International Journal of Advanced Computer Science and Information Technology (IJACSIT)
Vol. 3, No. 3, 2014, Page: 306-316, ISSN: 2296-1739

www.elvedit.com

Structured Stream Data Mining Using SVM Method as Basic Classifier

Authors

Hadi Barani Baravati *baranihadi1360@gmail.com*
Department of Computer Engineering/ Islamic Azad University, Iranshahr Branch, Iran *Iranshahr, Iran*

Javad Hosseinkhani *jhkhani@gmail.com*
Department of Computer Engineering/ Islamic Azad University, Science and Research Branch, Iran *Zahedan, Iran*

Solmaz Keikhaee *solmaz.keikhaee@gmail.com*
Department of Electrical Engineering/ Islamic Azad University, Science and Research Branch, Iran *Zahedan, Iran*

Javid Hosseinkhani Naniz *hosseinkhani.javid@yahoo.com*
Department of Computer Engineering/ Islamic Azad University, Kerman Branch, Iran *Kerman, Iran*

Abstract

Recently, the huge number of email spams has caused serious problems in essential email communication. In this paper, we describe the results of an empirical study on one spam detection method namely Support Vector Machines (SVMs). To conduct the study, first the recieved emails would be pre proccessed then stream data in order to learning the classification would be given to the proposed data miner system. The number of training data set with window based solution will be selected with default , W=100 , the first 100 data would be used as trainig set. each receieved email input to SVM to be classified in to 2 predefined classes named: Non spam, and Spam. A program is written that 4 different kinds of time window in order to SVM training are selected (100,200,500 and all the preset data or open window). The evaluation criteria include accuracy rate, recall, and precision rate. The results indicate that the approach has its pros and cons.

Key Words

Spam Detection, Email Classification, Support Vector Machine.

I. INTRODUCTION

Most Internet users use mail to communicate electronically. They depend on the mail system to deliver their mails to the recipient. Spam has made the mail system more unreliable because mail can get falsely caught by spam filters on the way to the recipient or mail can drown among spam in the recipient's inbox. The goal of the Internet community should be to work toward a more usable Internet with less spam. Possible ways to do this are through the law and the legal system, technical solutions and user awareness.

Recently, the huge number of email spams has caused serious problems in essential email communication. Traditional spam filters aim at analyzing email content to characterize the features that are commonly included in spams. However, it is observed that crafty tricks designed to avoid content-based filters will be endless owing to the economic benefits of sending spams. In view of this situation, there has been much research effort toward doing spam detection based on the reputation of senders rather than what is contained in emails. Motivated by the fact that spammers are prone to have unusual behavior and specific patterns of email communication, exploring email social networks to detect spams has received much attention [19].

Email communication is prevalent and indispensable nowadays. However, the threat of unsolicited junk emails, also known as spams, is increasingly serious. According to a survey by the website Top Ten REVIEWS [19], 40% (12.4 billion out of 31 billion per day) of emails were considered as spams in 2006. The statistics collected by Message- Labs show that spam rate persistently remains high. The primary challenge for spam detection problem lies in the fact that for the purposes of gaining economic benefits, distributing spyware, and spreading links to phishing websites, to name a few, spammers will always develop new sophisticated approaches to attack spam filters. For example, traditional text-based filters, such as Naive Bayes classifiers, have been commonly passed by obfuscated keywords and random paragraph insertion. In addition to essential information that spammers want to convey, there are various unrelated contents included in spams. Since unexpected tricks employed in email content are ceaseless, a number of studies have focused on identifying who sends the email rather than what is contained in the email.

