are all a prefix of "University California application department affirmative".

For very long contexts, there are a huge number of patterns. To overcome this problem the concept of closed pattern introduces:

Definition (6) - Closed Iterative Pattern: A frequent iterative pattern P is closed if there exists no super sequence Q such that:

1. P and Q have the same support
2. Every instance of P corresponds to a unique instance of Q, denoted as Inst (P) $\approx$ Inst (Q)

An instance of P (seqP; startP; endP) corresponds to an instance of Q (seqQ; startQ; endQ) iffseqP = seqQ and startP $\geqq$ startQ and endP $\leqq$ endQ where 'seq' means each record in WSDDB.

Definition (7) - Closed Unique Pattern: A frequent pattern P is a closed unique pattern if P contains no repeated constituent events, and there exists no super-sequence Q such that:

1. P and Q have the same support
2. Every instance of P corresponds to a unique instance of Q
3. Q contains no constituent events that repeat

The set of closed frequent patterns maintain the same information as the set of all frequent patterns. By using closet patterns the only pattern that is selected in previous prefixes is the longest because they all have the support count 2. So the total number of pattern is reduced considerably which improves the processing time.

III. Related Works

Web page authors usually use various methods to block spammers and filter spam messages. Their first goal is to discriminate humans and bots. They often want to protect useful information like email addresses from web crawlers. Hence they use javascript, e-mail addresses character replacement or CSS3 cascading style sheets to reformat them that can be readable only to humans. Other interests of spammers are registering many accounts in a forum, group or e-mail providers or inserting commercial or malicious text on the web automatically. Completely Automated Public Turing test to Tell Computers and Humans (CAPTCHA) [2] is frequently used on the web to defeat such problems. Although reformatting and CAPTCHA can help a lot, spammers can still evade such methods. [3-4] present a method for cracking Captcha. Therefore after showing possible methods for breaking Captcha or finding email addresses from web and after spammers can send their messages, we must get an ability to filter them. Based on existing types of spam, different spam detection techniques were presented. Spam messages, spam emails, review spam and profiles are some types of spam which respectively are used in mobile networks, mail servers, online shops and social networks.

[5] presented a data mining approach for detecting spam profiles in social networks. They analyze user messages, friends and activities for selecting effective features. Then they used clustering to affix profiles and create model based on labeled profiles to detect new spam profiles. In Spam messages, review spam and e-mails, spammers usually use text messages to achieve their goals. In addition, e-mail spammers use images instead of texts to elude text spam detection techniques. [6] considered different image attributes such as format, metadata, color, edges and etc. as features and then use classification algorithms on created dataset for predicting image goal. Major of spammers usually focus on text spam. For example in online shops, customers comment their ideas about a product and others often decide to purchase a product based on the comments. Unfortunately, the significance of such reviews encourages spammers to mislead customers with their malicious negative opinions. [7-8] leveraged duplicate review detection and spam classification methods to gain influential spam detection on forum or online market reviews. During recent years, email spam filtering made considerable progress. To solve the problem caused by spam, many solutions have been proposed to detect and filter spam from entering individual's mail box. Blacklist [9] and whitelist filtering can operate based on DNS, IP Address or email address. These methods keep the source of prior received spams in a database. Each time a new Email is received its source is compared by the database. The weakness is when spammers regularly change e-mail and IP addresses to cover their trails. Signature-based methods compare new spam messages with their signature database. In addition rule-based filtering improve detection by discovering the patterns, e.g. words or phrases, malformed headers and misleading dates. For example, RIPPER is based on key-word-spotting rules, which is a rule set generated by user's manual setting. SpamAssassin, popularly used open source spam filter, uses a large set of heuristic rules; however the main disadvantage of rule-based filters is that they tend to have high false positive rates [10].

