Machine learning as a last line of defense against spam emails

Elkhmasi Mustafa Asma
Advanced Protection Systems, Singidunum University/Belgrad-Serbia
am.asma45@yahoo.com

Abstract - **In this paper various learning based algorithms for classification of text messages as spam or non-spam. A number of learning based algorithms was tested with the emphasis on the importance of extraction of "good" feature vectors in order to improve the performance of classifiers. We did not focus in our paper on the feature extraction itself. The analysis of the testing results showed that, in terms of accuracy, there is no benefit in utilization of support vector machine algorithms or ensemble-learning algorithms compared to the naïve Bayesian algorithm. The tests have also shown that improvement of performance of NB can be achieved by extracting more meaningful classification features.**

Keywords - **Machine learning, malicious attacks, spam emails, classification spam emails.**

I. Introduction

Electronic mail is very efficient and popular communication medium and as such is very prone to misuse. It cannot be specified when or who exactly was the first to realize that at least one person, out of millions recipients, will respond to and email containing an advertisement without taking into account the proposal itself. Considering that email presents an easy and fast way for the sender to distribute an advertisement without any cost, it is easily understandable that many organizations take advantage of emailing. Consequently, our email boxes are becoming cluttered with unwanted advertisement also known as spam or junk mail varying in its content to a significant extent. Some of it may even carry a harmful content. General feature of spam emails is that they are of no interest to the majority of recipients.

Large amount of spam e-mails which is being distributed constantly, causes many troubles to the internet community, such as the following: delays in the traffic between the servers in delivery of legitimate email, sorting out of unwanted messages is time-consuming, there is always a risk of deleting regular mail by mistake, and finally there is also an amount of pornographic spam which should not be neglected, and access to which should be restricted to children. In other words, large spam traffic negatively affects email servers storage and processing power, networks bandwidth, work productivity and user's time. Some studies showed that app. 70% of total email traffic was spam traffic.

When it comes to fighting against spam mails, many ways have been introduced, legal measures, personal involvement referring to never responding to spam of never publish your email address on web sites, etc., as well as technological ways such as blocking spammers` IP address of email filtering[1]. Attempts to introduce legal measures against spam mailing have had limited effect. One of the most effective methods for fighting against spam are anti-spam filters operating in various ways, starting from blacklists of frequent spammers to content-based filters operating by searching and identifying particular keywords or keyword patterns for mail classification. Regardless of the previously mentioned methods of fighting against spam, there is no universal or perfect way to eliminate spam. There is an ongoing battle between spammers and spam-filtering methods. The finer and more sophisticated the spam-filtering methods get, the more tricks the spammers use to overcome them. In the meantime, the amount of junk mail is in constant increase.

II. Basics of the problem

Herein, we shall try to classify text messages, which are not convenient objects for handling. Objects, which are mostly classified by the machine learning algorithms, are numerical objects, i.e. real numbers or vectors, or objects that require some measure of similarity between them-selves.

In the first case all messages should be converted to vectors of numbers (feature vector) and then these vectors should be classified. As an example, frequently the vector of numbers of occurrences of certain words in a message is taken as feature vector. Due to the fact that there is always loss of information in the process of feature extraction, it is easy to perceive that the method we choose to define the feature-extractor is essential for the performance of the filter. If the chosen features enable both spam and legitimate mail to exist with the same feature vector, the machine-learning algorithm will make mistakes regardless of how good it is. However, a wise choice of features may allow easier classification. Taking under consideration the above said, it is clear that choosing the features for classification is much more significant than the algorithm to be used. Strangely, not many attentions was paid to the way of choosing "a good feature", mostly the basic vectors are used, such as the vector of word frequencies or something similar[2].

In our paper we shall not focus our attention on feature extraction. In the following feature vectors will be indicated with letter x and messages with letter m. Machine learning algorithms requiring distance metric or scalar product which will be defined on the set of messages will be considered[3,4]. In this paper, the feature vectors will be extracted and distance/scalar products of these vectors will be used. Due to the complexity of functions existing solely for strings, thus their restrictive use in practice, they will not be considered in

[1] Mason (2013)

[2] Stern (2008)

[3] Muller *et ad.*, (2001)

[4] Cristianini & Shawe-Taylor (2000).

this paper. Further to the above said, there will be obvously a major flaw in the approach, taking under consideration that no sophisticated feature extractors will be used.