On-line data stream mining has attracted much research interest, but systems that can be used as a workbench for online mining have not been researched, since they pose many difficult research challenges [18]. On-line data stream mining plays a key role in growing number of real-world applications, including network traffic monitoring, intrusion detection, web click-stream analysis, and credit card fraud detection. Thus, many research projects have recently focused on designing fast mining algorithms, whereby massive data streams can be mined with real-time response [10, 8, 9, and 7]. Similarly, many research projects have also focused on managing the data streams generated from these applications [11, 12, and 13]. However, the problem of

supporting mining algorithms in such systems has, so far, not received much research attention [14]. This situation seems unusual, since the need for a mining system for static data mining, was immediately recognized [17] and has lead to systems such as, Weka [15] and OLE DB for DM [16]. Furthermore, static mining algorithms can also be written in procedural language using a cache mining approach that makes little use of DBMS essentials. However, online mining tasks cannot be deployed as stand-alone algorithms, since they require many DSMS essentials, such as I/O buffering, windows, synopses, load shedding, etc. Clearly, KDD researchers and practitioners would rather concentrate on the complexities of data mining tasks and avoid the complexities of managing data streams, by letting the mining system handle them. In short, while mining systems are a matter of convenience for stored data, they are a matter of critical necessity for data streams.

A method of ordering linear and nonlinear data is Support vector machines (SVMs). In a case, SVM is an algorithm and the function of it is as follows. To change the original training data into a higher dimension, it applies a nonlinear mapping. It seeks for the linear ideal separating hyper plane through this new dimension. A hyper plane can always separate the data into two classes with a suitable nonlinear mapping to an appropriately high dimension. The SVM discovers this hyper plane utilizing support vectors that is “essential” training tuples and margins which is explained by the support vectors. Vladimir Vapnik and colleagues Isabelle Guyon and Bernhard Boser (1992) have done the first research on support vector machines since the groundwork for SVMs has been around. Even though the training time of SVMs is very extremely slow, they are very precise and can to model compound nonlinear decision limitations. In compare to other methods, they are much less predisposed to over fitting. The provision vectors also are a compressed explanation of the trained model. SVMs also are able to utilize for numeric calculation along with classification. They have been used for many areas such spam email detection. [6].

There are many anti-spam strategies and methods. In this paper, we describe the results of an empirical study on one spam detection method namely Support Vector Machines (SVMs). The reason for choosing this method is that it has good theoretical foundation, scale up well with large data, and lend itself to the text classification problem. In our study, we implemented an application of SVMs. To conduct the study, first the recieved emails would be pre proccessed then stream data in order to learning the classification would be given to the proposed data miner system. The number of training data set with window based solution will be selected with default , W=100 , the first 100 data would be used as trainig set. each receieved email input to SVM to be classified in to 2 predefined classes named: Non spam, and Spam. A program is written that 4 different kinds of time window in order to SVM training are selected (100,200,500 and all the preset data or open window). The evaluation criteria include accuracy rate, recall, and precision rate. The results indicate that the approach has it pros and cons.

II. RELATED WORKS

Since the email spam problem is more and more serious nowadays, various techniques have been explored to relieve this problem. According to what features of emails are being used, previous works on spam detection can be generally classified into three categories: (1) content-based methods, (2) non-content-based methods, and (3) integrated methods. Initially, researchers analyze email content text and model this problem as a binary text classification task. Representatives of this category are Naive Bayes [20, 21] and Support Vector Machines (SVMs) [22, 23] methods. In general, Naive Bayes methods train a probability model using classified emails, and each word in emails will be given a probability of being a suspicious spam keyword. As for SVMs, it is a supervised learning method, which possesses outstanding performance on text classification tasks. Traditional SVMs [23] and improved SVMs [22] have been investigated. While above conventional machine learning techniques have reported excellent results with static data sets, one major disadvantage is that it is cost prohibitive to constantly re-train these methods with the latest information to adapt to the rapid evolving nature of spams. Moreover, crafty content obfuscation tricks have always been developed to degrade the performance of these approaches. On the other hand, certain specific features such as URLs [24] and images [25] are also taken into account for spam detection.

The other group attempts to exploit non-content information such as email header, email traffic [26], and email social network [27, 28] to filter spams. Collecting notorious and innocent sender addresses (or IP addresses) from email header to create black list and white list is a commonly applied method initially. In [26], the authors intend to analyze email traffic flows to detect suspicious machines and abnormal email communication. It is noted that these approaches have to operate in coordination with other complementary methods to gain better results. Moreover, in [29], a pure reputation system is designed to apply in a large webmail service. This system is constructed by the past behavior of each sender with SPF and DomainKey authentication.