Another intelligent approach is text classification filtering that plays an increasingly important role in anti-spam in recent years because of their ability of self-learning and good performance. [11-12] also represented machine learning methods for e-mail spam detections. The main approach for detecting spam messages from non-spam messages are supervised learners. For automating anti-spam process, many classification algorithms are applied. The most famous of them are Naïve Bayes [13], k-nearnest neighbor [14], Support Vector Machine[15-16], Artificial Neural Network [17-18], Boosting [19]. The key point in classification problems is considering influential features and their values for example single words in bag of words with the value of their existence or frequency[20-21]. We proposed a method based on text classification filtering with considering frequent and closed sequential pattern as features.

IV. Proposed Method

Our method is outlined as the following steps:

- Preprocessing and stemming datasets
- Selecting best discriminating terms based on a term selection method
- Looking for frequent sequential patterns in corpus
- Using patterns as features
- Feature selection and classification

An overview of the system is shown in Figure 1. First the Body of the message is evaluated and after preprocessing the tokens are extracted. Then using a term selection method, the best discriminative terms are retained and other terms are removed. Then iterative patterns are extracted and a feature vector is built for each sample. Finally Random Forest is applied as classifier. The detail of each step is described in the following.

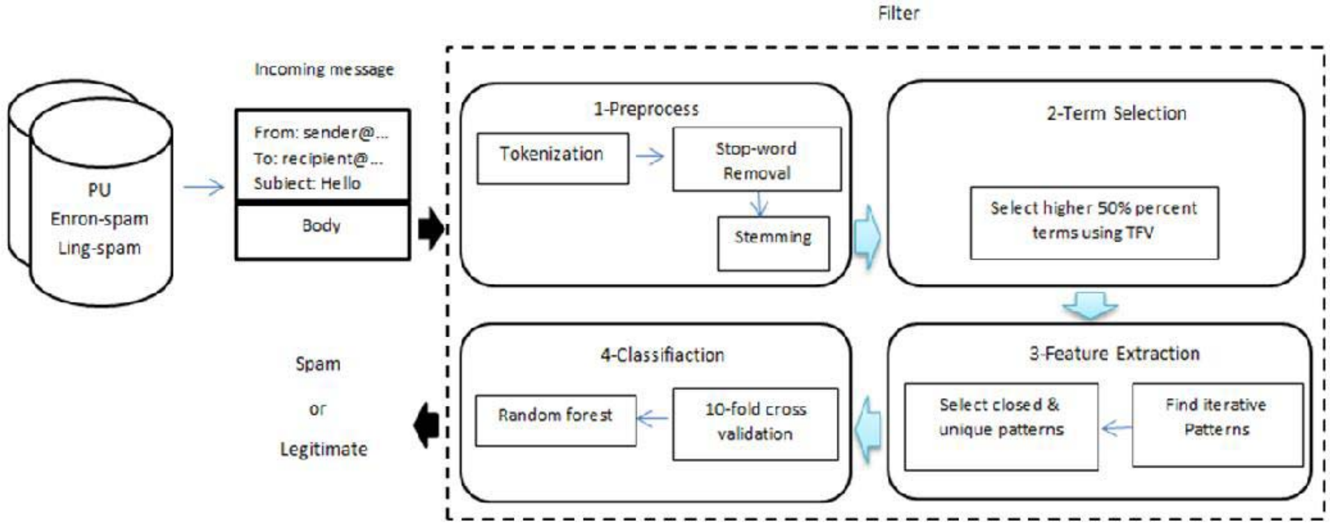

Figure 1. System overview

A. Datasets

We evaluated our proposed method on six benchmark corpora PU1, PU2, PU3, PUA, Enron-Spam and Ling-Spam[1], which are frequently used in the literature. In PU1, PU2, Ling-Spam duplicate spam messages received on the same day are excluded while in PU3 and PUA all duplicates including spam and legitimate are removed. In the Enron-Spam corpus, the legitimate messages of the owners of the mailbox and duplicate messages are removed.

