Spam Detection Using Character N-Grams

Ioannis Kanaris[1], Konstantinos Kanaris[2], and Efstathios Stamatatos[1]

[1] Dept. of Information and Communication Systems Eng.,
University of the Aegean,
83200 – Karlovassi, Greece
stamatatos@aegean.gr
[2] Dept. of Mathematics,
University of the Aegean,
83200 – Karlovassi, Greece

Abstract. This paper presents a content-based approach to spam detection based on low-level information. Instead of the traditional 'bag of words' representation, we use a 'bag of character *n*-grams' representation which avoids the sparse data problem that arises in *n*-grams on the word-level. Moreover, it is language-independent and does not require any lemmatizer or 'deep' text preprocessing. Based on experiments on Ling-Spam corpus we evaluate the proposed representation in combination with support vector machines. Both binary and term-frequency representations achieve high precision rates while maintaining recall on equally high level, which is a crucial factor for anti-spam filters, a cost sensitive application.

1 Introduction

Nowadays, e-mail is one of the cheapest and fastest available means of communication. However, a major problem of any internet user is the increasing number of unsolicited commercial e-mail, or *spam*. Spam messages waste both valuable time of the users and important bandwidth of internet connections. Moreover, they are usually associated with annoying material (e.g. pornographic site advertisements) or the distribution of computer viruses. Hence, there is an increasing need for effective *anti-spam filters* that either automate the detection and removal of spam messages or inform the user of potential spam messages.

Early spam filters were based on blacklists of known spammers and handcrafted rules for detecting typical spam phrases (e.g., 'free pics'). The development of such filters is a time-consuming procedure. Moreover, they can easily be fooled by using forged e-mail addresses or variations of known phrases that is still readable for a human (e.g., f*r*e*e.). Hence, new rules have to be incorporated continuously to maintain the effectiveness of the filter.

Recent advances in applying machine learning techniques to text categorization [1] inspired researchers to develop content-based spam filters. In more detail, a collection of both known spam and legitimate (non-spam) messages is used by a supervised learning algorithm (e.g., decision trees, support vector machines, etc.) to develop a model for automatically classifying new, unseen messages to one of these two

G. Antoniou et al. (Eds.): SETN 2006, LNAI 3955, pp. 95 – 104, 2006.
© Springer-Verlag Berlin Heidelberg 2006

categories. That way, it is easy to develop personalized filters suitable for either a specific user or a mailing list moderator.

Spam detection is not a typical text categorization task since it has some intriguing characteristics. In particular, both spam and legitimate messages can cover a variety of topics and genres. In other words, both classes are not homogeneous. Moreover, the length of e-mail messages varies from a couple of text lines to dozens of text lines. In addition, the message may contain grammatical errors and strange abbreviations (sometimes inspired by spammers in order to fool spam filters). Therefore, the learning model should be robust in such conditions. Furthermore, besides the content of the body of the e-mail messages, useful information can be found in e-mail address, attachments etc. Such additional information can considerably assist the effectiveness of spam filters [2]. Last, but not least, spam detection is a cost sensitive procedure. In the case of a fully-automated spam filter, the cost of characterizing a legitimate message as spam is much higher than letting a few spam messages pass. This fact of crucial importance should be considered in evaluating spam detection approaches.

All supervised learning algorithms require a suitable representation of the messages, usually in the form of an attribute vector. So far, the vast majority of machine learning approaches to spam detection use the *bag of words* representation, that is, each message is considered as a set of words that occur a certain number of times [2, 3, 4, 5]. Putting it another way, the context information for a word is not taken into account. The word-based text representations require a tokenizer (to split the message into tokens) and usually a lemmatizer (to reduce the set of words). A common practice of spammers is to attempt to confuse tokenizers, using structures such as 'f.r.e.e.', 'f-r-e-e-', 'f r e e', etc. The use of a lemmatizer is language-dependent procedure. There is no effective lemmatizers available for any natural language, especially for morphologically rich languages. On the other hand, word *n*-grams, i.e., contiguous sequences of *n* words, have also been examined [6]. Such approaches attempt to take advantage of phrasal information (e.g., 'buy now'), that distinguish spam from legitimate messages. However, word *n*-grams considerably increase the dimensionality of the problem and the results so far are not encouraging.