Furthermore, some researchers consider combining the merits of several techniques [30, 31, 32]. Even though the performance of classifier integration seems prominent, there is still no conclusion on what is the best combination. In addition, how to efficiently update the whole included classifiers is another unsolved issue.

In [33], certain network related features are extracted to characterize each user. A modified k-Nearest Neighbor (k-NN) model is then employed to perform the spam classification. In [34], graph theoretical analysis of networks is presented to discover good discriminators between legitimate emails and spams. In [35], the authors propose an email scoring mechanism that infers reputation ratings between individuals in networks. In [27], the authors exploit the feature of clustering coefficient in networks to devise a detection mechanism. Overall, these works generally suffer from the following two problems. First, they are not robust in diverse

environments. The other is that the update scheme, which is critical for evolving networks, has been ignored in these works.

III. SUPPORT VECTOR MACHINES

We consider spam detection as a text classification problem. There are two classes for email messages: y_i€ {-1, +l} where -1 indicates no spam and +1 spam. A feature is a word in an email message and a feature vector x_i represents an email in the feature space. Given ***n*** labeled training examples: **(x_1, y_1),** ..., **(x_n, y_n),** the task is to learn from the training examples a hypothesis that can be used to classify unseen email messages.

Support vector machines are a family of learning methods [l]. Linear hard-margin SVMs are the simplest model in SVMs and *are* also called the maximal margin classifier which works only for data linearly separable in the feature space. The linear hard-margin SVMs separate feature vectors into the two classes by finding a hyper plane with maximal margin. The feature vectors closest to the hyper plane are called support vectors. The maximal margin hyper plane bounds the generalization error of the linear machines given **a** training set S, and can be obtained by maximizing the function

$$W(\alpha) = \sum_{i=1}^{l} a_i - \frac{1}{2} \sum_{i=1}^{l} \sum_{j=1}^{l} a_i a_j y_i y_j x_i . x_j$$

Subject to:

$$\sum_{i=1}^{l} y_i a_i = 0, a_i \geq 0.$$

Soft-margin SVMs [1] can be used for non-linearly separable data. Soft-margin SVMs allow training errors. The optimization problem now becomes maximizing the following:

$$W(\alpha) = \sum_{i=1}^{l} a_i - \frac{1}{2} \sum_{i=1}^{l} \sum_{j=1}^{l} a_i a_j y_i y_j x_i . x_j$$

Subject to:

$$\sum_{i=1}^{l} y_i a_i = 0, 0 \leq a_i \leq C.$$

The *C* is the parameter that we need to tune to make the model fit to the non-linearly separable data. The soft margin SVMs behave like hard-margin SVMs if the parameter *C* is large enough. See [2, 3, 4, and 5] for details.

IV. Research Design and Proposed Framework

According to Figure 1, first the recieved emails would be pre proccessed then stream data in order to learning the classification would be given to the proposed data miner system. The number of training data set with window based solution will be selected with default , W=100 , the first 100 data would be used as trainig set.

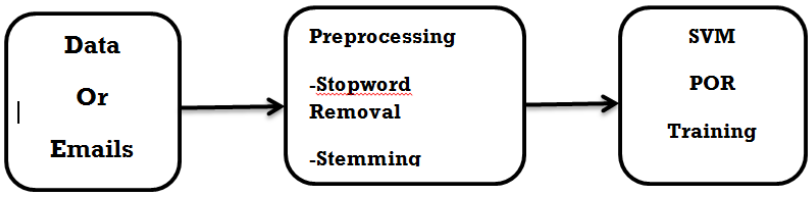

Figure 1: Proposed Framework

Before the emails or Data are used for retrieval, some preprocessing tasks are usually performed. The tasks are stopword removal and stemming. Stopwords are frequently occurring and insignificant words in a language that help construct sentences but do not represent any content of the documents. Articles, prepositions and conjunctions and some pronouns are natural candidates. In many languages, a word has various syntactical forms depending on the contexts that it is used.

For example, in English, nouns have plural forms, verbs have gerund forms (by adding "*ing*"), and verbs used in the past tense are different from the present tense. These are considered as syntactic variations of the same root form. Such variations cause low recall for a detection system because a relevant spam email may contain a variation of a query word but not the exact word itself. This problem can be partially dealt with by stemming.

