# Combating Threats to Collective Attention in Social Media: An Evaluation

**Kyumin Lee, Krishna Y. Kamath, James Caverlee**
Texas A&M University
College Station, TX 77843
{kyumin, kykamath, caverlee}@cse.tamu.edu

## Abstract

Breaking news, viral videos, and popular memes are all examples of the collective attention of huge numbers of users focusing in large-scale social systems. But this self-organization, leading to user attention quickly coalescing and then collectively focusing around a phenomenon, opens these systems to new threats like collective attention spam. Compared to many traditional spam threats, collective attention spam relies on the insidious property that users themselves will intentionally seek out the content where the spam will be encountered, potentially magnifying its effectiveness. Our goal in this paper is to initiate a study of this phenomenon. How susceptible are social systems to collective attention threats? What strategies by malicious users are most effective? Can a system automatically inoculate itself from emerging threats? Towards beginning our study of these questions, we take a two fold approach. First, we develop *data-driven models* to simulate large-scale social systems based on parameters derived from a real system. In this way, we can vary parameters – like the fraction of malicious users in the system, their strategies, and the countermeasures available to system operators – to explore the resilience of these systems to threats to collective attention. Second, we pair the data-driven model with a *comprehensive evaluation over a Twitter system trace*, in which we evaluate the effectiveness of countermeasures deployed based on the first moments of a bursting phenomenon in a real system. Our experimental study shows the promise of these countermeasures to identifying threats to collective attention early in the lifecycle, providing a shield for unsuspecting social media users.

## Introduction

Collective attention – exemplified by breaking news, viral videos, and popular memes that captivate the attention of huge numbers of users – is one of the cornerstones of large-scale social systems. As Wu and Huberman have noted, *collective attention* describes how "attention to novel items propagates and eventually fades among large populations" (Wu and Huberman 2007). In the context of social media, an item – be it a video, web page, image – attracts the interest of a small group, then gathers a larger following as additional attention focuses on it, then (in some cases) exploding across social media to large-scale attention, and then finally fading in interest. Popular examples include YouTube videos that accumulate millions of views in a few days, memes attracting huge audiences on Reddit (http://www.reddit.com) and 4chan (http://www.4chan.org), spikes in search volume on Google and Twitter following breaking news, and so forth. As a result, many researchers have begun examining these phenomena, to model their dynamics, lifecycles, and future spread, e.g., (Goetz et al. 2009; Hong, Dan, and Davison 2011; Lehmann et al. 2012; Lerman and Ghosh 2010; Romero, Meeder, and Kleinberg 2011).

Guided by the knowledge that collective user interest may quickly coalesce, malicious users have begun threatening the quality of information associated with this collective attention. As illustration, consider these two examples of *collective attention spam* found in large-scale social systems:

- **YouTube:** In the immediate aftermath of the London Olympics Opening Ceremony on July 27, 2012, we found that four of the top-five videos returned for the YouTube query "london olympics opening ceremony 2012" were videos tagged with keywords associated with the London Olympics Opening Ceremony, but that were expressly designed to promote an unrelated spammer-controlled website. Figure 1 shows one example, which includes a URL linking to a spam website.
- **Twitter:** Twitter publishes the current most-trending topics, and so spammers have been observed abusing this signal of collective user interest by "trend-stuffing" these popular topics with spam messages including malicious URLs (Irani et al. 2010). Figure 2 shows a sample search result for the trending topic "Glen Rice" for which three out of the most recently posted six messages are spam. All three spam messages include the same URL and multiple trending topics, but are posted from multiple accounts, adding to the growing evidence (e.g., (Ratkiewicz et al. 2011; Thomas et al. 2011)) that spammers strategically post to Twitter in an organic-like way to simulate the behavior of non-spam users.

In contrast to traditional email spam and social spam, *collective attention spam relies on users themselves to seek out the content where the spam will be encountered*. In email spam, as illustrated in Figure 3(a), the spammer relies on a bulk attack based on the hope that a small percentage of users who are contacted will actually click on a link in an


Figure 1: Example YouTube video designed to capitalize on collective interest during and immediately after the London Olympics Opening Ceremony.

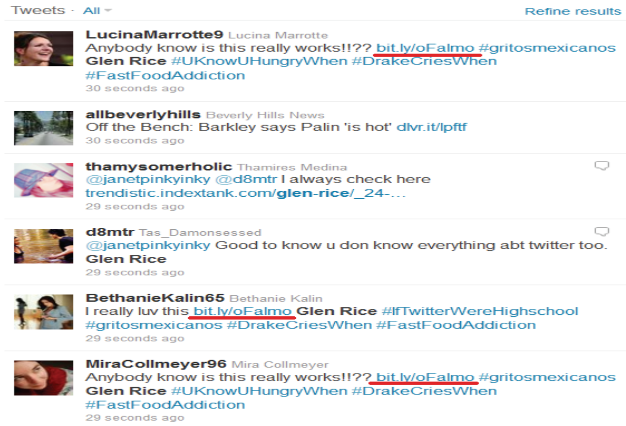



Figure 2: Spam messages targeting the Twitter trending topic “Glen Rice”.

email. Social spam, as illustrated in Figure 3(b), is typically a more targeted attack than email spam, and relies on some social mechanism for coupling a spammer with an intended target (e.g., becoming friends in a social network, following a user on Twitter). In contrast, collective attention spam in Figure 3(c) targets users *who are already inherently interested in the topic*. In this way, users themselves have self-selected for interest in the topic and made themselves susceptible to collective attention spam.

While email and social spam have been the subject of considerable study, there is a significant gap in our understanding of the susceptibility of social systems to collective attention threats. Our goal in this paper is to begin to understand this phenomenon better, building on our preliminary effort to detect collective attention spam reported in (Lee et al. 2012). How susceptible are social systems to malicious attacks? What strategies by malicious users are most effective? And least effective? How do users of a system access items of interest and how does this affect their exposure to threats? Can a system automatically inoculate itself from emerging attacks? What kinds of countermeasures can be deployed and how effective are they at limiting the effectiveness of malicious users?

*Our approach.*

Answering these questions is challenging. Large-scale social systems are typically proprietary and responsible to their current user base, so it is infeasible to automatically “stress-test” such a system by subjecting it to hundreds or thousands of malicious users. An alternative is to take a representative snapshot of a system and measure the current level of threats in the system and characterize their reach and effectiveness. However, this approach alone may not be suitable for understanding the system’s future state, as social systems are constantly evolving. Hence, we take a two fold approach. First, we take a *data-driven modeling* approach, in which we simulate a large-scale social system based on parameters derived from a real system. In this way, we can vary system parameters – like the fraction of malicious users in the system, their strategies, and the countermeasures available to system operators – to explore the resilience of these systems to threats to collective attention. We pair the data-driven model with a *comprehensive evaluation over a Twitter system trace*, in which we evaluate the effectiveness of countermeasures deployed based on the first moments of a bursting phenomenon in a real system.