In this paper, we focus on a different but simple text representation. In particular, each message is considered as a *bag of character n-grams*, that is *n* contiguous characters. For example, the character 4-grams of the beginning of this paragraph would be: 'In t', 'n th', ' thi', 'this', 'his ', 'is p', 's pa', ' pap', 'pape', 'aper', etc. Character *n*-grams are able to capture information on various levels: lexical ('the ', 'free'), word-class ('ed ', 'ing '), structural ('!!!', 'f.r.'). In addition, they are robust to grammatical errors and strange usage of abbreviations, punctuation marks etc. The bag of character *n*-grams representation is language-independent and does not require any text preprocessing (tokenizer, lemmatizer, or other 'deep' NLP tools). It has already been used in several tasks including language identification [7], authorship attribution [8], and topic-based text categorization [9] with remarkable results in comparison to word-based representations.

An important characteristic of the *n*-grams on the character-level is that it avoids (at least to a great extent) the problem of sparse data that arises when using *n*-grams on the word level. That is, there is much less character combinations than word combinations, therefore, less *n*-grams will have zero frequency. On the other hand, the proposed representation still produces a considerably larger feature set in comparison

with traditional bag of words representations. Therefore, learning algorithms able to deal with high dimensional spaces should be used. Support Vector Machines (SVM) is a supervised learning algorithm based on the *structural risk minimization* principle [10]. One of the most remarkable properties of SVMs is that their learning ability is independent of the feature space dimensionality, because they measure the complexity of the hypotheses based on the margin with which they separate the data, instead of the features. The application of SVMs to text categorization tasks [11] has shown the effectiveness of this approach when dealing with high dimensional data.

In this paper, we propose a content-based spam detection approach based on a bag of character *n*-grams representation and a SVM. No extra information coming from, e-mail address of the sender, attachments etc. is taken into account. Experiments on the publicly available *Ling-Spam* benchmark corpus provide evidence that our approach achieve high spam precision results while maintaining spam recall on equally-high level. Given a cost sensitive evaluation setting, we show that the proposed approach performs better than previous word-based methods.

The rest of this paper is organized as follows: Section 2 includes related work on spam detection. Section 3 describes our approach and Section 4 contains the performed experiments. Finally, section 5 summarizes the conclusions drawn from this study and indicates future work directions.

2 Related Work

Probably the first study employing machine learning methods for spam filtering was published in 1998 [2]. A Bayesian classifier was trained on manually categorized legitimate and spam messages and its performance on unseen cases was remarkable. Since then, several machine learning algorithms have been tested on this task, including boosting decision trees and support vector machines [5], memory-based algorithms [4], and ensembles of classifiers based on stacking [12].

On the other hand, a number of text representations have been proposed dealing mainly with word tokens and inspired from information retrieval. One common method is to use binary attributes corresponding to word occurrence [2, 4]. Alternative methods include word (term) frequencies [6], tf-idf [5], and word-position-based attributes [13]. The dimensionality of the resulting attribute vectors is usually reduced by removing attributes that correspond to words occurring only a few times. Recent work [13] has showed that the removal of the most frequent words (like ‘and’, ‘to’ etc.) considerably improves the classification accuracy. Another common practice is to use a lemmatizer [3] for converting each word-type to its lemma (‘copies’ becomes ‘copy’). Naturally, the performance of the lemmatizer affects the accuracy of the filter and makes the method language-dependent. Finally, the dimensionality of the attribute vector can be further reduced by applying a feature selection method [14] that ranks the attributes according to their significance in distinguishing among the two classes. Only a predefined number of top ranked attributes are, then, used in the learning model.

In addition, word *n*-grams have also been proposed [6, 13] but, so far, the results are not encouraging. Although such a representation captures phrasal information,

sometimes particularly crucial, the dimensionality of the problem increases significantly. Moreover, the sparse data problem arises since there are many word combinations with low frequency of occurrence.