Detail of each dataset is as follows:

- PU1: The corpus contains 1099 messages, including 481 Spam message and 618 legitimate messages. The ratio of legitimate message to spam is 1.28.
- PU2: The corpus contains 721 messages including 142 spam message and 579 legitimate messages. The ratio of legitimate message to spam is 4.01.
- PU3: The corpus contains 4139 messages including 1826 spam message and 2313 legitimate messages. The ratio of legitimate message to spam is 1.
- PUA: The corpus contains 1142 messages including 572 spam message and 572 legitimate messages. The ratio of legitimate message to spam is 1.
- Enron-Spam: The corpus contains 33716 messages including 17171 spam message and 16545 legitimate messages. The ratio of legitimate message to spam is 0.96.
- Ling-Spam: The corpus contains 2893 messages including 481 spam message and 2412 legitimate messages. The ratio of legitimate message to spam is 5.01.

B. Preprocessing

Preprocessing is considered as an important step in text mining. In many practical applications it takes more than 60 percent of the process. Preprocessing is influential because there are many brummagem words with little information in the message and removing these words increases the overall accuracy and processing speed. In this work, we first omit insignificant words such as stop words[2] , prepositions, and etc, from every message to find more discriminative features.

After removing insignificant words, Stemming was applied. The term “Stemming” means finding the origin of the words and removing prefixes and postfixes. By using Stemming, forms of a word, like adjectives, nouns and, verbs, are converted to homological-like word. For instance, both ‘capturing’ and ‘captured’ are converted to a same word, ‘capture’. In this work, to find instances of an iterative pattern, stemming is used. Because we want to ensure that two sequences of “going to drive” and “goes to driving” are seen as a single sequence “go drive”. We used Porter [22] stemmer as one of the most popular open-source application for stemming to fulfill our concerns.

C. Term Selection

After the preprocessing, the size of corpus was still very large which would cause high computational complexity. To reduce the computational complexity a term selection method should be utilized for removing less informative terms.

In this work TFV was used as the term selection method. Term Frequency Variance (TFV) method was developed by [23] method for selecting the terms with high variance. These terms are considered to be more informative. Evaluating all terms in the training corpus, terms occurring primarily in one category (spam or legitimate e-mail) would be retained. In contrast terms occurring in both categories with comparable term frequencies would be removed.

TFV is defined as follows:

$$TFV\,(t_i) = \sum_{C \in \{C_s, C_l\}} [TFV\,(t_i, C) - TFV_f^u(t_i)]^2$$

Where C denotes an e-mail class (C_s and C_l are the spam and legitimate email class respectively). $TFV\,(t_i, C)$ is the term frequency of t_i calculated with respect to category C, and $TFV_f^u(t_i)$ is the average of term frequencies calculated with respect to both categories.

[23] Showed that TFV outperformed the widely used and computationally more expensive Information Gain (IG) method. TFV was calculated for all the terms in the corpus, the computed values were sorted decreasingly. Afterward the higher 50 percent were retained and the lower 50 percent were removed.

Choosing the proper percent of terms is important because if a low percent be used many less informative terms would remain which leads to low accuracy and if a high percent be used many of the informative terms would be removed, leading to low accuracies. After experimental results on the percent of term it was found that 50 percent of terms are good enough to not loose informative terms and still have a high accuracy. The new corpus was built by removing the lowers 50 percent terms.

D. Classification

Classification is a data analysis task that extracts models to describe data classes. Such models, called classifiers, predict categorical class labels. For example, suppose to find whether a message is spam or legitimate, using classification, a model or classifier is constructed to predict class (categorical) labels, such as “Spam” or “Legitimate”. Data classification is a two-step process. In the first step, a classification algorithm builds the classifier by analyzing or “learning from” a training set made up of database tuples and their associated class labels. In the second step, the model is used for classification. A different set called test set is used to evaluate the correctness of the built model. Test data are independent of the training data, meaning that they were not used to construct the classifier.

To evaluate the performance of our system, we used 10-fold cross validation. Our train dataset is divided into 10 sub samples with the same number of instances. Each time, we use 9 of them as a train data and the remainder is used for

[1] http://www.aclweb.org/aclwiki/index.php?title=Spam_filtering_datasets

[2] http://www.ranks.nl/resources/stopwords.html

testing. After finding iterative patterns, we consider closed frequent iterative patterns as features of dataset. We use support for value of the patterns and build a feature vector for each sample. Afterward Random Forest was applied as classifier and its performance was evaluated with 10-Fold cross validation. We used [24] for conducting the experiments. [23] evaluated Random Forest, Decision Trees , SVM and Naïve Bayes on different spam corpora including Ling-spam, PU1, and others. They investigated that Random Forest outperformed all other classifiers. They found that Random Forest is a promising approach for email filtering since it is easy to tune, performs well, and runs efficiently on large datasets with many variables.