Classifier performance

The performance requirements of spam filter are the second major problem. If a spam email is misclassified as legitimate one, this kind of mistake is tolerable and does not cause too much problem to the user[5]. On the other hand, the situation in which a legitimate message is misclassified, as spam is completely intolerable, unacceptable and it can cause rather big problem to the user. In that case mail filtering losses its purpose, since the user would have to review „spam folder“ on regular basis. Filters which make these kinds of mistakes rarely, i.e. do not have as many „false possitives“ are as bad as the previouosly mentioned ones. The user tends to trust such filter and not check spam mail, which could bring to a situation that an imoprtant legitimate mail is lost.

The importance of classifying the legitimate mail correctly can be increased in most of the learning algorithms. The attention must be brought to the fact that if too high importance is assigned to legitimate mail, the filter will simply classify all messages as non-junk, and lose its practical value. There are certain safety measures to be undertaken in order to compensate for filter mistakes, e.g. in case that a message is classified as spam, a reply may be sent to the sender suggesting him to resend the original message to a different address or to include some particulat words in the subject; or a filter could have the possibility to sort the list of e-mails in the mailbox in ascending order of certainty that a specific message is spam[6].

III. Spam Detection Methods

One of the possible ways to detect and prevent spam is application of rule-base filter[7]. The said filtering approach can be applied to the message header in order to check that the source address does not belong to a spammer domain. Furthermore, it can be applied to the body of the message in order to check if there are text patterns or words usually used by spammers. This kind of filter can be bypassed by spammers by confusing the filter with unclear content of spam email, which would otherwise help the filter identify spam mail as such.

Another approach to detect and prevent spam mail is learning-based filter[8]. These filters are trained to, based on the usage of large dataset including spam and legitimate emails, extract knowledge to be used to classify newly received emails and detect spam emails. The majority of techniques under this filtering use learning algorithms such as Naive Bayes Classifier, Support Vector Machines and Artificial Neutral Networks[9]. The disadvantage of this filter is the fact that it lacks progressive learning capability, and in order to be adjusted to changes in the new emails, it has to be retrained with the updated dataset[10].

Taking under consideration the adaptive nature of spam generator and high increase in the email spam, a more efficient and adaptive filtering approach for email classifying must be explored so that features for classification can be easily added or removed withoud retraining the system[11,12].

A. Spam Features

Spam detection is based on the presumtion that its content differs from the content of a legitimate email in ways which can be quantified. Spam emails have certain similar characteristics when it comes to their structure, content and diffusion approaches. Several features such as frequency of particular words or special characters, digits or alphabet may indicate that the mail in question is spam[13]. For example, in one of the studies conducted in this regard typical spammer words such as *winner, dollar, award, cash prize, top job opportunities...* were extracted and ranked[14]. Users are attracted to these words which increases the possiblity for spam emails to be opened. One of the surveyes showed that 55% of all spam traffic in 2012 contained sexual and dating related content[15].

Spam emails can also contain „weird combinations“ mixing the lower case letters and uper-case letters or digits in order to obscure the meaning of the used words and bypass the filter (Credit 4U, O_F_F_E_R or 0ff3R)[16]. Using of only a number of selected features used in the Spam base dataset and optimized parameters shows that spam mail classification was possible with high detection rate.

B. Malicious Spam Features

The amount of spam containing malicious contents has increased. Kaspersky released a report showing that 3.2% of emal traffic had malicious attachments such as Trojans, Worms and Spyware[17]. Such situation gives an adverse dimension to spam emails. Most of the distributed malware was a Trojan having as a purpose to steal users' credentials. Malicious spammers usully change the file extensions in order to disguse malicious attachments. Mostly used extensions are .zip, .rar, .pdf and .jpeg. Large number spam emails contain URLs pointing to malicious websites, most of these with shortened HTTPs and services in order to deceive the users and present these links as trusted[18,19].