A couple of recent studies attempt to utilize a character-level representation of e-mail messages. In [15] a suffix-tree approach is described which outperforms a traditional Bayesian classifier that is based on a bag of words representation. On the other hand, a representation based on the combination of character 2-grams and 3-grams is proposed in [16]. However, preliminary results in an e-mail categorization task (where many message classes are available) show that approaches based on word-based representations perform slightly better.

Research in spam detection was considerably assisted by publicly available benchmark corpora, so that different approaches to be evaluated on the same testing ground. Nowadays, there are several such benchmark corpora that come from either mailing list messages, hence avoiding privacy issues of legitimate messages, (e.g., Ling-Spam[1]) or the mailboxes of specific users (e.g., SpamAssasin[2]).

3 Our Approach

First, for a given n, we extract the L most frequent character n-grams of the training corpus. Let $<g_1, g_2, \ldots, g_L>$ be the ordered list (in decreasing frequency) of the most frequent n-grams of the training corpus. Then, each message is represented as a vector of length L $<x_1, x_2, \ldots, x_L>$, where x_i depends on g_i. In more detail, we examine two representations:

Binary: The value of x_i may be 1 (if g_i is included at least once in the message) or 0 (if g_i is not included in the message).
Term Frequency (TF): The value of x_i corresponds to the frequency of occurrence (normalized by the message length) of g_i in the message.

The produced vectors can be arbitrarily long. On one hand, if L is chosen too short, the messages are not represented adequately. On the other hand, if L is chosen too long the dimensionality of the problem increases significantly. In the experiments described in the next section, L was set to 4,000. A feature selection method can then be applied to the resulting vectors, so that only the most significant attributes contribute to the classification model. A feature selection method that proved to be quite effective for text categorization tasks is *information gain* [14]. The information gain of a feature x_i is defined as an expected reduction in entropy by taking x_i as given:

$$IG(C, x_i) = H(C) - H(C| x_i) \quad (1)$$

where C denotes the class of the message ($C \in \{spam, legitimate\}$) and $H(C)$ is the entropy of C. In other words, $IG(C, x_i)$ is the information gained by knowing x_i. Information gain helps us to sort the features according to their significance in distinguishing between spam and legitimate messages. Only the first m most significant attributes are, then, taken into account.

[1] Available at: http://www.aueb.gr/users/ion/data/lingspam_public.tar.gz
[2] Available at: http://spamassasin.org/publiccorpus/

The produced vectors (of length m) of the training set are used to train a SVM classifier. The Weka [18] implementation of SVM was used (default parameters were set in all reported experiments).

4 Experiments

4.1 Benchmark Corpus

The corpus used in this paper is Ling-Spam consisting of 2,893 emails, 481 spam messages and 2,412 legitimate messages taken from postings of a mailing list about linguistics. This corpus has a relatively low spam rate (16%) and the legitimate messages are not as heterogeneous as the messages found in the personal inbox of a specific user. However, it has already been used in previous studies [3, 4, 15] and comparison of our results with previous word-based methods is feasible. Moreover, it provides evidence about the effectiveness of our approach as assistance to mailing list moderators.

The bare version of this corpus was used (no lemmatizing or stop-word removal was performed) so that to be able to extract accurate character n-gram frequencies. Unfortunately, this corpus was already converted to lower case, so it was not possible to explore the significance of upper case characters.

In the experiments described below, a ten-fold cross-validation procedure was followed. That is, the entire corpus was divided into ten equal parts, in each fold a different part is used as test set and the remaining parts as training set. Final results come from averaging the results of each fold.

4.2 Evaluation Measures

Two well known measures from information retrieval community, *recall* and *precision*, can describe in detail the effectiveness of a spam detection approach. In more detail, given that $n_{S\to S}$ is the amount of spam messages correctly recognized, $n_{S\to L}$ is the amount of spam messages incorrectly categorized as legitimate, and $n_{L\to S}$ is the amount of legitimate messages incorrectly classified as spam, then, spam recall and spam precision can be defined as follows:

$$\text{Spam Recall} = \frac{n_{S\to S}}{n_{S\to S} + n_{S\to L}} \tag{2}$$

$$\text{Spam Precision} = \frac{n_{S\to S}}{n_{S\to S} + n_{L\to S}} \tag{3}$$

In intuitive terms, spam recall is an indication of filter effectiveness (the higher the recall, the less spam messages pass) while spam precision is an indication of filter safety (the higher the precision, the less legitimate messages blocked).

