Link Spamming Wikipedia for Profit

Andrew G. West, Jian Chang, Krishna Venkatasubramanian, Oleg Sokolsky, and Insup Lee
Dept. of Computer and Information Science - University of Pennsylvania - Philadelphia, PA
{westand, jianchan, vkris, sokolsky, lee}@cis.upenn.edu

ABSTRACT

Collaborative functionality is an increasingly prevalent web technology. To encourage participation, these systems usually have low barriers-to-entry and permissive privileges. Unsurprisingly, ill-intentioned users try to leverage these characteristics for nefarious purposes. In this work, a particular abuse is examined – link spamming – the addition of promotional or otherwise inappropriate hyperlinks.

Our analysis focuses on the *wiki* model and the collaborative encyclopedia, Wikipedia, in particular. A principal goal of spammers is to maximize *exposure*, the quantity of people who view a link. Creating and analyzing the first Wikipedia link spam corpus, we find that existing spam strategies perform quite poorly in this regard. The *status quo* spamming model relies on link persistence to accumulate exposures, a strategy that fails given the diligence of the Wikipedia community. Instead, we propose a model that exploits the latency inherent in human anti-spam enforcement.

Statistical estimation suggests our novel model would produce significantly more link exposures than *status quo* techniques. More critically, the strategy could prove economically viable for perpetrators, incentivizing its exploitation. To this end, we address mitigation strategies.

Categories and Subject Descriptors

H.5.3 [**Group and Organization Interfaces**]: *collaborative computing, computer-supported cooperative work*;
K.6.5 [**Management of Computing and Information Systems**]: Security and Protection

Keywords

Web 2.0 spam, link spam, Wikipedia, wikis, collaborative security, attack model, measurement study, spam economics.

1. INTRODUCTION

Wikipedia [13], the collaborative encyclopedia, is the 6^{th} most popular Internet site as of this writing [1], with its English edition receiving 79 billion hits last year [15]. A distinguishing feature of Wikipedia, and most public-facing *wikis*, is that nearly *all* content can be modified by *any* participant. For Wikipedia this barrier-to-entry is effectively zero – anonymous users are free to contribute.

Unsurprisingly, malicious users see this high popularity and open infrastructure as an asset when mounting attacks. Common forms of attack on Wikipedia include the insertion of obscenities and the deletion of content. Recent years have seen much work focused on detecting damaging changes to Wikipedia [21, 22, 42, 43, 51]. However, this body of research has largely ignored the placement of promotional or otherwise inappropriate external links to Wikipedia. Such behavior constitutes *link spam* – the focus of this paper. The allure of such behaviors is obvious. Assuming a spammer can convince a user to click on a spam link (*i.e., click-through*), link spam could direct significant traffic to a *landing site*. From visitors, the site can yield monetary or other benefits.

Wikipedia already has mechanisms in place to combat such behavior. However, few of these mechanisms are automated, making it the responsibility of human editors to monitor link additions. As Wikipedia's popularity and reputation continue to grow, so do the incentives for abuse [30].

Historically, Wikipedia has dealt with varying spam models. Before enabling HTML `nofollow` for outgoing links, Wikipedia was often linked spammed for search-engine optimization (SEO) purposes [48]. In this manner, Wikipedia links could garner *indirect* traffic for a landing site by accumulating backlinks to improve search-engine rank.

Nowadays, spam models must have more *direct* intentions. Thus, the principal goal of Wikipedia link spam is maximizing *exposure*, the quantity of people who view a spam link. The *status quo* means of gaining exposure is to use subtle tactics in the hope that a link can become *persistent* in an article (*i.e.,* have a long lifespan). We find the impact of this strategy is minimal, with spam links receiving a median of 6 views before removal. This poor performance is a result of Wikipedia's diligent editors. Consequently, link spamming models relying on persistence are not particularly successful.

This lack of success motivated us to research whether more effective spam models might exist for Wikipedia and *wikis* in general. To this end, we describe a spam model that acknowledges and exploits the short lifespan of spam links. The model maximizes link exposure by leveraging the latency inherent in human anti-spam enforcement. Spam campaigns based on our model leverage four attack vectors:

1. High-traffic placement: Using popular pages.
2. Registered accounts: Gaming the privilege delegation system, accounts can be obtained that can edit rapidly and in a programmatic fashion.
3. Blatant nature: Prominently placed/styled links solicit reader attention and increase click-through.
4. Distributed: Distributed hosts provide the IP agility needed to sustain spam attacks at scale.