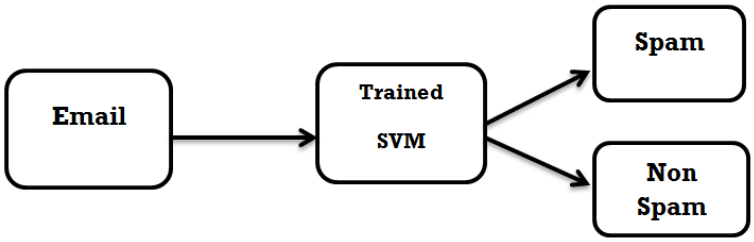

FIGURE 2: CLASSIFICATION OF RECEIVED EMAILS TO SVM

Figure 2 illustrates that each receieved email input to SVM to be classified in to 2 predefined classes named: Non spam, and Spam. Data set Usnet1 and Usenet 2 are applied inroder to training and proposed data miner learning .

V. EXPERIMENTAL RESULT

Table 1 shows the stream data classificiation exprimental results through using SVM in Spam data set. In this table, 4 different kinds of time window in order to SVM training are selected (100,200,500 and all the preset data or open window) that 3 evaluations criteria's including precision , recall and accuracy are evaluated that is shown in table 1.

TABLE1: STREAM DATA CLASSIFICIATION EXPRIMENTAL RESULTS

	Precision	**Recall**	**Accuracy**
Simple Incremental	0.9987	0.9172	0.9320
Time Windows(W=100)	0.9600	0.9165	0.9140
Time Windows(W=200)	0.9660	0.8954	0.9050
Time Windows(W=500)	0.9937	0.8275	0.8990

Figure 3 shows the mean of vector precision for 1000 experimental samples.