However, spam detection is a cost sensitive classification task. So, it is much worse to misclassify a legitimate message as spam than vice versa. Therefore, we need an evaluation measure that incorporates an indication of this cost. A cost factor λ is assigned to each legitimate message, that is, each legitimate message is considered as λ messages [3, 4]. In other words, if a legitimate message is misclassified, λ errors

occur. A cost-sensitive evaluation measure, the *Total Cost Ratio* (TCR) can, then, be defined [3, 4] as follows:

$$\mathrm{TCR} = \frac{n_{S\to S} + n_{S\to L}}{\lambda \cdot n_{L\to S} + n_{S\to L}} \tag{4}$$

The higher the TCR, the better the performance of the approach. In addition, if TCR is lower than 1, then the filter should not be used (the cost of blocking legitimate messages is too high). To be in accordance with previous studies, three cost scenarios were examined:

Low cost scenario (λ=1): This corresponds to an anti-spam filter that lets a message classified as spam to reach the mailbox of the receiver along with a warning that the message is probably spam.
Medium cost scenario (λ=9): This corresponds to an anti-spam filter that blocks a message classified as spam and the sender is informed to resend the message.
High cost scenario (λ=999): This corresponds to a fully-automated filter that deletes a message classified as spam without notifying either the receiver or the sender.

4.3 Results

Three sets of experiments were performed based on character 3-gram, 4-gram, and 5-gram representations, respectively. In all three cases, both binary and TF attributes were examined. Moreover, different values of the *m* attributes left after the feature selection procedure were tested (*m* starts from 250 and then varies from 500 to 4000 by 500).

The results of the application of our approach to Ling-Spam are shown in Fig. 1. As can be seen, for binary attributes, 4-grams seems to provide the more reliable representation (for $m>2000$). On the other hand, for TF attributes there is no clear winner. More significantly, binary attributes seem to provide better spam precision results while TF attributes are better in terms of spam recall. In most cases, spam recall was higher than 97% while, at the same time, spam recall was higher than 98%. Moreover, a few thousands of features are required to get these results. This is in contrast to previous word-based approaches that deal with limited amount (a few hundreds) of attributes. This provides another evidence that SVM can effectively cope with high dimensional data.

The results of the cost-sensitive evaluation are shown in Fig. 2. In particular, TCR values for 3-grams, 4-grams, and 5-grams are given for varying number of attributes. Results are given for both binary and TF attributes as well as the three evaluation scenarios (λ=1, 9, and 999, respectively). As can be seen, in all three scenarios, a representation based on character 4-grams with binary attributes provides the best results. This stands for a relatively high number of attributes ($m>2500$). For λ=1, and λ=9 the TCR results are well above 1 indicating the effectiveness of the filter. On the other hand, for λ=999, the TCR results are less than 1 indicating that the filter should not be used at all. However, it is difficult for this scenario to be used in practice.
Table 1 shows a comparison of the proposed approach with previously published results on the same corpus in terms of spam recall, spam precision, and TCR values.