V. Experiments

In this section we compared our method through the experiments with some relevant approaches on six datasets. These approaches are Naïve Bayesian-BoW and SVM-BoW [23, 25] which were examined on PU1,PU2,PU3,PUA and Enron-Spam. Because these approaches did not examined their performance on Ling-Spam, we used another popular Naïve Bayesian approach for comparison on Ling-Spam[26]. In Naïve Bayesian-BoW and SVM-BoW, Naïve bayes and SVM are utilized as classifier, respectively BoW (Bag of Words) is utilized as the feature extraction approach.

The results of the PU, Enron and Ling-Spam comparison are in Table 1. We could improve accuracy on Enron dataset and PU datasets and achieved near optimal results on Ling-Spam dataset. In addition of achieving high accuracy, Spammers cannot easily evade from our method. Because our method is tenacious against added single irrelevant words and the words are removed in the step of finding iterative patterns.

The method's accuracy is dependent on support count, as support count decreases the system's accuracy increases so in the table we showed the best support count between 0.1 til 0.01 for each dataset.

Recall (RC) is defined as the portion of total spam e-mails that are correctly classified. Precision (PR) refers to the probability that an e-mail is correctly classified as spam. Area under ROC Curve (AUC) is equal to the probability that a classifier rank a randomly chosen positive sample higher than a randomly chosen negative one. Root mean squared error (RMSE) is a quadratic scoring rule that measures the average magnitude of the error. FMeasure (FM) is the harmonic mean of precision and recall. All of the experiments were carried out on a 2.27GHz Intel Core i5 PC with 4 GB physical memory.

TABLE I. Results on 6 benchmark datasets

Corpus	PR	RC	FM	RMSE	AUC	ACC[a]	NB-FM[b]	NB-ACC[c]	SVM-FM[d]	SVM-ACC[e]
PU1	0.951	0.968	0.954	0.11	0.988	96.26	94.23	94.59	94.79	95.32
PU2	0.855	0.863	0.859	0.12	0.962	94.53	85.14	93.66	83.74	93.66
PU3	0.956	0.96	0.958	0.12	0.982	96.76	94.21	94.79	95.57	96.08
PUA	0.948	0.954	0.951	0.12	0.984	95.06	94.57	94.47	93.08	92.89
Enron-Spam	0.974	0.981	0.986	0.10	0.991	97.47	87.32	88.41	94.62	95.13
Ling-Spam	0.991	0.965	0.983	0.03	0.997	99.93	81.83	99.99	97.41	98.27

a. Accuracy of our proposed method
b. F-measure of the best naïve Bayesian related approach
c. Accuracy of the best naïve Bayesian related approach
d. F-measure of the best SVM related approach
e. Accuracy of the best SVM related approach

VI. Conclusion and Future Works

In this paper we described as a new approach toward spam filtering by introducing iterative patterns. These patterns are sequences of words that are not necessarily coherent like n-grams but the order of words in the context is preserved. We applied our method with Random Forest on standard benchmark datasets and outperformed the results compared to state of arts approaches. In future, we will attempt to replace synonyms in the corpus with a unique defining work which may lead to better patterns.

References

[1] G. Schryen, "Anti-Spam Measures: Analysis and Design", Springer, 2007.

[2] Captcha, "The Captcha project", 2010. Available: http://www.captcha.net

[3] S. B. Elie Bursztein, John C. Mitchell, Dan Jurafsky , Céline Fabry, "How Good are Humans at Solving CAPTCHAs? A Large Scale Evaluation," presented at the Symposium on Security and Privacy, USA, 2010.