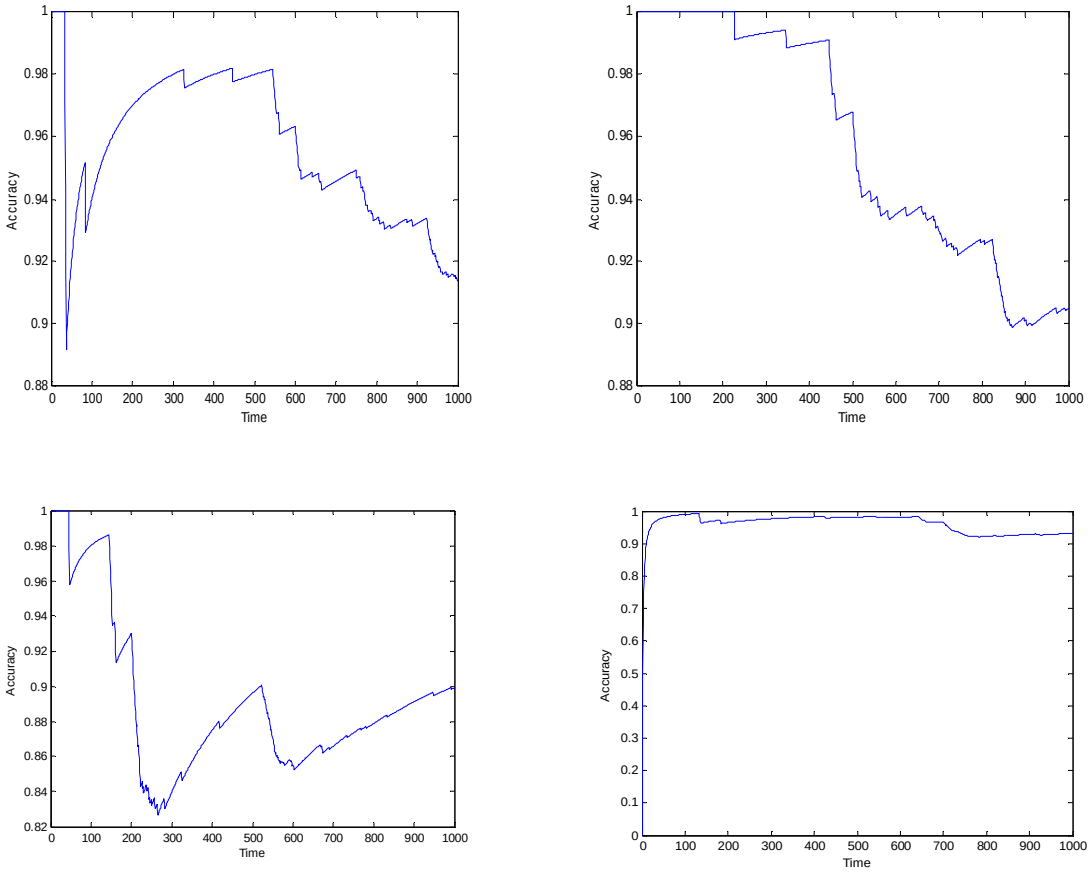

FIGURE III: THE MEAN OF VECTOR PRECISION FOR 1000 EXPERIMENTAL SAMPLES.

VI. CONCLUSION

On-line data stream mining has attracted much research interest, but systems that can be used as a workbench for online mining have not been researched, since they pose many difficult research challenges. On-line data stream mining plays a key role in growing number of real-world applications, including network traffic monitoring, intrusion detection, web click-stream analysis, and credit card fraud detection. Thus, many research projects have recently focused on designing fast mining algorithms, whereby massive data streams can be mined with real-time response.

In our experiments, the linear SVMs have several advantages. One important feature is that SVMs are less influenced by the sizes of training cases in the two classes because they are not geared toward minimizing the error rate, but instead attempt to separate the patterns in feature space. However, the performance is the potential issue. If there are large numbers of training cases, learning process can be long. Execution can be slow for nonlinear SVMs.

To get more accurate results, such study needs to be repeated on larger data sets. It is also clear in our study that the performance of those classification methods really depend on the training examples, i.e. the feature vectors extracted from the original email messages. In the experiments, only words in the message body were used as the candidates of the features. More important information might be missed in the feature extraction process. For example, the subject title of email can be the good candidate of the features. And also, recent spam messages are coded with html so it might be a good idea to include the html codes in the features. The preprocessing before running the learners is the important phase for the learners to perform better classification. For the further analyses, the spam emails for training and testing should be decomposed into the multiple classes according to the kinds of spam emails such as investment and vacations. This makes the analysis results more useful and refined.

REFERENCES

[1] Matsumoto, Ryota, Du Zhang, and Meiliu Lu. "Some empirical results on two spam detection methods." *Information Reuse and Integration, 2004. IRI 2004. Proceedings of the 2004 IEEE International Conference on*. IEEE, 2004.

[2] T. Joachims, Learning To Classify Text Using Support Vector Machines, Kluwer Academic Publishers, 2002**.**

[3] E. Osuna, R. Freund, and E Girosi, Improved Training Algorithm for Support Vector Machines, Roc. IEEE NNSP'97, pp.276-285, 1997.

[4] J**.** C. Platt, Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines, Advances in Kernel Method: Support Vector Learning, by Scholkopf, Burges, and Smola, MIT Press, pp**.** 185- 208,1998.

[5] V**.** Vapnik, Estimation of Dependencies Based on Empirical Data, Springer-Verlag, New York, 1982.

[6] Moradi Koupaie, Hossein, Suhaimi Ibrahim, and Javad Hosseinkhani. "Outlier Detection in Stream Data by Machine Learning and Feature Selection Methods." *International Journal of Advanced Computer Science and Information Technology (IJACSIT) Vol* 2 (2014): 17-24.

[7] B. Mozafari, H. Thakkar, and C. Zaniolo. Verifying and mining frequent patterns from large windows over data streams. In *ICDE*, 2008.

[8] F. Chu and C. Zaniolo. Fast and light boosting for adaptive mining of data streams. In *PAKDD*, volume 3056, 2004.

[9] H. Wang and et al. Mining concept-drifting data streams using ensemble classifiers. In *SIGKDD*, 2003.

[10] George Forman. Tackling concept drift by temporal inductive transfer. In *SIGIR*, pages 252–259, 2006.

[11] Yan-Nei Law, Haixun Wang, and Carlo Zaniolo. Data models and query language for data streams. In *VLDB*, 2004.