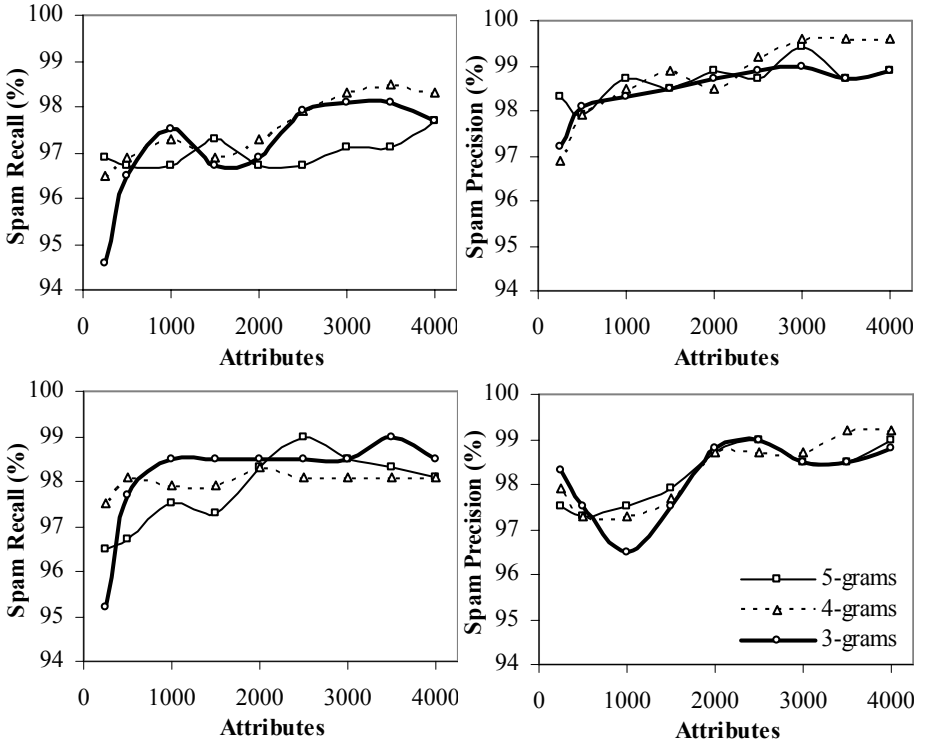

Fig. 1. Spam recall and spam precision of the proposed approach based on character 3-grams, 4-grams, and 5-grams and varying number of attributes. Top: binary attributes. Bottom: TF attributes.

In more detail, best results achieved by three methods are reported: a Naïve Bayes (NB) classifier [3], a Memory-Based Learner (MBL) [4], and a Stacked Generalization approach (SG) [12] using word-based features and a Suffix Tree (ST) [15] approach based on character-level information. The number of attributes that correspond to the best results of each method is also given. It should be noted that the results for the ST approach are referred to a sub-corpus of Ling-Spam with a proportion of spam to legitimate messages approximately equal to the entire Ling-Spam corpus (200 spam and 1,000 legitimate messages). Moreover, no results were reported for the SG approach based on the high cost scenario.

As concerns the TCR, the proposed approach is by far more effective than word-based approaches for the low and medium cost scenarios. This is due to the fact that it manages to achieve high spam recall while maintaining spam precision on equally-high level. ST is also quite competitive. This provides extra evidence that character-based representations are better able to capture the characteristics of spam messages. On the other hand, the proposed approach failed to produce a TCR value greater than 1 for the high cost scenario. That is because the precision failed to be 100%. It must