*Summary of key findings.*

In summary, this paper presents the first comprehensive study of collective attention spam in social systems.

- Through our data-driven model, we find that social systems are extremely susceptible to collective attention spam. With spammers accounting for only 5% of all users, we find that every legitimate user can be exposed to spam. At even higher spammer penetration, the social system becomes unusable with spam dominating.
- We find that strategically organized spammers can collude to selectively push spam payloads, increasing the exposure of legitimate users to spam content.
- On a positive note, we find that the countermeasures deployed early in the lifecycle of a collective attention attack can dramatically reduce the amount of spam in the system. Through testing over 20 million Twitter messages, we validate the model findings and see that these countermeasures can effectively identify threats to collective attention early in the lifecycle with 98% accuracy, reducing “spamness” up to 73% and providing a shield for unsuspecting social media users.

## A Data-Driven Model for Studying Collective Attention Threats

In this section, we present a data-driven modeling approach for simulating collective attention and threats. Our goal is to answer questions about the susceptibility of social systems to collective attention threats and to explore techniques for limiting this impact. We begin by describing how both good and bad users post content to the social system, and how the system itself supports information access. We describe how the model is seeded and validated, and then we investigate (i) threats from individual spammers; (ii) threats from coordinated spammers; and (iii) finally, we examine countermeasures. In the following section, we revisit these

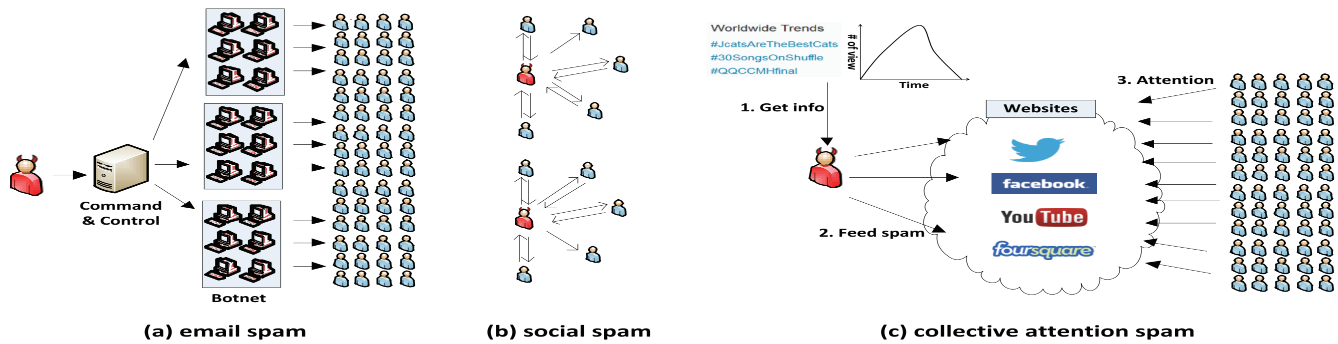



Figure 3: Three spam approaches. Collective attention spam relies on the users themselves to seek out the content where the spam will be encountered.

model-driven results through an experimental study over a real Twitter trace.

## System Model

We consider a social system of interest $\mathcal{S}$, consisting of a set of content items $\mathcal{C}$ (e.g., videos, tweets, etc.), a set of topics $\mathcal{T}$ for which each content item is associated (e.g., the "London Olympics" topic, the "Steve Jobs" topic, etc.), and a set of users $\mathcal{U}$, who participate in the system by posting and viewing content items. For example, a user in $\mathcal{U}$ may post a tweet "Thank you #SteveJobs The world will miss you", where the tweet is associated with the topic indicated by the hashtag #SteveJobs. Similarly, a user may post a video to YouTube associated with the "London Olympics" topic by including a tag or descriptive text at the time of upload. We use the symbols $u$, $c$, and $t$ to denote a user in $\mathcal{U}$, a content item in $\mathcal{C}$, and a topic in $\mathcal{T}$.

## Posting Model

To populate a social system, we initialize the system with a set of topics and a set of users. To model users in a social system, we define two sets of users: a good user set $U^+$ and a bad user set $U^-$. Good users post content items that are associated with a "correct" topic. Bad users, on the other hand, post content items that are irrelevant to the topic they are associated with. For example, a bad user may post a spam video, but intentionally describe it as a "London Olympics" video. When users post to the system, we assume they have access to both the set of topics $\mathcal{T}$ as well as the current subset of "popular" topics $\mathcal{T}_{pop}$ (in practice, these popular topics may be known to users via prior knowledge or explicitly advertised by the system, as in the case of Twitter trending topics or popular YouTube videos). The system proceeds in step-wise fashion; at each time increment, users generate content items according to a particular posting model. Good users post according to the *good user model*:

**Good User Model:**
for each user $u \in U^+$ do
  with probability $\gamma^+$ decide to post:
    with probability $\delta^+$:
      select a popular topic $t \in \mathcal{T}_{pop}$ and relevant item $c$;
    else:
      select at random a topic $t \in \mathcal{T}$ and relevant item $c$.

At each time increment, a good user chooses to post something with the user content generation probability $\gamma^+$. If a user decides to post a content item, an already popular topic is selected with probability $\delta^+$; alternatively, the user decides to post to a random topic. A bad user follows a similar process, but always posts spam content items:

**Bad User Model:**
for each user $u \in U^-$ do
  with probability $\gamma^-$ decide to post:
    with probability $\delta^-$:
      a popular topic $t \in \mathcal{T}_{pop}$ and spam item $c$;
    else:
      select at random a topic $t \in \mathcal{T}$ and spam item $c$.

Notice that the user content generation probability $\gamma$ and the popular topic probability $\delta$ may vary between the good and bad user models. As part of our data-driven simulation, we will vary these parameters to reflect different spammer behaviors. For example, a spammer may adopt a high rate of content generation relative to good users (e.g., $\gamma^- \gg \gamma^+$) in an attempt to flood the system with spam content. Alternatively, a spammer seeking to maximize their potential audience may choose to focus only on popular topics and so adopt a popular topic probability much greater than the good user model (e.g., $\delta^- \gg \delta^+$).

