Poster: Effort-based Detection of Comment Spammers

Acar Tamersoy
College of Computing
Georgia Institute of Technology
tamersoy@gatech.edu

Hua Ouyang
Yahoo! Labs
houyang@yahoo-inc.com

Duen Horng Chau
College of Computing
Georgia Institute of Technology
polo@gatech.edu

***Abstract*—Social media has become ubiquitous and important for content sharing. A typical example of how users contribute content to a social media platform is through comment threads in online articles. Unfortunately, there is an increasing prevalence of malicious activity in these threads by spammers through comment messages. The existing approaches tackling comment spam are comment-level in that they attempt to classify a comment message as spam or not spam. We propose EDOCS, a graph-based user-level approach that quantifies how much *effort* a user exerted over his or her comments, to detect if the user is a comment spammer or not. We conjecture that spammers can only exert limited effort in terms of time and money over preparing and disseminating their comments, hence their effort scores are expected to be lower than those of the legitimate users. Our experimental evaluation of EDOCS shows its effectiveness in detecting comment spammers accurately with 95% true positive rate at 3% false positive rate as well as preemptively.**

I. Introduction

In recent years, social media has become ubiquitous and important for content sharing. An example of how users contribute content to a social media platform is through comment threads in online articles (e.g., news), which allow users to share their insights and engage in discussions with each other. An important aspect of the comment space is its open nature; in most social media platforms one can post a comment anonymously or with an account that can be obtained in a matter of seconds. Also, comments posted on a popular social media platform can easily reach a significant number of users.

Unfortunately, this open nature of the comment space provides malicious users with various opportunities to abuse it. For instance, abusers often use comment threads to post content irrelevant to the article. Such content is typically referred to as spam, posted by so-called comment spammers [1]. Comment spammers are posing a serious problem; a recent study showed that more that 75% of the one million blog comments collected were indeed spam [2]. Furthermore, some spam comment messages are extremely malicious; they contain text luring users to click links leading to malware sites [3].

However, detecting comment spam is challenging for the following reasons. Comment spam is different from other forms of spam in that a typical spam comment message is usually short and carefully crafted by humans; even human experts have hard times differentiating some spam comments from legitimate ones [3].[1] In contrast, the majority of spam email messages, for instance, are generated by botnets using certain predefined templates [4]—an important property leveraged by many approaches tackling email spam (see [5] for a survey). Relying solely on human experts to detect comment spam is also not feasible; human experts simply do not have the bandwidth to deal with the enormous amounts of content generated by users in today's social media era [3]. In addition, recent research showed that human experts are not very effective in detecting spam messages [6], [7].

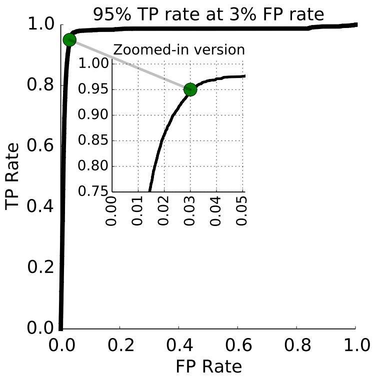

Fig. 1. Receiver operating characteristic (ROC) curve for the spammer detection experiment. EDOCS achieves 95% true positive (TP) rate in detecting spammers at 3% false positive (FP) rate, while labeling over 197k users.

The existing approaches proposed for comment spam take a comment-level view to the problem in that they attempt to classify a comment message as spam or not spam by mainly considering the characteristics of the comment and sender [1], [2], [3], [8]. We take a different slant on the problem and propose *E̲ffort-based D̲etection o̲f C̲omment S̲pammers* (EDOCS), a graph-based user-level approach that quantifies how much *effort* a user exerted over his or her comments, to detect if the user is a comment spammer or not. As we will explain below, we expect that the effort scores of the comment spammers are lower than those of the legitimate users.

II. Our Approach: EDOCS

Why quantifying effort can help detect spammers? We conjecture that spammers can only exert limited effort in terms of time and money over preparing and disseminating their comments. For instance, we expect that spammers recycle their spam comment messages and post the same message on different articles as each message is time-consuming to craft. We propose EDOCS to capture this intuition, by analyzing a bipartite graph of users and effort-related feature values to quantify how much effort a user exerted over his or her comments. EDOCS outputs an overall *effort score* for each user, taking into account all the comments that the user posted. Given their limited effort budget, intuitively we expect that the effort scores of the comment spammers are lower than those of the legitimate users.

[1] In our context, human experts are editors whose job responsibility include labeling users' comments as spam or not spam in a social media platform.

Number of users	197,464 (20.03% spammers)
Number of comments	1,201,277
Mean/median number of comments per user	6.08/1
Dataset duration	May 1–31, 2014
Duration of follow-up period	June 1–August 5, 2014

TABLE I. CHARACTERISTICS OF OUR COMMENTS DATASET.

The EDOCS algorithm. EDOCS operates on a bipartite graph of users and effort-related feature values. A user is connected to all the feature values that apply to her (e.g., an edge connecting the user with her IP address). EDOCS performs iterative message propagation on this graph. Specifically, messages are first propagated from users to feature values, where they are aggregated using feature-specific aggregation functions, and these aggregated messages are then propagated back to the users. The propagation ends when a maximum number of iterations is reached, after which an overall effort score is computed for each user using a general aggregation function.