Spam emails containg malicious attachments and URLs cause malware infection. A study carried out by Alazab & Broadhurst (2014) over three real world datasets containg ove

[5] Androutsopoulos *et ad.,* (2000)
[6] Drucker, Shaharary & Gibbon (2011)
[7] Kamboj (2010)
[8] Stern (2008)
[9] Guzella & Caminhas (2009)
[10] Su, Lo & Hsu (2010)
[11] Guzella & Chaminhas (2009)
[12] Idris *et al.,* (2015)
[13] Garuana & Li (2012)
[14] Santhi, Wenisch & Sengutuvan (2013)
[15] Symantec (2013)
[16] Luckner, Gad & Sobkowiak (2014)
[17] Gudkova (2013)
[18] Alazab & Broadhurst (2014)
[19] Symantec (2013)

13 million spam emails collected in 2012 showed that over 20% of spam emals had at least one attachment or URL with malicious content and that 90% of the said malicious content were compressed (.zip) files whose true extension was disguised by various obfuscation techniques. These techniques identified in the afore said study can be taken as classification features in order to detect and prevent spam emails with malicious content.

A combination of 58 features used in spam detection were devided into five main categories according to the location of the feature in the email: header features, subject features, payload (body) features, attachments and URLs[20].

In the study of targeted social engineering attacks utilizing emails with malicious content, conducted by Le Blond *et al.*, (2013). using a dataset with malicious attachments, the results showed that the language, topic and timing of emails were highly adjusted to the recipients of these emails, as well as that the senders' email addresses were disguised using of several techniques.

These malicious attachments were analyzed through Virus Total and dynamic taint analysis. Sixty-five different classification features for detection of emails with malicious content were defined. They devided to 2 large categories, persistant threat and recipient oriented features. The first group of features are connected with the attacker's environment (IP address, time zone, character coding and tools), the second group of features are related to the spam victim[21].

IV. Approch for Spam Detection

Presently, the spammers are sending large number of spam messages particularly to the users of Gmail, Hotmail, I Cloud and social networks such as Facebook and Twitter. Delivery of important emails is delayed due to large amount of spam traffic, time is wasted on deleting of annoying messages which all makes spam filtering very difficult. A popular method for detection of spam is Bayesian (Thomas Bayes) spam filtering which is based on correlation of the use of tokens with spam and non-spam and then using Bayesian conclusion in order to determine a probability if an email is a spam or not[22]. When focusing on detecting the maximum number of spams from spam-base dataset, we look particularly for the following:

1) BAYES NEST (BN);
2) LOGIC BOOST (LB);
3) RANDOM TREE (RT);
4) JRIP (JR);
5) J48 (J48);
6) MULTILAYER PERCEPTRON (MP);
7) KSTAR (KS);
8) RANDOM FOREST (RF);
9) RANDOM COMMETTEE (RC) [23].

Bayes network model is a probibalistical graphical model representing a set of random variable and their conditional dependencies through a direct graph. Random forests are ensemble learning method for classification functioning on the basis of building the multitude of decision trees resulting in the mode of the classes output by individual trees[24].

Learning models with corresponding learning algorithms analyzing data and recognizing patterns, which are used for the analysis of classification, are monitored in machine learning[25]. The basic j48 takes input data and envisages which of the two possible classes forms the output, for each given input. Multilayer Perceptron (MLP) is a feedwork artificial neutral network model that maps sets of input data onto a set of appropriate outputs[26]. MLP uses a supervised learning technique for training the newtwork. The said technique is called back propagation[27].

One of the networks highly used for experimenting is WEKA (Waikato Environment for Knowledge Analysis). It possesses highly customizable interface and is very simple to use. It operates as an open software for making and testing classifiers[28].

V. SMS Spam Detection

The mobile phone market is in constant growth. The mobile phone users have the possibility to use SMS, a platform which enables to mobile phone users to communicate by sending text messages. Being that the mobile phones are widely used and SMS are very cheap to be sent, and have become commonplace, it is no surprise that commercial advertisments are sent to the users as text messages. SMS spam is not as common as email spam, but some studies have shown in certain parts of Asia 30% of text messages was spam[29].

