A Case for Unsupervised-Learning-based Spam Filtering

Feng Qian[1], Abhinav Pathak[2], Y. Charlie Hu[2], Z. Morley Mao[1], and Yinglian Xie[3]
[1]University of Michigan [2]Purdue University [3]Microsoft Research

Categories and Subject Descriptors: C.2.0 [Computer Communication Networks]: General – Security and protection

General Terms: Design, Security

Keywords: SpamCampaignAssassin (SCA), Spam campaign, unsupervised learning, latent semantics analysis (LSA)

1. INTRODUCTION

Traditional content-based spam filtering systems rely on supervised machine learning techniques. In the training phase, labeled email instances are used to build a learning model (*e.g.,* a Naive Bayes classifier or support vector machine), which is then applied to future incoming emails in the detection phase. However, the critical reliance on the training data becomes one of the major limitations of supervised spam filters. Preparing labeled training data is often labor-intensive and can delay the learning-detection cycle. Furthermore, any mislabeling of the training corpus (*e.g.,* due to spammers' obfuscations) can severely affect the detection accuracy.

Supervised learning schemes share one common mechanism regardless of their algorithm details: learning is performed on an individual email basis. This is the fundamental reason for requiring training data for supervised spam filters. In other words, in the learning phase these classifiers can never tell whether an email is spam or ham because they examine one email instance at a time.

We investigate the feasibility of a completely unsupervised-learning-based spam filtering scheme which requires no training data. Our study is motivated by three key observations of the spam in today's Internet. (1) The vast majority of emails are spam. (2) A spam email should always belong to some campaign [2, 3]. (3) The spam from the same campaign are generated from templates that obfuscate some parts of the spam, *e.g.,* sensitive terms, leaving the other parts unmodified [3]. These observations suggest that in principle we can achieve unsupervised spam detection by examining emails at the campaign level. In particular, we need robust spam identification algorithms to find common terms shared by spam belonging to the same campaign. These common terms form signatures that can be used to detect future spam of the same campaign.

This paper presents SpamCampaignAssassin (SCA), an online unsupervised spam learning and detection scheme. SCA performs accurate spam campaign identification, campaign signature generation, and spam detection using campaign signatures. To our knowledge, SCA is the first unsupervised spam filtering scheme that achieves accuracy comparable to the *de-facto* supervised spam filters by explicitly exploiting online campaign identification. The full paper describing SCA is available as a technical report [4].

SIGMETRICS'10, June 14–18, 2010, New York, New York, USA.
ACM 978-1-4503-0038-4/10/06.

2. SCA: UNSUPERVISED SPAM FILTERING

We describe SCA by highlighting how we address four fundamental challenges of designing an unsupervised spam filtering scheme.

SCA's online operational model. The first challenge is that SCA should operate in an online manner in order to quickly capture campaigns at their onset. Figure 1 shows the online signature generation and spam detection scheme of SCA, which operates in a "split-and-merge" manner. The incoming email stream is split into multiple *segments*, each consisting of emails received in contiguous duration. The signature generation component processes the input email stream at the unit of one segment. As emails are incoming, whenever a new full segment is ready, it is passed to the signature generation component for processing. The processing results, *i.e.,* preliminary identified clusters and signatures, for recent individual segments are merged at low cost to generate purified final signatures. To avoid generating stale SC signatures, the signature generation component merges preliminary identified signatures in a sliding window of w most recent segments. Note that each segment needs to be processed only once even though it stays in the sliding window w times. Such a "split-and-merge" design makes it feasible to apply sophisticated but expensive machine learning algorithms such as Latent Semantic Analysis (LSA) [1], as the segment size (the number of emails in a segment) is bounded.