## Collective Attention Access Models

Given the approach for populating a social system, we now consider how users access the content posted in the system. We assume that users monitor topics by one of two methods:

- *By recency*: In the first access model, users interested in a topic access the $k$-most recently posted items related to the topic. This recency approach is akin to the "Most Recent Uploads" functionality on YouTube, viewing comments associated with a blog by their posting order (from recent to oldest), and Twitter's basic search.
- *By relevance*: The second access model imposes a relevance ordering over content items associated with a topic. This relevance-based approach may incorporate the popularity of an item (e.g., rank images in order of the number of clicks they have accumulated), content and link-based ranking (e.g, applying IR principles), or learning-to-rank methods. For modeling purposes, we

Table 1: A sample of 101 popular topics. In total, there are ~13m messages, of which 3.7% are spam.

| No. | Topic | Period | Total Lifespan | # of messages |
|---|---|---|---|---|
| 1 | #SomeWhereInTheHood | 2011-09-26 04:13:22 ∼ 2011-09-26 17:01:22 | 12 hrs 48 mins | 93,871 (2.5% spam) |
| 2 | #ThatOneEx | 2011-09-26 13:00:12 ∼ 2011-09-26 23:21:15 | 10 hrs 21 mins | 58,217 (3.0% spam) |
| 3 | #thewayiseeit | 2011-09-27 07:17:59 ∼ 2011-09-28 01:23:39 | 18 hrs 06 mins | 201,682 (4.6% spam) |
| 4 | #LawsMenShouldFollow | 2011-09-27 08:54:23 ∼ 2011-09-28 01:24:03 | 16 hrs 30 mins | 181,524 (4.0% spam) |
| ... | ... | ... | ... | ... |
| 98 | #DoctorsBetterThanConradMurray | 2011-11-07 16:17:07 ∼ 2011-11-08 04:18:30 | 12 hrs 02 mins | 68,370 (12.2% spam) |
| 99 | #WhaILove | 2011-11-08 03:49:57 ∼ 2011-11-09 03:51:00 | 24 hrs 02 mins | 174,695 (3.1% spam) |
| 100 | #hometownslogans | 2011-11-08 05:07:32 ∼ 2011-11-09 03:09:14 | 22 hrs 02 mins | 59,529 (5.5% spam) |
| 101 | #ThingsThatYouShouldKnow | 2011-11-08 21:45:02 ∼ 2011-11-09 10:53:53 | 13 hrs 09 mins | 95,542 (3.6% spam) |

assume that content items are ranked by their occurrence count, with all duplicates removed to maintain diversity (i.e., item $c_i$ posted 20 times is ranked first; item $c_j$ posted 10 times is ranked second; and so on).

User interest in a topic is based on the amount of content items posted to the topic. So, if topic $t_i$ is the most popular topic according to the good and bad user models, then it will be monitored by the most users. In this way, as items become more bursty, collective attention in them rises accordingly.

## Measuring Spam Impact

To evaluate the impact of bad users on inserting spam into the system, we measure the overall *spamness*, which is a measure similar to NDCG@k (Järvelin and Kekäläinen 2000). Note that NDCG@k is a metric to measure the quality of top $k$ search result. For a user accessing topic $t$, we have:

$$Spamness(t,k) = \frac{\sum_{i=1}^{k} w(c_i) * \frac{1}{log_2(1+i)}}{Norm(k)}$$

where

$$w(c_i) = \begin{cases} 1, & \text{if } c_i \text{ is a spam content item;} \\ 0, & \text{otherwise.} \end{cases}$$

and $k$ is the number of items (e.g., messages or tweets) shown in a search result by a search system, and $Norm(k) = \sum_{i=1}^{k} \frac{1}{log_2(1+i)}$ is a normalizing constant. Spamness varies from 0 to 1, with 0 signifying no impact to 1 signifying all of the items viewed by a user are spam. If users view 10 items at a time ($k$ = 10), with three spam items, spamness ranges between 0.200 and 0.469, depending on where the spam items are located in the search result; if they are positioned in the top, spamness will be high. As a rule-of-thumb we consider a *spamness of 0.2 or greater to indicate a high-level of spam*, corresponding to a user encountering 3 or more spam items for every 10 items encountered.

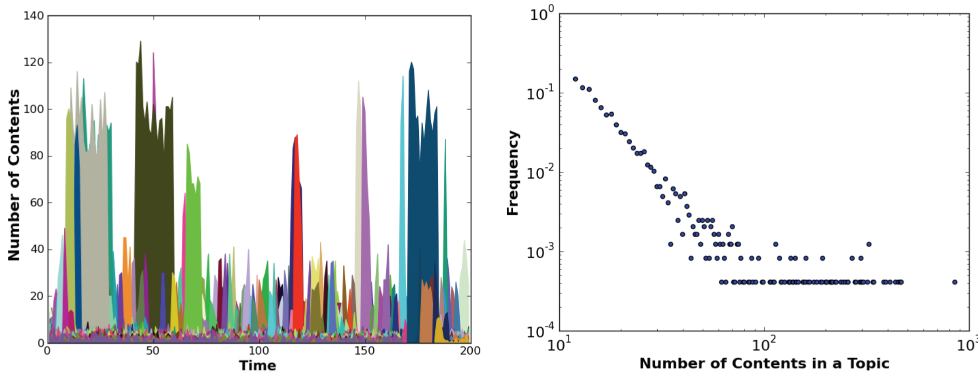
## Seeding and Validating the Model

To accurately model real social systems for a *data-driven simulation*, we require baseline parameter settings. However, there are no standard datasets of collective attention spam. Hence, we sampled a collection of popular topics and their associated messages from Twitter between September and November 2011. We polled Twitter's *trending topics* every 5 minutes and collected the messages associated with each trending topic. In total, we collected 19,275,961 messages posted by 3,989,563 users across 354 trending topics.

But how many of these messages are actually spam? It is important to find a baseline estimate so that the model parameters can be seeded realistically. To assess the amount of spam in the dataset, we systematically checked whether a user associated with a message had been suspended by Twitter for engaging in spam behaviors. If an account is suspended, Twitter will redirect the request to a standard "suspension" page: http://twitter.com/account/suspended. Not all suspended accounts may have actually engaged in spam, so we further assessed these accounts. Concretely, we randomly sampled 200 messages each from the messages posted by suspended accounts and from those posted by non-suspended accounts. Two human judges manually labeled the 400 messages as either spam or non-spam. From the non-suspended accounts, 199 out of 200 messages sampled were labeled as non-spam messages. From the suspended accounts, 187 out of 200 messages sampled were labeled as spam messages. Based on this high accuracy, we make the simplifying assumption that all messages posted by suspended users are indeed spam so that all ~19 million messages can be automatically labeled.

