

Learning to Predict Bad Behavior

Nadeem Ahmed Syed, Nick Feamster, Alex Gray
College of Computing, Georgia Tech, Atlanta, GA 30332
{nadeem, feamster, agray}@cc.gatech.edu

1 Introduction

The current spam-filtering techniques rely mainly on either 1) content-based filtering, or 2) IP blacklisting, or a combination of the two. A variety of pattern classification and text categorization methods have been successfully applied to content-based spam filtering systems (see ICML 2004 tutorial [4] for a summary of current approaches). Both content-based filtering and IP blacklisting approaches face challenges as spammers get more sophisticated. With the use of images, pdf files and misleading keyword patterns etc. by the spammers, the content-based filters are always in an arms race with the spammers. Similarly, the IP blacklists are rendered less effective by use of dynamic IPs for spamming and use of large spam-bots [6].

There has been some recent evidence that network level behavior of spammers tends to differ from the good senders [6, 7, 9]. In [7], the authors propose the idea of *behavioral blacklisting*. Based on their earlier findings in [6], they argue that the spamming IPs have a different sending pattern from non-spammers if we were to look at their messaging pattern across a set of target domains to which they send emails.

In this report we discuss our success in designing good feature sets for *behavioral spam filtering*. We use support vector machines to build a spam classifier and obtain very encouraging results.

2 Designing features for spam classification using network-level behavior

In content-based filtering the training data starts off as the original emails and then transformed into a feature vector for machine learning methods using text analysis techniques. On the other hand, for network level behavioral spam filtering we need to start with server/data aggregator logs that look like: $\langle \text{sender-ip}, \text{target-domain}, \text{time-stamp class} \rangle$, where the sender-ip is the IP of the email server that sent the message, target-domain is the domain of the recipient's email address, time-stamp is the time at which this email message was logged and class is spam/non-spam label. For the current study we analyze two such logs from two different email hosting organizations. For privacy reasons, we will simply refer to these datasets as *DS1* and *DS2*.

Since this is a completely new approach to the problem, it is not well understood yet as to how these logs can be best transformed into suitable feature sets that will provide good discriminatory power. One of our main lines of investigation has been exploratory data analysis [8] on such logs to understand how the spammers' behavior differs from the non-spammers at this level and design relevant and effective feature sets. From [7] we already know that volume of email sent to various target domains provides some discriminatory ability. At this juncture we have unearthed some other interesting properties: 1) Since spammers tend to predominantly use dynamic IPs and botnets to send messages and hop around to different IPs, we would expect to see many more IPs from the subnet or IP block used by spammers when compared to valid senders. This difference is clearly brought out by the histogram of the frequency of "number of IPs from a single /24 subnet" for each class of senders. Figure 1(a) and 1(b) show these plots for the *DS1* dataset. The horizontal axis has number of IPs from a single subnet (from 0 to 253) and y axis has the corresponding frequency. 2) The *diurnal sending pattern* of spammers differs somewhat from the legitimate senders. While legitimate emails messages tend to peak during two periods of the day, the spammers tend to have a more smooth volume transition. Figure 1(c) shows a plot of email volume at different times of the day on *DS2*. 3) While ham volume tends to drop significantly during the weekends, the spam tends to be reasonably constant throughout. Figure 1(d) shows this pattern for *DS2*.

The above findings are utilized to construct features of four different types: 1) log-volume of email from a single IP to each of the target domains. For *DS1* we have 29 target domains that are common to spam and ham classes and for *DS2* we have 90 domains, 2) log-volume of email at different times of the day in 3 hour bins, 3) log-volume of email on different days of the week, 4) Number of IPs seen from the current IP's /24 subnet, normalized to lie between 0 and 1.

We used this combination of features to train a support vector machine with RBF kernel [1] to predict whether a given IP is spam-sender and carried out 10-fold cross-validation to calculate the prediction accuracy. Our experiments showed that combination of features 1, 2, and 4 provided the best performance in terms of prediction accuracy. Addition of feature 3 doesn't help much. While it slightly improves the prediction accuracy on the training set, the prediction accuracy on the test set gets a little worse. It appears that adding in feature 3 just increases the dimensionality and over-fitting without providing any new discriminatory power. Table 1 shows the confusion matrix for SVM prediction using a combination of features 1, 2 and 4 on the two datasets. These tables also reveal that the features we have selected are robust across datasets and the fact the network level behavioral spam filtering can be applied effectively to datasets from different service providers.

Given the potentially noisy data labeling, the prediction accuracy looks very encouraging but the false positive rate is clearly not acceptable if the classifier was to be used by itself for spam filtering. Seeking a simple approach to deal with the high false-positive rate problem, we explored three class (spam, ham, don't-know) prediction using probabilistic SVM [5].