CEAS '11, September 1–2, 2011, Perth, Western Australia, Australia.

Our work begins by examining Wikipedia's existing anti-spam infrastructure and producing/analyzing the first ever Wikipedia link spam corpus. A measurement study reveals much about *status quo* spam behaviors, but provides no basis for evaluating our novel spam model. Instead, we rely on statistical estimation to demonstrate that the proposal would outperform those techniques currently in use. Further, economic analysis suggests the model might prove financially viable for an attacker.

Given evidence that current anti-spam mechanisms are insufficient, we propose detection strategies. While our analysis concentrates on the English language Wikipedia, the vulnerabilities identified are likely present in other Wikipedia editions, *wiki* software [6], and collaborative applications on the whole. Thus, mitigating such threats holds enormous importance for the survival of the collaborative paradigm.

The principal contributions of this work are:

1. Creating/analyzing the first Wikipedia spam corpus.
2. Identifying Wikipedia's link spam vulnerabilities.
3. Proposing solutions to patch these vulnerabilities.

Moving forward, we provide background information and review terminology (Sec. 2) before examining the status quo of Wikipedia spam behaviors (Sec. 3). Then, we propose a spam model and estimate its practical effectiveness (Sec. 4). Finally, we propose mitigation strategies (Sec. 5), discuss related work (Sec. 6), and conclude (Sec. 7).

2. BACKGROUND

In this section, preliminaries are handled. First, terminology is standardized; both Wikipedia vocabulary (Sec. 2.1) and spam-specific (Sec. 2.2). Then, the anti-spam techniques employed by Wikipedia are examined (Sec. 2.3).

2.1 Wikipedia Terminology

Wikipedia [13] is a collaborative encyclopedia, whose content is composed of *articles* or *pages*. Individuals who create/modify these articles are called *contributors* or *editors*. Editors can be *registered* (have a username/password), or choose to edit *anonymously* (with only their IP displayed). Taken as a whole, the user-base is referred to as a *community*. An article's history consists of a series of *versions*, and the individual changes introduced at each step are termed *revisions* (often visualized as a `diff`). A *revert* occurs when an editor restores a previous version of an article.

Articles may contain hypertext links to other content. When an article links to an internal article, it is a *wikilink*. Off-encyclopedia links are *external links*, which are the primary focus of this work. A syntax [16] defines external links, and we investigate only well-formed links of this kind[1].

2.2 Defining Wikipedia Link Spam

Put simply, a spam link is one that violates Wikipedia's (subjective) external link policy [16]. An objective definition for external link spam on Wikipedia is beyond the scope of this work. An external link can be inappropriate because of either its *destination*, its *presentation*, or both.

Inappropriate *destinations* (URLs) are intuitive. Linking for commercial or promotional purposes is prohibited. Further, blogs, personal webpages, and fan sites usually lack the credibility and notoriety necessary to justify linking. A link's *presentation* concerns its description text and placement within an article. It should be emphasized that Wikipedia spam need not be commercial in nature, rather, it just must be undesirable traffic (and thus, analogous to many definitions of email spam).

Fortunately, a precise definition of spam is unnecessary for analysis. When building a spam corpus (Sec. 3.1), we leverage the power of community experts to make the spam/ham distinction. When we propose a spam model (Sec. 4), we assume the external links are unambiguously inappropriate.

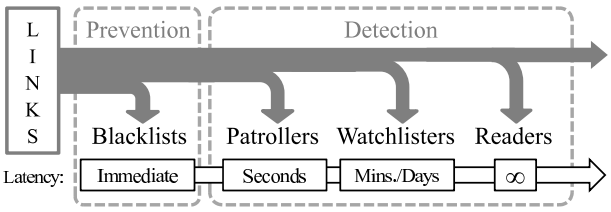

Figure 1: Pipeline for single-link spam detection

2.3 How Wikipedia Mitigates Link Spam

Wikipedia has protections to prevent/detect external link spamming. Mechanisms are classified at four granularity: (1) single link, (2) account, (3) collective, and (4) bot.