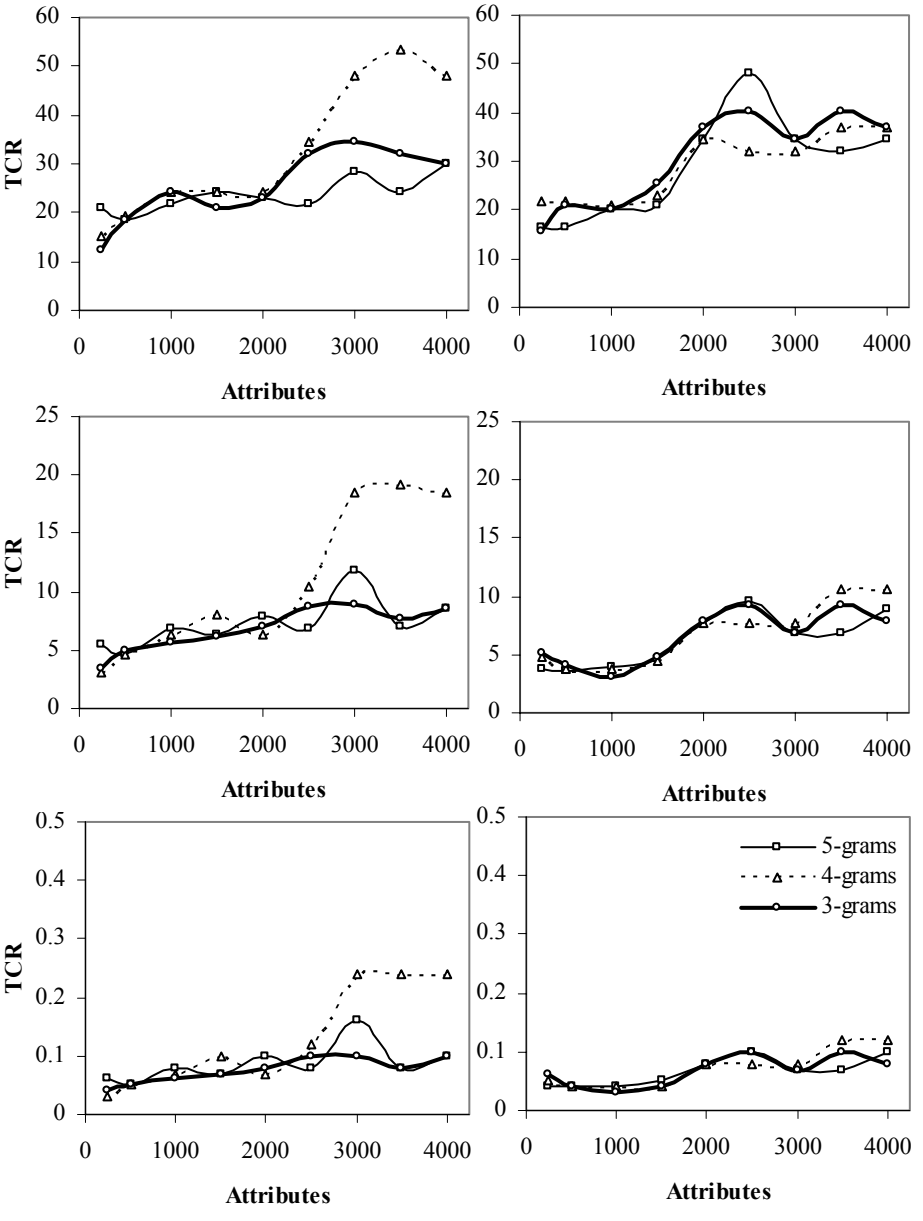

Fig. 2. Results of cost-sensitive evaluation. TCR values for λ=1(top), λ=9 (middle), and λ=999 (bottom) and varying number of attributes and *n*-gram length. Left column: binary attributes. Right column: TF attributes.

be underlined that previous studies [3, 4] show that TCR is not stable for the high cost scenario and it is common for TCR to exceed 1 only for very specific settings. Hence, it is not yet feasible to construct a practical filter based on this scenario.

Table 1. Comparison cost-sensitive evaluation (λ=1, 9, and 999) of the proposed approach with previously published results on Ling-Spam. Best reported results for spam recall, spam precision, and TCR are given. ST results refer to a sub-corpus of Ling-Spam.

Approach	λ	Attributes	Recall	Precision	TCR
NB	1	100	82.35%	99.02%	5.41
MBL	1	600	88.60%	97.39%	7.81
SG	1	300	89.60%	98.70%	8.60
ST	1	-	97.22%	100%	35.97
Proposed	1	3,500	98.50%	99.60%	52.75
NB	9	100	77.57%	99.45%	3.82
MBL	9	700	81.93%	98.79%	3.64
SG	9	100	84.80%	98.80%	4.08
ST	9	-	98.89%	98.89%	9.01
Proposed	9	3,500	98.50%	99.60%	19.76
NB	999	300	63.67%	100%	2.86
MBL	999	600	59.91%	100%	2.49
ST	999	-	97.78%	100%	45.04
Proposed	999	3,500	98.50%	99.60%	0.25

5 Conclusions

In this paper we presented a content-based approach to spam detection. In contrast to the majority of previous studies, character-level information is used to represent the messages. The performed experiments indicate that a character *n*-gram representation in combination with a support vector classifier is an effective approach for anti-spam filters. The presented results show that the proposed method considerably improves the best reported results on the same corpus for two out of three cost-sensitive scenarios. The amount of attributes required for achieving that performance is considerably higher in comparison to word-based approaches. On the other hand, the proposed method failed to be competitive in the framework of a fully-automated filter (λ=999). However, that scenario does not yet correspond to systems of everyday use.