Spam detection is always performed on a per-email basis. For each email, performing textual signature matching for hundreds of signatures may incur non-trivial computational overhead. SCA alleviates such performance overhead by exploiting the heavy-tail distribution of campaign sizes, *i.e.,* a large fraction of spam belong to a small number of campaigns. SCA periodically (every 1K emails) reorders the SC signatures based on their hit counts, *i.e.,* the number of spam they matched. With dynamic signature reordering enabled, up to 47% emails in our dataset are detected as spam when tested against no more than 10 signatures.

Robust signature generation algorithm. The second challenge is that SCA needs a robust signature generation algorithm that can accurately identify common terms shared by spam belonging to the same campaign and consequently the corresponding campaign. We propose a novel online text-mining signature generation framework that takes as input a window of recent w segments of incoming email stream containing spam (possibly from multiple campaigns) and legitimate emails (ham) and outputs textual signatures that separates spam from ham.

As preprocessing, we remove common stop words and low-frequency terms from each segment, which is then tokenized into a *tf-idf* (term frequency - inverse document frequency) matrix. Next, the semantics of invariant texts, which are good indicators of spam campaign templates, are boosted by Latent Semantic Analysis (LSA)[1]. LSA transforms the original vocabulary space into a compact concept

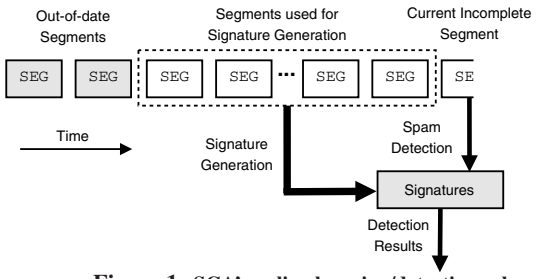

Figure 1: SCA's online learning/detection scheme.

space. For example, if *Rolex*, *Panerai*, and *replica* have high co-occurrence in the corpus, then the three terms are grouped into one "term", which can be thought as "replica watches" in the concept vocabulary. Subsequently, in the concept space we separate low-entropy components (*i.e.,* spam campaigns) by bisect k-means clustering algorithm which, without supervision, finds clusters of documents with similar content. From these clusters, we generate preliminary signatures consisting of frequent term subsequences using the suffix tree algorithm. Then the clusters are *purified* by preliminary signatures and other heuristics such as cluster sizes to effectively reduce false positives. All above operations are performed on a per-segment basis. The final step is called *cluster agglomeration*, which agglomerates all purified clusters from all w segments into final clusters and produces final signatures that are used for detecting spam in future segments.

Handling legitimate campaigns. The third challenge is that legitimate emails that exhibit similarities (*e.g.,* newsletters, broadcast announcements, and mailing lists) may be falsely classified into spam campaigns and contribute to false positives in spam detection. We effectively address this problem using two approaches. First, we eliminate the entire cluster generated by the bisect k-means algorithm if its all messages are originated from no more than u distinct IPs where u is a threshold. Such a cluster is more likely to be a legitimate campaign or a long threaded discussion, since the *de facto* spamming strategy today is to exploit botnets involving a large number of infected hosts. Second, mailing lists may exhibit behaviors very similar to spam campaigns, *i.e.,* messages from heterogeneous sources share similar content. We use a keyword whitelist that is tested against the preliminary signatures to filter clusters containing legitimate mailing list messages.

The Visibility Challenge. SCA is designed to be deployed in a single organization (or mail service provider), which may not witness enough volume or concentration of spam belonging to a campaign for the mining algorithm to detect the campaign (or soon enough). This can contribute to false negatives in spam detection. Such a "visibility challenge" poses perhaps the most significant challenge to unsupervised learning (the same challenge is faced by supervised learning). Our proposed text-mining of email bodies alone generates campaign signatures that cover over 80% of the spam for one data set. To overcome the "visibility challenge", SCA employs two optimization techniques. First, SCA also mines HTML and URL information in email bodies to generate additional campaign signatures to complement the aforementioned textual signatures in order to reduce the false negative rate.