[4] M. M. t. Elie Bursztein, John C. Mitchell, "Text-based CAPTCHA Strengths and Weaknesses," presented at the Computer and Communication security, 2011.

[5] C. K. Gianluca Stringhini, Giovanni Vigna, "Detecting Spammers On Social Networks," presented at the 26th Annual Computer Security Applications Conference (ACSAC), USA, 2010.

[6] R. G. Mark Dredze, Ari Elias-Bachrach, "Learning Fast Classifiers for Image Spam," presented at the 4th Conference on Email and Anti-Spam (CEAS), 2007.

[7] B. L. Nitin Jindal, "Analyzing and Detecting Review Spam," presented at the 7th International Conference on Data Mining, USA, 2007.

[8] B. L. Nitin Jindal, "Review spam detection," presented at the 16th international conference on World Wide Web, USA, 2007.

[9] SpamCop, “SpamCop Blocking List”,2010. Available: http://www.spamcop.net/bl.shtml

[10] C. O’Brien, Vogel, C., "Comparing SpamAssassin with CBDF Email Filtering," presented at the 7th annual CLUK research Colloquium, UK, 2004.

[11] H. W. D. DeBarr, "Spam Detection using Clustering, Random Forests, and Active Learning," presented at the 6th Conference on Email and Anti-Spam, California, 2009.

[12] S. M. E. W.A. Awad, "Machine Learning methods for E-mail Classification," International Journal of Computer Applications, 2011.

[13] S. D. M. Sahami, D. Heckerman, E.Horvitz, "A baysian Approach to filtering junk e-mail," presented at the AAAI tech. Rep., 1998.

[14] I. A. G. Sakkis, G. Paliouras, V. Karkaletis, C. D. Spyropoulos, P. Stamatopoulos, "A memory-based approach to antispam filtering for mailing lists," Inform. Retrieval, vol. 6, pp. 49-73, 2003.

[15] D. W. H.Drucker, V. Vapnik, "Support vector machines for spam categorization," IEEE Trans. Neural Network, vol. 10, pp. 1048-1054, 1999.

[16] S.-W. L. Kuo-Ching Ying, Zne-Jung Lee, Yen-Tim Lin, "An ensemble approach applied toclassify spam e-mails," Expert Systems with Application, vol. 37, pp. 2197-2201, 2010.

[17] I. K. J. Clark, J. Poon, "A neural network based approach to authomate e-mail classification," presented at the IEEE int. Conf. Web Intelligence, Canada, 2003.

[18] C.-H. Wu, "Behavior-based spam detection using a hybrid method of rule-based techniques and neural networks," Expert systems with Applications, vol. 36, pp. 4321-4330, 2009.

[19] L. M. X.carreras, "Boosting trees for anti-spam email filtering," presented at the 4th international Conference of Recent Advance in Natural Language Processing (RANLP), 2001.

[20] M. C. T. S. Guzella, "A review of machine learning approaches to spam filtering," Expert Systems with Application, vol. 32, pp. 10206-10222, 2009.

[21] J. H. Karel Jezek, "The Fight against Spam - A Machine Learning Approach," presented at the Electronic Publishing (ELPUB), 2007.

[22] M. Porter. Available: http://tartarus.org/martin/PorterStemmer/

[23] I. Koprinska, et al., "Learning to classify e-mail.," Information Sciences Including Special Issue on Hybrid Intelligent Systems, pp. vol. 177, pp. 2167-2187, 2007.

[24] Waikato, “Weka 3: Data Mining open source Software”, 2008. Available: http://www.cs.waikato.ac.nz/ml/weka/

[25] Ion Androutsopoulos, et al., "Learning to Filter Unsolicited Commercial E-mail NCSR "Demokritos" Tech, Rep. 2004/2, minor corrections," 2006.

[26] Ion Androutsopoulos, et al., "An Evaluation of Naive Bayesian Anti-Spam Filtering," Proceedings of the workshop on Machine Learning in the New Information Age, G. Potamias, V. Moustakis and M. van Someren (eds.), 11th European Conference on Machine Learning, Barcelona, Spain, : p. pp. 9-17, 2000.