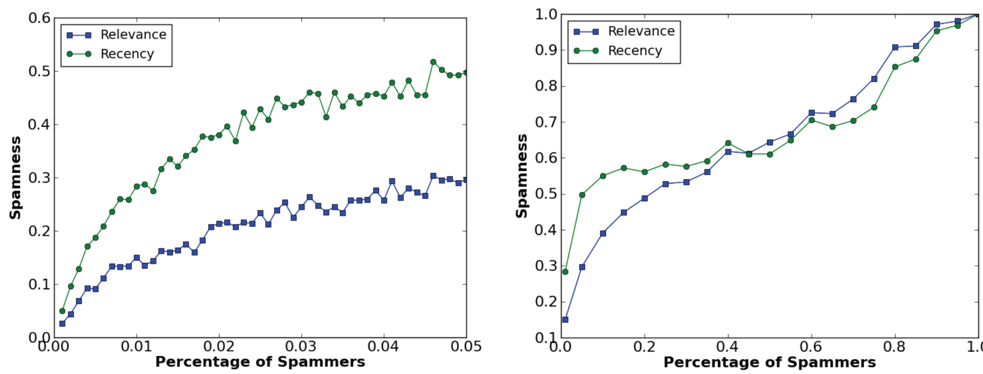
A sample from the top-101 topics with the most messages is shown in Table 1. Together, these topics account for 12,954,965 messages. A topic has on average 132,725 messages and 3.7% of them are generated by spammers, who account for around 1.5% of all accounts in the dataset.

Following the observed spam amount in the real data, we set the fraction of spammers in the system as 1.5%. We then varied the content generation probability ($\gamma$), and probability of picking popular topics ($\delta$) to find an initial model setting that emulated the real data distribution. Arriving at initial settings of $\gamma^+ = 0.1$, $\gamma^- = 1.0$, $\delta^+ = 0.4$, and $\delta^- = 0.75$, we arrive at a topic distribution shown in Figure 4(a) following the heavy-tailed distribution as shown in Figure 4(b), which is similar to the expected distribution of bursty social media. Note that these initial settings fit our intuition, with bad users posting more often than good users and posting exclusively to popular topics. We find that small changes to these parameters make little qualitative difference to the conclusions drawn in the following. Based on these initial parameter settings, we next explore the following research questions: how susceptible are social systems to malicious




(a) Topic distribution (b) Log-log graph

Figure 4: The left figure depicts a topic distribution generated by the model. Each color denotes a topic. The right figure depicts a log-log graph showing the frequency of number of content items associated with each topic in the simulation data. The heavy-tailed distribution is similar to bursty social media.




(a) From 0 to 5% (i.e., 0 $\sim$ 0.05 in the $x$-axis) (b) From 0 to 100% (i.e., 0 $\sim$ 1 in the $x$-axis)

Figure 5: Evaluating the impact of increasing the fraction of spammers in the system.

attacks? what strategies by malicious users are most effective (e.g., individual attack, group-based coordinated attack, or combination of individual and coordinated attacks)? What kinds of countermeasures can be deployed and how effective are they at limiting the effectiveness of malicious users?

## Threats from Individual Spammers

We've seen in one example system (Twitter) that about 1.5% of users are collective attention spammers. Suppose this fraction of spammers increases. What impact will this have on the amount of spam that legitimate users are exposed to? For this first experiment, we vary the fraction of spammers from 0 to 100%, (we keep the same $\gamma$ and $\delta$, but increase the fraction of spammers). We see in Figure 5(a) that naturally, the spamness of the system increases with an increasing number of spammers. Interestingly, the recency-based access approach fairs significantly worse than the relevance-based one, crossing the spamness threshold of 0.2 when less than 1% of all users are spammers. The relevance-based approach is less susceptible to spam since individual spammers cannot selectively *push* particular items; in contrast so long as users access the most-recent items, spammers can easily insert spam items that will be viewed. Although the relevance-based approach is more resistant to spammers, if the fraction of spammers were to increase only slightly to 2%, then the spamness threshold would be passed. As the fraction of spammers increases beyond 5%, we see in Figure 5(b) that neither access approach can significantly limit the amount of spam in the system, with both approaches near or above a spamness of 0.5 with just 20% spammers. At even higher ranges, presumably the social system would become unusable and unappealing to legitimate users, with spam dominating.

## Threats from Coordinated Spammers

The threat so far has considered individual spammers who do not coordinate their actions; that is, there is no common spam payload shared across multiple spammers for perhaps increasing its reach. Hence, in this next experiment we consider a coordinated spam approach in which spammers are assigned to a group which is associated with a common pool of spam payloads. For the following experiment, we assume that spammers share a common pool of spam payloads, and we vary the number of spam payloads.

Using this coordinated approach, we observe in Figure 6 that the recency-based approach is largely unaf-

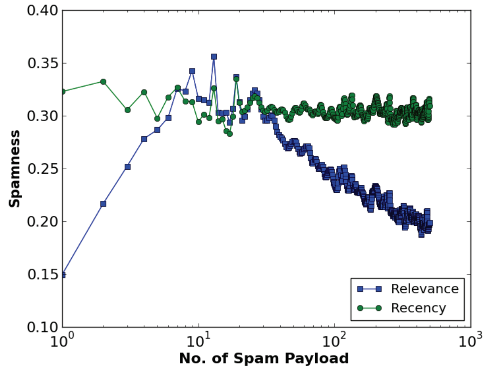



Figure 6: Coordinated Spam: By focusing their efforts, groups can achieve even higher impact.

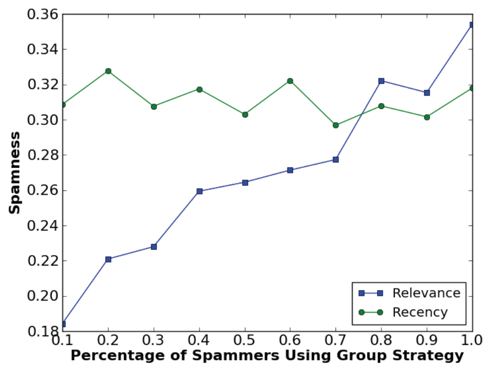



Figure 7: With as few as 20% adopting the group strategy, spamness passes the 0.20 threshold.

fected, but that it remains highly susceptible to spam. The relevance-based approach shows that spammers have a potential "sweet spot" for targeting spam. At a low number of payloads, the spamness is relatively low since the spammers promote a few payloads which possibly pollute one or two out of the top-$k$ results. As the number of payloads increases, the coordinating spam group can achieve an impact equal to or even better than under the recency-based approach. However, as the number of payloads continues to increase, the effectiveness for the coordinating spam group falls, because the power promoting payloads is distributed across too many payloads, meaning no single one can penetrate the top-$k$, and hence be exposed to end users interested in the topic.