An understanding of these protections is necessary to capture why the spam model of Sec. 4 is a viable threat. Discussion is Wikipedia-centric[2], but other collaborative applications are likely to employ a subset of this functionality.

2.3.1 Single Link Mitigation

Entities involved in the removal of a single-link spam instance are best visualized as a pipeline (see Fig. 1):

- BLACKLISTS: URL *blacklists* are a preventative mechanism that reject edits adding prohibited URLs. Both a local and global blacklist are employed [12], containing ≈17k entries (regular expressions) in combination.
- PATROLLERS: The first human defenses are *patrollers*, gate-keepers who monitor recent changes, usually without regard for subject matter. Software tools [29, 49] employing some prioritization mechanism often assist in this process, although none have functionality specific to link spam detection.
- WATCHLISTERS: Every Wikipedia user has a self selected *watchlist* of articles and are notified when one is modified. Given that an individual has indicated interest in an article, one can presume they have some incentive to ensure changes are beneficial.
- READERS: Spam edits passing the previous stages are said to be *embedded* or *persistent*. Now, only readers of an article are likely to encounter the damage. The discoverer may choose not to undo the damage for reasons of apathy, unfamiliarity with the editing system, *etc.*. Anecdotal evidence suggests the average reader is *unlikely* to fix the problem [23].

The spam model proposed in Sec. 4 assumes patrollers will detect link spam additions and only places links to non-blacklisted URLs (the list is publicly-viewable).

[1]Wikipedia "spam" is broader than well-formed external links. URLs might be provided in plain-text or entire articles could be commercially motivated. We do not study alternative strategies.

[2]We do not intend to provide a comprehensive explanation of Wikipedia's anti-spam mechanisms. Both generalizations and omissions are made for the sake of succinctness and readability.

CORP	SIZE	CONTAINS
C_{damage}	204k	Edits that: (1) were rolled-back.
C_{spam}	4.7k	Edits that: (1) added exactly one link, (2) were rolled-back, and (3) passed manual inspection.
C_{ham}	182k	Edits that: (1) added exactly one link and (2) were *not* rolled-back.

Table 1: Summarizing project corpora

2.3.2 Account Mitigation

An account consistently demonstrating spam tendencies will be blocked. When a user is caught making a damaging edit, they are issued a warning. Subsequent damage generates sterner warnings, eventually leading to a block [29]. The spam model proposed in Sec. 4 is cognizant of this warning/blocking process. By adding links as rapidly as possible, we take advantage of human latency – enabling one to have inserted *many* links before sufficient detections and warnings have accumulated to warrant a block.

2.3.3 Collective Mitigation

A user may create multiple accounts or use multiple IP addresses to evade blockage ("sock-puppets"). Similarly, distinct human users may collaborate for some malicious purpose. Aside from manual signature detection, the `checkuser` tool exists to detect these behaviors [7]. Checkuser's can investigate the IP addresses and user-agent strings that accounts use to draw correlations and block entire IP ranges.

The spam model proposed in Sec. 4 stipulates that a broad and sustained spam effort would require a large quantity of diverse hosts (*e.g.,* open proxies or a botnet) to have the IP agility needed to avoid range blocks.

2.3.4 Bot Mitigation

Malicious script-driven *bots* pose a severe threat in *wiki* environments. Such tools operate at zero marginal cost and can outpace human users. Thus, methods are employed to distinguish humans from bots. First, *rate-limits* are employed, and privileged accounts are allowed to edit at greater speed. Second, *CAPTCHAs* are employed in sensitive situations. Third, there are a number of *software extensions* [7] which claim protection against bot attackers.

Our spam model proposal (Sec. 4) employs autonomous bots. However, by manipulating the privilege-granting system, one can create accounts with high rate-limits that are unaffected by software protections.

3. STATUS QUO OF WIKIPEDIA SPAM

Here, we seek to understand the *status quo* of Wikipedia link spam behaviors. To this end, we create and analyze the first ever Wikipedia link spam corpus. Analysis reveals an abundance of subtle tactics, indicating a desire for *persistent* (*i.e.,* long-living) links. The efficiency of these strategies is then measured, where *efficiency* is defined to be the number of exposures a link receives before it is removed. We find current techniques perform quite poorly in this regard.