There are certain differences between email spam-filtering ans SMS spam-filtering[30]. The real databases for SMS filtering are very limited comparing to the existance of a significant number of email datasets available. Due to the small length of text messages, the number of features which can be used for their classification is significantly smaller than the number of features used for email classification. The language used in text messages is less formal and usually full of abbreviations, which all affects the performance of spam filtering algorithm used for spam identification.

Taking under consideration that mobile phone spam-filtering software is very limited[31], spam detection for text messages represent a problem which should be looked into.

A. *Feature Extraction and Initial Analysis*

For experimenting spam-based dataset is used, it is a large text file in which each line corresponds to one text message.

[20] Tran, Alazab & Broadhurst (2013)
[21] Amin (2011)
[22] Sahami *et al.,* (1998)
[23] Sharma & Arora (2013: 23)
[24] Judea (2000)
[25] Breiman (2001)
[26] Cortes & Vapmik (1995)
[27] Rumelhart, Hinton & Williams (1985)
[28] Santiago (2015)
[29] Shirani-Mehr (2013).
[30] Delany, Buckley & Greene (2012)
[31] Almeida, Gomez & Yamakami (2011)

Firstly, feature extraction is conducted. The length of message is also onw of the features. Later on, an initial analysis of the data is performed using naive Bayes algorithm (NB)[32]. For initial analysis each message is split into tokens of alphabetical characters, any special character and space, comma, etc. was removed from extraction at this point. Too rare and too tokens are removed from the extracted tokens since being that their analysis would not have any practical contribution. The result of applying NB algorithm show very little difference between the analyzed training set and test set. NB algorithm shows very good overall accuracy (Figure 1). Some improvement in the performance of the algorithm can be achieved by adding more meaningful features to the list of tokens, which can decrease the error rate.

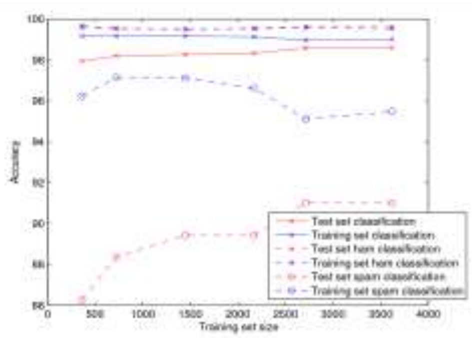

Figure 1: Learning curve for naive Bayes algorithm applied to the dataset and evaluated using cross validation (30% of initial dataset is our test set), by Shirani-Mehr (2013)

What makes this classifier very suitable and desirable for spam detection are its high speed and accuracy.

Support vector machines (SVM) with different kernels can be also applied on the dataset (Figure 2). The results show that while the overal training set error of the model is far less than the error rate for NB, the test set error is well above the one for NB, when it comes to appliying of SVM with linear kernel reducing of the features only shows degradation in performance.

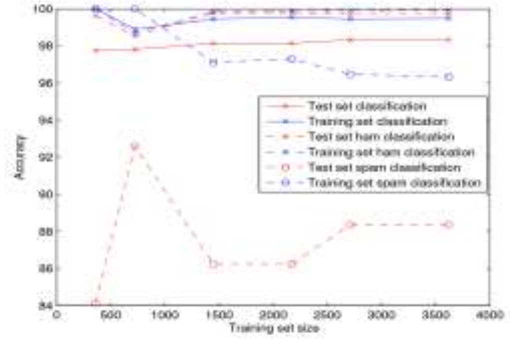

Figure 2: Learning curve for SVM algorithm applied to final features, by Shirani-Mehr (2013)

The experiments have shown no benefit in using of SVM compared to the usage of NB in terms of accuracy.

B. *Ensemble Methods*

Random forests and Ad a boost are two ensemble learning algorithms. Ensemble learning algorithms cab be devide into two categories: averaging methods and boostin methods[33,34,35].

Random forests is an averaging ensemble metod for classification. Comparing the performance of this algorithm with NB, the results show no improvement in performance, even thogh the complexity of the model is increased.

Ad a boost is a boosting ensemble method which sequentially builds classifiers that are modified in favor of misclassified instances by previous classifiers. As said previously for the Random forests, this classifier is still outbeaten by NB when it comes to the performance.