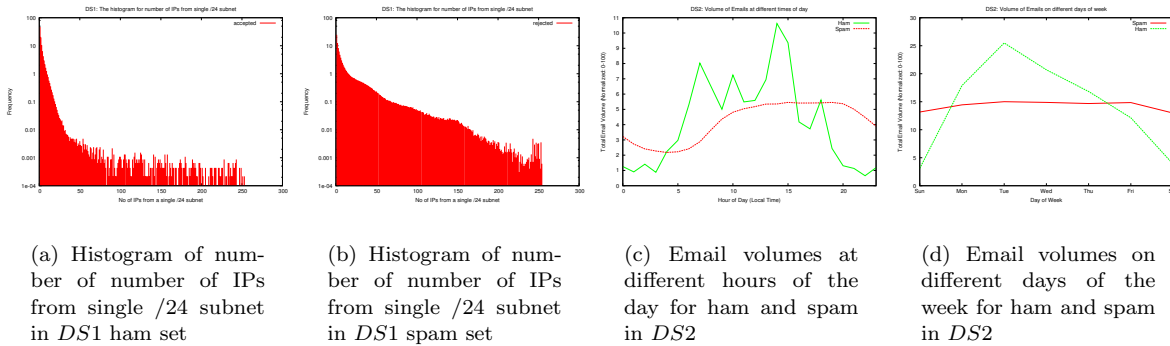


Figure 1: Various plots showing network-level behavioral differences between ham and spam

	Classified as			Classified as	
	Spam	Ham		Spam	Ham
Spam	14268 (71.34%)	5732 (28.66%)	Spam	1217 (84.69%)	220 (15.30%)
Ham	2402 (12.01%)	17598 (87.99%)	Ham	266 (18.51%)	1171 (81.49%)

Confusion Matrix for DS1

Confusion Matrix for DS2

Table 1: Confusion Matrix for SVM Prediction using RBF Kernel

	Classified as				Classified as		
	Spam	Ham	Don't Know		Spam	Ham	Don't know
Spam	10711 (53.55%)	4731 (23.66%)	4558(22.79%)	Spam	703 (48.92%)	243 (16.91%)	491(34.17%)
Ham	839 (4.19%)	17095 (85.48%)	2066 (10.33%)	Ham	59 (4.11%)	1191 (82.88%)	187 (13.01%)

Confusion Matrix for DS1 (threshold $p = 0.80$)

Confusion Matrix for DS2 (threshold $p = 0.86$)

Table 2: Confusion Matrix for 3-class prediction using probabilistic SVM

Since we are mostly interested in moving the false positives to the “don’t know” basket, only the boundary for the spam class is moved such that an email is classified as ham if the SVM predicts it to be ham irrespective of the class probabilities. On the other hand an email is considered spam only if the probability of spam class is above a certain threshold. Table 2 shows the confusion matrix using this approach. While the false-positive rate is significantly lower, about 50% spam still gets caught. We are currently working on training a second stage classifier to classify the senders in the don’t know basket.

3 Ongoing and future work

This is an ongoing effort and we are still exploring multiple avenues to build a reliable, robust behavioral spam filter with low false-positive rate. One property of this problem is that the data is imbalanced, in that, we have about 10 times as many spam examples as the ham examples. Furthermore the cost-metric is asymmetric since the cost of false-positive is very high as we cannot afford to lose any valid email. We are currently exploring methods for cost-sensitive learning [2] as well ensemble methods like bagging and boosting [3]. We are also investigating whether the behavioral filtering tends to be *orthogonal* in some sense to the content-based filtering. If that is the case, we can fruitfully combine the two approaches to get a robust and highly accurate spam-filter.

References

- [1] C.J.C. Burges. A Tutorial on Support Vector Machines for Pattern Recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167, 1998.
- [2] C. Elkan. The foundations of cost-sensitive learning. In *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence*, 2001.
- [3] T. Hastie, R. Tibshirani, J.H. Friedman, et al. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, 2001.
- [4] E. Hulton and J. Goodman. Tutorial on junk email filtering. *ICML*, 2004.
- [5] J. Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in Large Margin Classifiers*, 10(3), 1999.
- [6] A. Ramachandran and N. Feamster. Understanding the network-level behavior of spammers. In *Proceedings of the ACM SIGCOMM*. ACM Press New York, NY, 2006.
- [7] A. Ramachandran, N. Feamster, and S. Vempala. Filtering spam with behavioral blacklisting. In *ACM Conference on Computer and Communications Security*. ACM Press New York, NY, 2007.
- [8] J.W. Tukey. *Exploratory data analysis*. Addison-Wesley Menlo Park, CA, 1977.
- [9] Y. Xie, Yu F., Achan Y., Gilum E., Goldszmidt M., and Wobber T. How dynamic are ip addresses. In *Proceedings of the ACM SIGCOMM*, 2007.