[12] A. Arasu, S. Babu, and J.Widom. CQL: A language for continuous queries over streams and relations. In *DBPL*, 2003.

[13] D. Abadi et al. Aurora: A new model and architecture for data stream management. *VLDB Journal*, 2003.

[14] H. Thakkar, B. Mozafari, and C. Zaniolo. Designing an inductive data stream management system: the stream mill experiences. In *Scalable Stream Processing Systems*, 2008.

[15] Weka 3: data mining with open source machine learning software in java. http://www.cs.waikato.ac.nz.

[16] Z. Tang and et al. Building data mining solutions with OLE DB for DM and XML analysis. *SIGMOD Record*, 2005.

[17] Tomasz Imielinski and Heikki Mannila. A database perspective on knowledge discovery. *Commun. ACM*, 1996.

[18] Thakkar, Hetal, Barzan Mozafari, and Carlo Zaniolo. "A data stream mining system." *Data Mining Workshops, 2008. ICDMW'08. IEEE International Conference on*. IEEE, 2008.

[19] Tseng, Chi-Yao, and Ming-Syan Chen. "Incremental SVM model for spam detection on dynamic email social networks." *Computational Science and Engineering, 2009. CSE'09. International Conference on*. Vol. 4. IEEE, 2009.

[20] J. Hovold. Naive bayes spam filtering using wordposition- based attributes. *Proc. of the 2nd Conference on Email and Anti-Spam (CEAS)*, 2005.

[21] V. Metsis, I. Androutsopoulos, and G. Paliouras. Spam filtering with naive bayes – which naive bayes *Proc. of the 3rd Conference on Email and Anti-Spam (CEAS)*, 2006.

[22] E. Blanzieri and A. Bryl. Evaluation of the highest probability svm nearest neighbor classifier with variable relative error cost. *Proc. of the 4th Conference on Email and Anti-Spam (CEAS)*, 2007.

[23] H. Drucker, D. Wu, and V. N. Vapnik. Support vector machines for spam categorization. *Proc. of the IEEE Transactions on Neural Networks*, pages 1048–1054, 1999.

[24] K. M. Schneider. Brightmail url filtering. *Proc. of the MIT Spam Conference*, 2004.

[25] M. Dredze, R. Gevaryahu, and A. Elias-Bachrach. Learning fast classifiers for image spam. *Proc. Of the 4th Conference on Email and Anti-Spam (CEAS)*, 2007.

[26] R. Clayton. Email traffic: A quantitative snapshot. *Proc. of the 4th Conference on Email and Anti-Spam (CEAS)*, 2007.

[27] P. O. Boykin and V. Roychowdhury. Leveraging social networks to fight spam. *Proc. of the IEEE Computer*, 2005.

[28] P.-A. Chirita, J. Diederich, and W. Nejdl. Mailrank: Using ranking for spam detection. *Proc. of the 14th ACM International Conference on Information and Knowledge Management (CIKM)*, pages 373–380, 2005.

[29] B. Taylor. Sender reputation in a large webmail service. *Proc. of the 3rd Conference on Email and Anti- Spam (CEAS)*, 2006.

[30] S. Hershkop and S. J. Stolfo. Combining email models for false positive reduction. *Proc. of the 11th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 98–107, 2005.

[31] T. R. Lynam and G. V. Cormack. On-line spam filter fusion. *Proc. of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 123–130, 2006.

[32] R. Segal. Combining global and personal anti-spam filtering. *Proc. of the 4th Conference on Email and Anti-Spam (CEAS)*, 2007.

[33] H.-Y. Lam and D.-Y. Yeung. A learning approach to spam detection based on social networks. *Proc. Of the 4th Conference on Email and Anti-Spam (CEAS)*, 2007.

[34] L. H. Gomes, R. B. Almeida, L. M. A. Bettencourt, V. Almeida, and J. M. Almeida. Comparative graph theoretical characterization of networks of spam and legitimate email. *Proc. of the 2nd Conference on Email and Anti-Spam (CEAS)*, 2005.

[35] J. Golbeck and J. Hendler. Reputation network analysis for email filtering. *Proc. of the 1st Conference on Email and Anti-Spam (CEAS)*, 2004.