What if spammers adopt a mixed strategy, balancing between the individual and the coordinated approach? Figure 7 compares the robustness of the two access approaches to a mixed spam strategy. We observe the continued poor resistance of the recency-based approach. To effectively target the relevance-based approach, spammers need only adopt very little collusion (i.e., with 20% adopting the group strategy, spamness passes the 0.20 threshold). At even higher-levels of collusion ($\geq$80%), spammers are even more effective than under the recency-based approach, further confirming the dangers of strategically organized spammers.

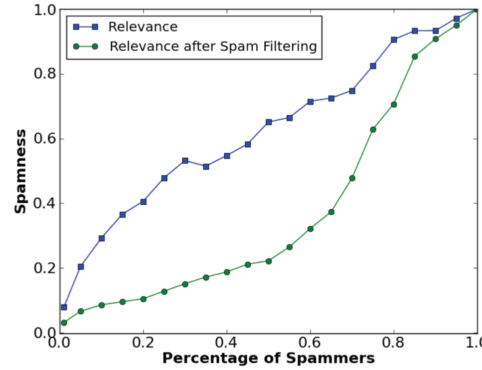



Figure 8: Applying a simple rule-based countermeasure greatly reduces spamness, but is not effective against strategic behavior.

## Countermeasures

So far we have seen that the relevance-based access approach is generally more resistant than recency to collective attention spam, but that both are extremely susceptible to only slight changes in the fraction of spammers and to strategic efforts to coordinate spam behavior. We now consider the impact of countermeasures to collective attention spam to better understand under what scenarios spam may be detected and filtered. We consider two countermeasures:

**Countermeasure 1: Rule-Based Filtering.** The first is a rule-based filtering approach, which is potentially easy to deploy in a real-system, but that may not be adaptable to changes in behavior by malicious users. We consider a simple rule that considers the ratio of users to content items:

$$PayloadScore(t,p) = 1 - \frac{\#\ of\ distinct\ users}{\#\ of\ content\ items}$$

where $t$ and $p$ denote a topic and a payload, respectively. The rule-based filtering approach counts # of content items containing a payload $p$ in the topic $t$ and # of distinct users who generate the content items, and then filters out content items exceeding a threshold. The intuition is that collective attention spammers may strategically use common payloads, so if fewer users post more of the same item (e.g., a common URL or spam image) they can be filtered out.

Setting a threshold of 0.1 and applying this countermeasure to the recency-based approach makes little difference since the spamness is already so high (as we saw in previous experiments). However, applying this countermeasure to the relevance-based approach results in a dramatic reduction in spamness as shown in Figure 8. While encouraging, it is not obvious that such improvements could be observed in practice, with spammers strategically changing their behavior. We'll revisit the effectiveness of such a rule-based countermeasure in the following section.

**Countermeasure 2: Supervised Classification.** A second countermeasure is a spam detector relying on supervised classification principles. The intuition is that system operators may be able to sample evidence of spam early in the

Table 2: Evaluating the potential effectiveness of a low-accuracy (40%) and a high-accuracy (90%) collective spam detector.

| Access Approach | Avg | Min | Max |
|---|---|---|---|
| Recency | 0.228 | 0.198 | 0.279 |
| + low-accuracy detection (40%) | 0.120 | 0.102 | 0.156 |
| + high-accuracy detection (90%) | 0.041 | 0.030 | 0.052 |
| Relevance | 0.176 | 0.148 | 0.215 |
| + low-accuracy detection (40%) | 0.115 | 0.099 | 0.138 |
| + high-accuracy detection (90%) | 0.036 | 0.027 | 0.044 |

lifecycle of a collective attention phenomenon (e.g., sampling and labeling spam tweets from a trending topic). Based on this early evidence, perhaps an effective classifier can be quickly deployed for filtering out subsequent spam. To evaluate such an approach, we consider two detectors: a low-accuracy spam detector that can only filter out 40% of all spam items as they enter the system, and a high-accuracy spam detector that can filter out 90% of all spam items. As an example, a low-accuracy detector may be built on imperfect crowdsourced spam labeling, while a high-accuracy detector may have been refined over large carefully curated spam datasets.

We show in Table 2, the hypothetical performance of two detectors versus the baseline (no countermeasure) case over a 90 minute "run" of the system model. At each one-minute time unit, users post content, the detectors are applied, and the spamness of the results from the access approaches are calculated. We see over the 90 minutes that even the low-accuracy spam detector achieves good results, pushing the spamness well below the 0.2 threshold. The high-accuracy performs very well, with spamness below 0.06 in all cases. When increasing the fraction of spammers in the system, we find similarly robust results suggesting that effective countermeasures are a necessity for countering threats to collective attention in social media.

## Countermeasure Deployment on Twitter

Based on the data-driven model, we have identified the need for collective attention spam countermeasures. Though effective in simulation, it is unclear of such countermeasures are achievable in real social systems. Since many instances of collective attention are bursty and unexpected, it is difficult to build spam detectors to pre-screen them before they arise. Hence, in this section we study the viability of quickly deploying collective spam countermeasures based on the first moments of a bursting phenomenon. We examine the Twitter trace described in the previous section, consisting of 101 topics associated with 13 million messages. We investigate when a countermeasure may be optimally deployed to a trending topic. Early deployment of a supervised classifier has the potential to greatly reduce spam subsequently associated with the topic, but at a risk of learning only a limited model and resulting in less robust classification (resulting in higher false positives and false negatives). Late deployment has less potential to reduce the total amount of spam (since presumably most of it will have already arrived by the time of deployment), but will be more robust in its detection.

### Metrics

To evaluate the quality of a countermeasure, we augment the spamness measure with several standard spam metrics: accuracy, false positive rate (FP) and false negative rate (FN). Additionally, we measure the *total spam detected (TSD)* over a topic's lifespan:

$$TSD_{topic}(\%) = \frac{\textit{\# of detected spam}}{\textit{total \# of spam in the topic}}$$

The goal of a countermeasure is to reduce the most amount of spam, so *total spam detected* complements the traditional measures of accuracy, false positive rate, and false negative rate. For example, a countermeasure that is deployed late in the lifecycle of a topic may be very robust, with high accuracy and low false positives and false negatives, but may only detect a small fraction of all spam. Why? Because most of the spam occurred *before* the countermeasure was ever deployed. An effective countermeasure should balance accuracy and the other measures with the total spam detected, so that unsuspecting users are shielded from spam.