A publicly available corpus (Ling-Spam) was used for evaluating our approach. Since the legitimate messages of this corpus include mailing list messages about a specific topic (linguistics), they are less heterogeneous than the messages found in the inbox of a particular user. Therefore, the experimental results suggest the application of the proposed method to anti-spam filters assisting mainly mailing list moderators. However, more extensive evaluation on corpora coming from personal user inboxes is needed.

The presented experiments were based on a predefined *n*-gram length (*n*=3, 4, or 5) and all messages were converted to lower case. A promising future work direction would be the combination of variable-length *n*-grams and the distinction between lower case and upper case *n*-grams.

References

1. Sebastiani, F.: Machine Learning in Automated Text Categorization. ACM Computing Surveys, 34(1) (2002) 1–47
2. Sahami M., Dumais S., Heckerman D., Horvitz E.: A Bayesian Approach to Filtering Junk E-mail. In Proc. of AAAI Workshop on Learning for Text Categorization (1998).
3. Androutsopoulos I., Koutsias J., Chandrinos K.V., Paliouras G., Spyropoulos C.D.: An Evaluation of Naive Bayesian Anti-Spam Filtering. In Potamias, G., Moustakis, V. and van Someren, M. (Eds.), Proc. of the Workshop on Machine Learning in the New Information Age, 11th European Conference on Machine Learning (2000) 9-17
4. Sakkis, G. , Androutsopoulos I., Paliouras G., Karkaletsis V., Spyropoulos C.D., Stamatopoulos P.: A Memory-Based Approach to Anti-Spam Filtering for Mailing Lists. Information Retrieval, 6(1) (2003) 49-73
5. Drucker, H., Wu, D., Vapnik, V.: Support Vector Machines for Spam Categorization. IEEE Trans Neural Network, 10 (1999) 1048-1054
6. Androutsopoulos I., Paliouras G.. Michelakis E.: Learning to Filter Unsolicited Commercial E-Mail. Technical report 2004/2, NCSR "Demokritos" (2004)
7. Cavnar, W., Trenkle, J.: N-gram-based text categorization. In Proc. 3rd Int'l Symposium on Document Analysis and Information Retrieval (1994) 161-169
8. Keselj, V., Peng, F., Cercone, N., Thomas, C.: N-gram-based Author Profiles for Authorship Attribution. In Proc. of the Conference Pacific Assoc. Comp. Linguistics (2003)
9. Lodhi, H., Saunders, C., Shawe-Taylor, J., Cristianini, N., Watkins, C.: Text Classification Using String Kernels The Journal of Machine Learning Research, 2 (2002) 419 – 444
10. Vapnik V.: The Nature of Statistical Learning Theory. Springer, New York (1995).
11. Joachims T.: Text Categorization with Support Vector Machines: Learning with Many Relevant Features. In Proc. of the European Conference on Machine Learning (1998)
12. Sakkis G., Androutsopoulos I., Paliouras G., Karkaletsis V., Spyropoulos C.D., Stamatopoulos P.: Stacking Classifiers for Anti-Spam Filtering of E-Mail. In Proc. of 6th Conf. Empirical Methods in Natural Language Processing (2001) 44-50
13. Hovold J.: Naive Bayes Spam Filtering Using Word-Position-Based Attributes. In Proc. of the Second Conference on Email and Anti-Spam (2005)
14. Yang, Y., Petersen, J.O.: A Comparative Study on Feature Selection in Text Categorization. In Proc. of the 14th Int. Conference on Machine Learning (1997) 412-420
15. Pampapathi, R., Mirkin, B., Levene, M.: A Suffix Tree Approach to Text Categorisation Applied to Spam Filtering. http://arxiv.org/abs/cs.AI/0503030
16. Berger, H., Koehle, M., Merkl, D.: On the Impact of Document Representation on Classifier Performance in e-Mail Categorization. Proc. of the 4th International Conference on Information Systems Technology and its Applications (2005) 19-30
17. Witten I.H., Frank E.: Data Mining: Practical Machine Learning Tools with Java Implementations. Morgan Kaufmann, San Francisco (2000)