Second, we enhance the online detection process of SCA by adding a self-maintained IP blacklist of end hosts that have originated a high spam-to-ham ratio. This is motivated by the well-known *bimodal* behavior of email sources (the vast majority of IP addresses originate either a spam-to-ham ratio or ham-to-spam ratio close to 0, *i.e.,* their behaviors are either consistently malicious or consistently benign) and by the fact that spamming IPs are known to participate in multiple spam campaigns launched in similar time [2]. For each IP, SCA actively maintains the spam ratio

Table 1: Email traces used for evaluation.

	DEPT	RELAY (dest. to Yahoo)
# Emails	1.68 M	316.4 K
Sampling	1:1	1:10
Time	April - May 2009	February 2009
Source	Dept. mail servers	Open relay
Type	Spam + ham	All spam
FPR	0.3%	N/A
TPR w/self BL	96.5%	99.9%

(the ratio of emails that matched some campaign signatures in the recent w segments). In the detection phase, when an email does not match any of the signatures, SCA checks the spam ratio and the source IP, and flags the email as spam if the ratio is close to 1.

3. EVALUATIONS

We evaluate the online detection results of SCA using two datasets described in Table 1. The DEPT trace was collected from two mail servers serving about 3K users in a large university department. The RELAY trace [2] was a subset of one month's trace collected from an open relay located in Eastern US which has been running since Oct 2007; only emails going to the destination domain of yahoo.com were kept in the subset. We employ the same parameters of SCA for both traces.

Table 1 shows the evaluation results in terms of False Positive Rate (FPR) and True Positive Rate (TPR). SCA itself can detect 92.4% and 99.7% of spam for DEPT and RELAY, respectively, with very low FPR. By enabling the self-maintained IP blacklist, we achieve a higher TPR of 96.5% and 99.9% for both datasets with no additionally incurred false positives. Our reported detection accuracy is comparable to today's supervised-learning-based spam filters such as SpamAssassin [5]. We further manually examined the spam belonging to the 3.5% false negatives of DEPT and confirmed the reason they escaped SCA is because SCA did not witness enough of similar spam to identify the campaigns they belong to. We found that supplementing SCA with commonly used DNSBL lookup can further increase the TPR to 97.1% for DEPT.

4. CONCLUSIONS AND FUTURE WORK

Motivated by key observations about today's Internet spam, we have demonstrated the feasibility of designing a completely unsupervised spam filter system whose detection accuracy is comparable to that of the *de-facto* supervised anti-spam solutions as SpamAssassin [5]. We also identified the visibility challenge as the main obstacle to the spam detection accuracy of such a filtering system when deployed at individual organizations. In our ongoing work, we are collecting diverse email datasets to further evaluate SCA. We are seeking solutions for sharing signatures among multiple organizations to effectively address the visibility challenge.

5. REFERENCES

[1] Thomas K Landauer, Peter W. Foltz, and Darrell Laham. An introduction to latent semantic analysis. *Discourse Processes 25:259-284*, 1998.

[2] Abhinav Pathak, Feng Qian, Y. Charlie Hu, Z. Morley Mao, and Supranamaya Ranjan. Botnet spam campaigns can be long lasting: Evidence, implications, and analysis. In *SIGMETRICS*, 2009.

[3] Andreas Pitsillidis, Kirill Levchenko, Christian Kreibich, Chris Kanich, Geoffrey M. Voelker, Vern Paxson, Nicholas Weaver, and Stefan Savage. Botnet judo: Fighting spam with itself. In *NDSS*, 2010.

[4] Feng Qian, Abhinav Pathak, Y. Charlie Hu, Z. Morley Mao, and Yinglian Xie. A case for unsupervised-learning-based spam filtering. Technical report, University of Michigan, February 2010. http://www.eecs.umich.edu/~fengqian/CSE-TR-561-10.pdf.

[5] S. Sinha, M. Bailey, and F. Jahanian. Shades of grey: On the effectiveness of reputation-based blacklists. In *MALWARE 2008*, 2008.