### Countermeasure 1: Rule-Based Filtering

We begin by considering a static rule-based filtering approach, based on the principles described in the previous section. In our observations of Twitter trending topics, we see that many spam messages contain a common advertisement or URL payload. In contrast, messages posted by legitimate users are more varied. For example, for the topic #DearHair, we noticed similar messages of the form:

*@9rappermode9 OMG, #DearHair RT Have you seen this??WTF how could it happen with hair?? : http://t.co/xcPx6JFe*

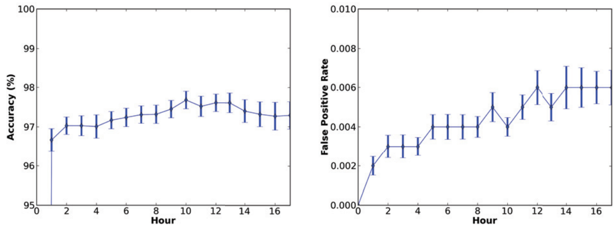
*@_enoughsaid_ OMG, #DearHair RT Have you seen this??WTF how could it happen with hair?? : http://t.co/fVD4UAbC*

where both URLs redirect to the same spam destination.[1]

We can define the *payload* as the message content after eliminating all hashtags, usernames, and URLs. In the example, the payload is *OMG, RT Have you seen this? ?WTF how could it happen with hair??*. With this payload definition and the simple payload score rule as presented in the previous section: $PayloadScore(t,p) = 1 - \frac{\#\ of\ distinct\ users}{\#\ of\ contents}$, we evaluate how many spam messages can be detected from the Twitter trace. In the best case, with a threshold of 0.1, we find that only 20% of all spam messages across all 13 million messages can be filtered (i.e., the average TSD is 20%).

While the space of all potential rules is large, we can see that a rule-based approach is likely to be insufficient by itself to reduce collective attention spam. Hence, we next explore in greater detail the supervised classification approach,

[1] To further illustrate the potential impact of collective attention spam, we accessed the bitly records for these URLs. URLs in 102 messages redirect to the same destination via various bitly URLs. One of the bitly URLs had been clicked a total of 1,424 times, indicating the effectiveness of targeting collective attention (available at http://bitly.com/usaend+).




(a) Accuracy (b) False positive rate

Figure 9: Evaluating Countermeasure 2: Supervised Classification. Average accuracy and false positive rate reported across 101 topics.

which promises potentially more adaptability to ongoing collective spam prevention.

## Countermeasure 2: Supervised Classification

We next investigate the viability of a supervised classifier for detecting collective attention spam that targets popular topics. Our goal is to predict whether a message $m$ posted to a trending topic (i.e., by including the associated hashtag or keyword) is a spam message through a classifier $c$:

$$c : m \rightarrow \{spam, non - spam\}$$

Our classification approach is that given a set of messages associated with a popular topic, we create a training set containing messages generated before a deployment time $x$ since the topic has become popular, and the rest of the messages associated with the topic belong to a testing set. We create multiple pairs of training and testing sets for different hourly deployment times. For example, for a trending topic with a 10-hour lifespan, we consider deploying the countermeasure at hour 1, at hour 2, and so on up to hour 9. In this way, we independently create 9 training sets, each of which contains messages posted during the first 1 hour, 2 hours, and so on up to 9 hours, respectively. Corresponding to the training sets, we create 9 testing sets containing the rest of messages.

Since collective attention spam targets topics as they become popular, detecting these spam messages as soon as possible is very important. Our goal is to explore the tradeoff between early deployment and late deployment. Under what circumstances does a supervised classifier filter collective attention spam? For the classifier, we adopt a decision tree based Random Forest classifier as a supervised learning method following previous success reported in (Lee, Eoff, and Caverlee 2011).

**Feature Selection.** Before building a classifier, finding good features is very important for high accuracy. We build classifiers based on 10 features extracted from each message: (1) # of URLs; (2) # of hashtags; (3) # of @mentions; (4) is a message retweeted?; (5) does a message contain a question mark?; (6) does a message contain an exclamation mark?; (7) the length of a message; (8) the number of words in a message; (9) the length of a payload (again, given a message, we first remove @mention, URLs and hashtags and call the remaining text a payload); and (10) the number of words in a payload.

Table 3: Top 10 features.

| Feature | $\chi^2$ Value | Avg Spam | Avg Good |
|---|---|---|---|
| # of URLs | 56,795 | 0.67 | 0.01 |
| length of message | 13,700 | 85.27 | 75.7 |
| length of payload | 11,398 | 46.86 | 48.43 |
| # of words in payload | 6,497 | 9.13 | 10.31 |
| # of words in message | 6,407 | 10.63 | 11.03 |
| # of hashtags | 3,343 | 1.25 | 1.1 |
| # of @mentions | 3,162 | 0.1 | 0.54 |
| is retweet | 2,115 | 0.06 | 0.38 |
| has exclamation mark | 1,797 | 0.23 | 0.14 |
| has question mark | 843 | 0.08 | 0.04 |

Table 4: On average, the supervised classifier countermeasure achieves 98% accuracy, detecting 50% of all spam messages.

| Topic | Tr. Time | Acc | FP | FN | TSD |
|---|---|---|---|---|---|
| #SomeWhere... | 2 hrs | 99.09 | 0.003 | 0.247 | 72.43 |
| #ThatOneEx | 8 hrs | 98.85 | 0.002 | 0.152 | 62.11 |
| #thewayiseeit | 5 hrs | 99.29 | 0.003 | 0.113 | 47.67 |
| #LawsMenSh... | 4 hrs | 99.51 | 0.003 | 0.083 | 49.01 |
| ... | ... | ... | ... | ... | ... |
| #DoctorsBe... | 3 hrs | 97.75 | 0.006 | 0.096 | 76.82 |
| #WhatILove | 3 hrs | 98.17 | 0.005 | 0.496 | 36.33 |
| #hometowns... | 5 hrs | 96.97 | 0.013 | 0.344 | 46.61 |
| #ThingsTha... | 8 hrs | 98 | 0.006 | 0.312 | 33.09 |
| **Average** | **5 hrs** | **97.57** | **0.007** | **0.384** | **50.16** |

In order to measure whether each feature has power to distinguish between spam and non-spam messages, we compute its $\chi^2$ value. If a feature has a positive $\chi^2$ value, it will have distinguishing power. Table 3 presents the average $\chi^2$ values of the 10 features across 101 topics. We observed that all features have power to distinguish between spam and non-spam messages. For example, we see that the number of URLs per message is 0.67 for spam, but only 0.01 for non-spam messages.