Implementation details. In our current implementation of EDOCS, we perform the message propagation for two iterations given the scale of our dataset (see details below) and we utilize the two important features present in our dataset: the body of the comment and the IP address of the comment poster. Our intuition is that if a user posts the same comment body multiple times, possibly with other users, and shares the same IP address with other users, this might be an indication of a spamming activity or campaign. To capture this intuition, EDOCS executes with the following message values and aggregation functions.

Comment body effort: Each user node sends to the neighboring comment body nodes a message containing as its value the total number of times the user posted the corresponding message. Each comment body node computes the sum of all the incoming messages' values and sends the reciprocal of the sum to the neighboring user nodes.

IP effort: Each user node sends to the neighboring IP address nodes a message containing the value 1. Each IP address node computes the sum of all the incoming messages' values and sends the reciprocal of the sum to the neighboring user nodes.

Overall effort: Each user node computes the sum of all the messages' values arriving from the comment body nodes and normalizes the sum by the total number of comments the user posted. Similarly, the user node computes the sum of all the messages' values arriving from the IP address nodes. Finally, the user node returns the sum of these two values as the overall effort score for the corresponding user.

Dataset. We use a dataset containing user comments posted on the finance portal of a large internet company during May 2014. The characteristics of our dataset are shown in Table I. A user is assumed to be a spammer if he or she posted at least one comment labeled as spam by human experts.

Experiments (Part I: Detecting spammers). Figure 1 shows EDOCS's effectiveness in detecting spammers with a receiver operating characteristic (ROC) curve; EDOCS achieves an impressive 95% true positive (TP) rate at 3% false positive (FP) rate, assuming that spammers belong to the positive class. We generated the ROC curve as follows: (i) we ran EDOCS to obtain an effort score for each user; (ii) we considered each effort score in ascending order (recall that low effort scores are indicative of spammers) and used the effort score as a cutoff value for classification—a user who had an effort score smaller than the cutoff value was labeled as spammer, or clean otherwise; (iii) using the classifications of users generated from each cutoff value, we finally computed a pair of TP rate and FP rate values; plotting and connecting these pairs of values gave us the smooth ROC curve in Figure 1.

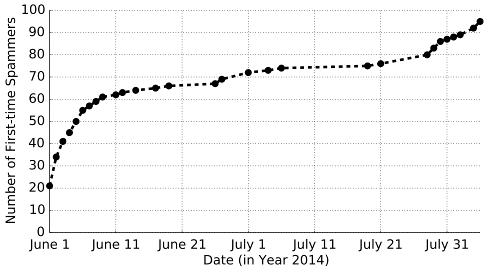

Fig. 2. Conversion trend of users from "clean" to spammer based on the date of their first spam comment messages during the follow-up period (June 1–August 5, 2014). EDOCS preemptively detected these 95 users (top right corner) as spammers using data from May 2014.

Experiments (Part II: Follow-up on false alarms). We next focus on the users belonging to the FP set that we obtained from the cutoff value used in the 95% TP rate at 3% FP rate result in Part I above. Note that these are the users that EDOCS labeled as spammers, however they did not have any spam message within the duration of our dataset. To examine if these users were indeed "clean", we followed them for two more months (June 1–August 5, 2014) and we checked if they posted any spam comments. Out of 937 users who had a comment during this follow-up period, 95 of them posted at least one spam comment message, resulting in a 10.1% clean-to-spammer conversion rate. Figure 2 shows the conversion trend based on the date of the first spam comment messages. Note that conversions occur consistently, showing the effectiveness of EDOCS in detecting spammers preemptively (i.e., it can detect spammers early on).

III. Conclusion

We tackle the crucial problem of comment spam and propose EDOCS, a graph-based approach that quantifies how much *effort* a user exerted over his or her comments, to detect if the user is a comment spammer or not. Our experimental evaluation of EDOCS shows its effectiveness in detecting comment spammers accurately with 95% true positive rate at 3% false positive rate as well as preemptively. As future work, we plan to incorporate additional features to EDOCS.

References

[1] G. Mishne, D. Carmel, and R. Lempel, "Blocking blog spam with language model disagreement," in *AIRWeb*, 2005.

[2] S. Abu-Nimeh and T. M. Chen, "Proliferation and detection of blog spam," *IEEE Security & Privacy*, vol. 8, no. 5, pp. 42–47, 2010.

[3] A. Kantchelian, J. Ma, L. Huang, S. Afroz, A. Joseph, and J. D. Tygar, "Robust detection of comment spam using entropy rate," in *AISec*, 2012.

[4] A. Ramachandran and N. Feamster, "Understanding the network-level behavior of spammers," in *SIGCOMM*, 2006.

[5] E. Blanzieri and A. Bryl, "A survey of learning-based techniques of email spam filtering," *Artificial Intelligence Review*, vol. 29, no. 1, pp. 63–92, 2008.

[6] M. Ott, Y. Choi, C. Cardie, and J. T. Hancock, "Finding deceptive opinion spam by any stretch of the imagination," in *ACL*, 2011.

[7] M. Ott, C. Cardie, and J. T. Hancock, "Negative deceptive opinion spam," in *HLT-NAACL*, 2013.

[8] D. Sculley and G. M. Wachman, "Relaxed online svms for spam filtering," in *SIGIR*, 2007.