VI. CONCLUSIONS

In this paper the attention was brought to the high amount of spam emailing causing various troubles to the users. A number of methods for spam filtering were mentioned with the focus on the accuracy and performance of algorithms used for classification of spam and non-spam. Based on the conducted the studies in this regard, we could notice that classifiers are showing higher performance rate in terms of accuracy of email classification and spam-filtering with better selected classification features. Due to their importance, more investigation should be done when it comes to the extraction of features used in spam detection. The same can be applied on filtering of text messages from spam, although the situation here is more complexed due to their nature (unformal language usually full of abbreviations, briefness, absence of header) as well as limited software or databases. Having in mind that amount of spam is in constant increase, spam-filtering most certainly represents an issue which should be subject to further research and analysis so as to improve the performance of classifiers for spam detection.

REFERENCES

{1} Mason. S (2013) New Law Designed to Limit Amount of Spam in E-Mail Thapar University, Patiala. http://spam.abuse.net/

{2}Stern. H (2008) A Survey of Modern Spam Tools. 5th Conference on Email and Anti-Spam, CEAS, California.

{3}Muller K-R, Mika. S, Ratsch. G, Tsuda. K, nad Scholkopf. B (2001) An Introduction to Kernel-Based Learning Algorithms. IEEE Transaction on Neural Networks, Vol 12, No 2: 181- 202.

{4}Cristianini. N, and Shawe-Taylor. J (2000) An Introduction to Support Vector Machines, and other Kernel-Based Learning Methods. Cambridge University Press. http://www.support-vector.net

{5}Androutsopoulos. I, Oaliouras. G, Karkaletsis. V, Sakkis. G, Spyropoulos. D. C, and Stamatopoulos. P (2000) Learning to Filter Spam E-Mail: A Comparison of a Naive Bayesian and a Memory-Based Approach. http://arxiv.org/pdf/cs/0009009v1.pdf

[32] Xiong, Li, Jiang & Zhao (2013)

[33] Bauer & Kohavi (1998)

[34] Machova, Bracak & Bendar (2006)

[35] Machova, Puszta & Bednar (2005)

{6}Drucker. H, Shaharary. B., and Gibbon. C. D (2011) Support Vector Machines: relevance feedback and information retrieval. Information Processing and Management 38: 305-323. http://www.journals.elsevier.com/information-processing-and-management.

{7}Kamboj. R (2010) A Rule Based Approach for Spam Detection. Master Thesis: Computer Science and Engineering Department - Thapar University, Patiala. http://dspace.thapar.edu:8080/jspui/bitstream/10266/1166/3/1166.pdf

{8}Stern. H (2008) A Survey of Modern Spam Tools. 5th Conference on Email and Anti-Spam, CEAS, California.

{9}, {11} Guzella. T. S, and Caminhas. W. M (2009) A Review of Machine Learning Approaches to Spam Filtering. *Expert Systems with Applications*, 36, 10206-10222. http://dx.doi.org/10.1016/j.eswa.2009.02.037

{10}Su. M-C, Lo. H-H, and Hsu. F-H (2010) A Neural Tree and Its Application to Spam E-Mail Detection. Expert Systems with Applications, 37, 7976-7985. http://dx.doi.org/10.1016/j.eswa.2010.04.038

{12}Idris. I, Selamat. A, Thanh-Nguyen. N, Omatu. S, Krejcar. O, Kuca. K, and Penhaker. M (2015) A Combined Negative Selection Algorithm-Particle Swarm Optimization for an Email Spam Detection System. *Engineering Applications of Artificial Intelligence*, 39, 33-44. http://dx.doi.org/10.1016/j.engappai.2014.11.001

{13}Caruana. G, and Li. M (2012) A Survey of Emerging Approaches to Spam Filtering. *ACM Computing Surveys* (*CSUR*), Vol. 44, No 2: 9. http://dx.doi.org/10.1145/2089125.2089129

{14}Santhi. G, Wenisch. S.M, and Sengutuvan. P (2013) A Content Based Classification of Spam Mails with Fuzzy Word Ranking. IJCSI International Journal of Computer Science Issues, 10, 48-58.