**Detection Across 101 Topics.** Next, we build a collective attention spam classifier over each of the 101 popular topics and evaluate them. For each topic, we build a classifier every hour since the topic has become popular. In total, we built 2,020 classifiers for 101 topics (i.e., 2,020 classifiers = 101 topics * 20 classifiers). The first question is whether spam messages detected in the early stages may accurately identify spam that follows as a topic becomes popular. Hence, in Figure 9(a) we report the average classification accuracy for training sets of varying time windows. We measure accuracy for each topic independently and then report the average accuracy in each hour. That is, 1 hour in the $x$-axis means that the training set consists of messages posted within 1 hour after the topic became a trending topic (and hence, made available to spammers as a potential target), and the testing set consists of messages posted after 1 hour. The $y$-axis shows the accuracy when we use the training set to build a classifier and predict labels of the messages in the testing set. This experiment emulates a real deployment scenario of such a collective attention spam detector, in which partial data is available for predicting future spam. Notice that as the training set grows in size the classification result becomes better. Figure 9(b) shows the false positive rate – in-

Table 5: Combining countermeasure outperformed either the rule-based filtering approach and the supervised classification approach.

| Accuracy (%) | FP | FN | TSD (%) |
|---|---|---|---|
| 97.63 | 0.007 | 0.359 | 54.89 |

dicating how many real non-spam messages are classified as spam messages by the classifier. Overall, the false positive rate is low.

As we have discussed, however, the goal is not only to have high accuracy and low false positives, but also to detect more spam messages as early as possible. In Table 4 we present a sample of the detection results, along with the average result across all 101 topics. Each topic's best training time varies depending on the volume of generated messages and the number of spam messages before the training time. Overall, building a classifier with the first five hours' messages gives us 97.57% accuracy, 0.006 FP, 0.384 FN and 50.16% total spam detected (i.e., how many spam messages out of all spam messages in the topic a classifier detected correctly). Not only does this countermeasure outperform the rule-based filtering approach (50% TSD versus 20% TSD), but it has the advantage of being adaptable to future spammer behaviors (so long as the feature set is maintained). We also observe a high variability in the TSD across topics; some topics are easy for spam detection (with $TSD > 80\%$), while others are very difficult. This suggests that our preliminary feature set could be refined to better target these difficult-to-detect cases.

## Combining Countermeasures

Finally, we consider the effectiveness of combining both countermeasures (rule-based + supervised classification). Does rule-based filtering detect spam messages that a classifier would misclassify? For this combination, we first apply the rule-based filter and then apply the supervised classifier to the remaining messages.

Table 5 presents the evaluation result of the combined spam detection approach across 101 topics, achieving 97.63% accuracy, 0.007 FP, 0.359 FN and 54.89% TSD. We can observe that the combined approach outperformed either the rule-based filtering approach and the supervised classification approach.

We evaluate this combined approach from the perspective of our users accessing collective attention information in the system. Returning to the spamness measure (again, which indicates the prevalence of spam items in the top-k results accessed by users), we evaluate the quality of the recency-based and relevance-based information access approaches both with and without the combined countermeasure.

For this experiment, we assume that a user issues a topic as a query (a hashtag in Twitter domain or a phrase) once per minute. For the recency approach, the system returns the 10-most recently posted messages. For the relevance-based approach, the system first retrieves all relevant messages posted within the past one hour and then ranks messages (by grouping popular payloads, ranking by their occurrence count, and then removing duplicates to maintain diversity). In this experiment, the combined countermeasure reduces spamness by average 59% for the recency-based approach and by average 73% for the relevance-based approach across all 101 topics.

## Summary

Through our twofold approach – data-driven modeling coupled with evaluation over a system trace – we have seen that social systems are extremely susceptible to collective attention spam. With spammers accounting for only 5% of all users, we have found that every legitimate user can be exposed to spam. At even higher spammer penetration, the social system becomes unusable with spam dominating. We have also seen how this threat to collective attention can be augmented through strategically coordinated spammer behaviors to selectively push particular spam payloads, increasing the exposure of legitimate users to spam content. While daunting, we have seen preliminary evidence that carefully-crafted countermeasures may be effective deterrents to collective attention spam – based on high accuracy (up to 98%) and spamness reduction (up to 73%) with a low false positive rate (meaning few non-spam messages are incorrectly labeled). We found that it is possible to filter collective attention spam messages by learning from early-age spam messages in a topic. And since the countermeasures using rules and supervised classification are relatively lightweight, these methods can be applied for near real-time spam filtering.

An open question is how to verify that the spam messages in the first few hours used to bootstrap the learning approach are indeed spam. We're considering two approaches: (i) filtering spam messages by URLs based on Blacklists; (ii) using crowd workers in crowdsourcing sites to label samples of early messages containing a popular topic.

# Related Work

Threats to information systems have been omnipresent for years. We characterize two related streams of research to collective attention spam: email spam and social spam.

*Email spam:* To prevent and detect email spam, many approaches have been developed, including content-based filtering like whitelisting, blacklisting, keyword-based, statistical classification (Androutsopoulos et al. 2000), heuristic-based filtering (Team 2004), collaborative filtering (Prakash 2004), network-level clustering approach (Qian et al. 2010), spambot identification (Stringhini et al. 2011), and behavioral blacklisting (Ramachandran, Feamster, and Vempala 2007). Researchers have also analyzed the network-level characteristics of spammers (Ramachandran and Feamster 2006), the underlying business operations of spam-advertised enterprises (Kanich et al. 2011) and common spam in tweets and email (Lumezanu and Feamster 2012), have quantified the effect of email spam on behavior and engagement of email users (Dasgupta et al. 2012), and have studied the spam value chain (Levchenko et al. 2011).

*Social spam:* Several research efforts have found a high degree of reciprocity in social networks (e.g., (Kwak et al. 2010)), meaning that many users may elect to make themselves susceptible to a spammer (e.g., by becoming "friends" and subsequently the target of spam messages). Jagatic et al. (Jagatic et al. 2007) have shown that adding "social" contextual clues (like sending a spam message from a known "friend" account) can increase the effectiveness of such attacks. Similarly, Brown et al. (Brown et al. 2008) showed that context-aware attacks in social systems are very effective. Other types of social spam have been described and solutions proposed. Examples include Twitter-based threats like link farms (Ghosh et al. 2012) and trend-stuffing (Irani et al. 2010), video spam (Benevenuto et al. 2009), and tag spam (Koutrika et al. 2008).

Complementary to many of these existing spam threats, collective attention spam relies on the users themselves to seek out the content where the spam will be encountered. And since collective attention spam is often bursty and unexpected, it is important to understand how effectively a countermeasure may be deployed to limit its impact, particularly as collective attention begins to coalesce.

## Conclusion

In this paper, we have presented a dual study of the robustness of social systems to collective attention threats through both a *data-driven modeling* approach and *deployment over a real system trace*. We have explored the resilience of large-scale social systems to threats to collective attention, observing that relevance-based access methods are more robust than recency-based ones and that only slight increases in the fraction of spammers in a system can fundamentally disrupt the quality of information. We have identified two countermeasures – rule-based filtering and supervised classification – and demonstrated their effectiveness at filtering spam during the early development of a bursting phenomenon in a real system.

## Acknowledgements

This work was supported in part by NSF grant IIS-1149383 and AFOSR Grant FA9550-12-1-0363. Any opinions, findings and conclusions or recommendations expressed in this material are the author(s) and do not necessarily reflect those of the sponsors.

## References

Androutsopoulos, I.; Koutsias, J.; Chandrinos, K. V.; and Spyropoulos, C. D. 2000. An experimental comparison of naive bayesian and keyword-based anti-spam filtering with personal e-mail messages. In *SIGIR*.

Benevenuto, F.; Rodrigues, T.; Almeida, V.; Almeida, J.; and Gonçalves, M. 2009. Detecting spammers and content promoters in online video social networks. In *SIGIR*.

Brown, G.; Howe, T.; Ihbe, M.; Prakash, A.; and Borders, K. 2008. Social networks and context-aware spam. In *CSCW*.

Dasgupta, A.; Punera, K.; Rao, J. M.; and Wang, X. 2012. Impact of spam exposure on user engagement. In *USENIX Security*.

Ghosh, S.; Viswanath, B.; Kooti, F.; Sharma, N. K.; Gautam, K.; Benevenuto, F.; Ganguly, N.; and Gummadi, K. P. 2012. Understanding and combating link farming in the twitter social network. In *WWW*.

Goetz, M.; Leskovec, J.; McGlohon, M.; and Faloutsos, C. 2009. Modeling blog dynamics. In *ICWSM*.

Hong, L.; Dan, O.; and Davison, B. D. 2011. Predicting popular messages in twitter. In *WWW*.

Irani, D.; Webb, S.; Pu, C.; and Li, K. 2010. Study of trend-stuffing on twitter through text classification. In *Collaboration, Electronic messaging, Anti-Abuse and Spam Conference (CEAS)*.

Jagatic, T. N.; Johnson, N. A.; Jakobsson, M.; and Menczer, F. 2007. Social phishing. *Commun. ACM* 50(10):94–100.

Järvelin, K., and Kekäläinen, J. 2000. Ir evaluation methods for retrieving highly relevant documents. In *SIGIR*.

Kanich, C.; Weavery, N.; McCoy, D.; Halvorson, T.; Kreibichy, C.; Levchenko, K.; Paxson, V.; Voelker, G. M.; and Savage, S. 2011. Show me the money: characterizing spam-advertised revenue. In *USENIX Security*.

Koutrika, G.; Effendi, F. A.; Gyöngyi, Z.; Heymann, P.; and Garcia-Molina, H. 2008. Combating spam in tagging systems: An evaluation. *ACM Trans. Web* 2(4):1–34.

Kwak, H.; Lee, C.; Park, H.; and Moon, S. 2010. What is twitter, a social network or a news media? In *WWW*.

Lee, K.; Caverlee, J.; Kamath, K. Y.; and Cheng, Z. 2012. Detecting collective attention spam. In *WWW Workshop on Web Quality (WebQuality)*.

Lee, K.; Eoff, B. D.; and Caverlee, J. 2011. Seven months with the devils: A long-term study of content polluters on twitter. In *ICWSM*.

Lehmann, J.; Gonçalves, B.; Ramasco, J. J.; and Cattuto, C. 2012. Dynamical classes of collective attention in twitter. In *WWW*.

Lerman, K., and Ghosh, R. 2010. Information contagion: An empirical study of the spread of news on digg and twitter social networks. In *ICWSM*.

Levchenko, K.; Pitsillidis, A.; Chachra, N.; Enright, B.; Félegyházi, M.; Grier, C.; Halvorson, T.; Kanich, C.; Kreibich, C.; Liu, H.; McCoy, D.; Weaver, N.; Paxson, V.; Voelker, G. M.; and Savage, S. 2011. Click trajectories: End-to-end analysis of the spam value chain. In *IEEE Symposium on Security and Privacy*.

Lumezanu, C., and Feamster, N. 2012. Observing common spam in tweets and email. In *IMC*.

Prakash, V. V. 2004. Vipul's razor.

Qian, Z.; Mao, Z. M.; Xie, Y.; and Yu, F. 2010. On network-level clusters for spam detection. In *NDSS*.

Ramachandran, A., and Feamster, N. 2006. Understanding the network-level behavior of spammers. In *SIGCOMM*.

Ramachandran, A.; Feamster, N.; and Vempala, S. 2007. Filtering spam with behavioral blacklisting. In *CCS*.

Ratkiewicz, J.; Conover, M.; Meiss, M.; Gonçalves, B.; Flammini, A.; and Menczer, F. 2011. Detecting and tracking political abuse in social media. In *ICWSM*.

Romero, D. M.; Meeder, B.; and Kleinberg, J. 2011. Differences in the mechanics of information diffusion across topics: idioms, political hashtags, and complex contagion on twitter. In *WWW*.

Stringhini, G.; Holz, T.; Stone-Gross, B.; Kruegel, C.; and Vigna, G. 2011. Botmagnifier: Locating spambots on the internet. In *USENIX Security*.

Team, S. D. 2004. The apache spamassassin project.

Thomas, K.; Grier, C.; Song, D.; and Paxson, V. 2011. Suspended accounts in retrospect: an analysis of twitter spam. In *IMC*.

Wu, F., and Huberman, B. A. 2007. Novelty and collective attention. *Proceedings of the National Academy of Sciences* 104(45):17599–17601.