{15}, {19}Symantec (2013) Internet Security Threat Report 2013

{16}Luckner. M, Gad. M, and Sobkowiak. P (2014) Stable Web Spam Detection Using Features Based on Lexical Items. Computers & Security, 46, 79-93. http://dx.doi.org/10.1016/j.cose.2014.07.006

{17}Gudkova. D (2013) Kaspersky Security Bulletin. Spam Evolution 2013.

{18}Alazab. M, and Broadhurst. R (2014) Spam and Criminal Activity. Trends and Issues (Australian Institute of Criminology), Forthcoming.
http://papers.ssrn.com/sol3/papers.cfm?abstract_id=2467423

{20}Tran. K-N, Alazab. M, and Broadhurst. R (2013) Towards a Feature Rich Model for Predicting Spam Emails Containing Malicious Attachments and URLs. 11th Australasian Data Mining Conference, Canberra.

{21}Amin. R.M (2011) Detecting Targeted Malicious Email through Supervised Classification of Persistent Threat and Recipient Oriented Features. The George Washington University, Washington DC.

{22}Sahami. M, Dumais. S, Heckerman. D, and Horvitz. E (1998) A Bayesian approach to filtering junk e-mail.

{23}Sharma. S, and Arora. A (2013) Adaptive Approach for Spam Detection. IJCSI International Journal of Computer Science Issues, Vole 10, Issue 4, No1: 23-26.

{24}Judea. P (2000) Causality: Model Sasoning and Inference. Cambridge University Press. ISBN 0-521-77362-8.

{25}Breiman. L (2001) Random Forests. Machine Learning 45 (1): 5–32.10.1023/a: 1010933404324

{26}Cortes. C, and Vapnik. V. N (1995) Support-Vector Networks. Machine Learning, 20: 273-297.

{27}Rumelhart. E. D, Hinton. E. G, and Williams. J. R (1985) Learning Internal Representations by Error Propagation. Institute for Cognitive Science, University of California, San Diego.

{28}Santiago. C. O. B (2015) Machine Learning Bloks. Massachusetts Institute of Technology. http://groups.csail.mit.edu/EVO-DesignOpt/groupWebSite/uploads/Site/Machine_Learning_Blocks___Bryan_Thesis.pdf

{29}Shirani-Mehr. H (2013) SMS Spam Detecting using Machine Learning Approach. Tech. rep., Stanford University. http://cs229.stanford.edu/proj2013/ShiraniMehr-SMSSpamDetectionUsingMachineLearningApproach.pdf

{30}Delany. J. S, Buckley. M, and Greene. D (2012) SMS spam filtering: Methods and data. Expert system with Applications, Vol 39, Issue 10: 9899-9908.

{31}Almeida. A. T, Gomez. M. J, and Yamakami. A (2011) Contributions to the Study of SMS Spam Filtering: New Collection and Results. DocEng 11, Mountain View, California, USA. http://www.dt.fee.unicamp.br/~tiago/smsspamcollection/doceng11.pdf

{32}Xiong. H, Li. M, Jiang. T, and Zhao. Sh (2013) Classification Algorithm based on NB for Class Overlapping Problem. Applied Mathematic & Information Sciences 7, No. 2l: 409-415. http://www.naturalspublishing.com/files/published/kl69zaz4562d9b.pdf

{33}Bauer. E, and Kohavi. R (1998) An Empirical Comparison of Voting Classification Algorithms: Bagging, Boosting, and Variants. Kluwer Academic Piblishers, Boston. Machine Learning, vv, 1-38. http://robotics.stanford.edu/~ronnyk/vote.pdf

{34}Machova. K, Bracak. F, and Bendar. P (2006) A Bagging Method using Decision Trees in the Role of Base Classifiers. Acta Polytechnic Hungarica Vole. 3, No 2: 121-132. http://epa.oszk.hu/02400/02461/00006/pdf/EPA02461_acta_polytechnica_hungarica_2006_02_121-132.pdf

{35}Machova. K, Puszta. M, and Bednar. P (2005) A Boosting method in Combination with Decision Trees. https://www.researchgate.net/profile/Kristina_Machova/publication/239603435_A_Boosting_method_in_Combination_with_Decision_Trees/links/0deec529c68d377a62000000.pdf