Moving forward, the spam corpus is described (Sec. 3.1) and used to analyze URL destinations (Sec. 3.2) as well as the perpetrators (Sec. 3.3). Then, the impact and efficiency of these spam instances is quantified (Sec. 3.4), before broadening the search for efficient strategies (Sec. 3.5).

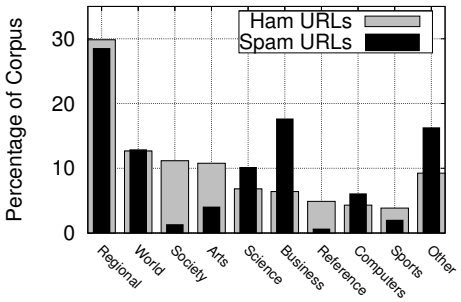

Figure 2: Links by ODP [10] category

3.1 Creating a Link Spam Corpus

To build a link spam corpus, experienced Wikipedia users are relied upon. A functionality called *rollback* (an expedited form of *revert*) is delegated to trusted users and used to undo "blatantly unproductive" contributions. Existing work [51] describes how rollback can be detected and used to flag *damaging edits*. Rollback-based labeling is advantageous because it is: (1) autonomous, (2) performed by experts, and (3) allows a case-by-case definition of "damage."

Edits flagged in this way contain diverse forms of damage, and the link spam subset must be extracted. First, damaging edits with *exactly one* external link addition are identified. Then, those revision `diffs` are manually inspected, discarding edits where there are content changes besides the link addition. Remaining revisions form the spam corpus. Intuitively, we know these to be spam edits because the link is the *only* change made, and therefore the decision to rollback the edit speaks directly to the inappropriateness of that one link. We emphasize that links handled in this fashion are not the result of simple editorial decisions, which would not be a justifiable use of the *rollback* tool.

During 2 months of data collection in mid-2010 there were 7.4 million article-edits to English Wikipedia. Of these, 265k (3.59%) added *at least* one external link and 188k (2.55%) added *exactly* one link. Of the 7.4 million edits, 204k (2.77%) were undone via rollback[3] (forming corpus, C_{damage}). The intersection between "one-link-added" and "rolled-back" sets is 6.1k revisions. After inspection[4], 4,756 edits form corpus, C_{spam}. A complementary set, C_{ham}, contains "one-link-added" edits *not* undone via rollback.

Tab. 1 summarizes the project corpora. Corpora were amassed by processing Wikipedia edits in real-time using the STiki framework [49], built atop the MediaWiki API [5].

3.2 Link Spam Destinations

Analysis begins with the destinations (URLs) of C_{spam}. First, destination genres are characterized and the most abusive pages identified. Then, corpus links are correlated with those from other spam-prone environments.

3.2.1 Genres of Spam Links

First, corpus domains are mapped to their Open Directory Project (OPD) [10] categorization, where available. Per Fig. 2, observe that spam URLs encompass a breadth of categories. Unlike in other environments (*e.g.,* email) where spam typically has a narrow scope [34], Wikipedia spam is diverse. Unsurprisingly, it is "business" sites which are

[3]Rollback actions were processed for one month after the last corpus edit was made, giving editors time to vet/label content.
[4]Manual inspection at this scale is non-trivial. The "Offline Review Tool" included in [49] was authored/utilized for this task.

DESTINATION PROPERTY	%-age
Commercial storefront	15.5%
Local directory or tourism	7.8%
Social media destinations	2.4%
Foreign language page	1.9%
Adult or offensive link	0.5%

PLACEMENT PROPERTY	%-age
Link uncorrelated w/article	0.9%
Unusual link placement	0.9%
Visual manipulation of link	0.2%

Table 2: Characterizing spam edits

DOMAIN	SPAM	HAM	SPAM-%
`www.youtube.com`	101	3058	3.2%
Area code look-up	72	11	86.7%
`www.facebook.com`	48	3294	1.5%
Cinematic rankings	41	8	83.6%
`www.billboard.com`	35	1340	2.6%
Soccer statistics	29	7	80.5%

Table 3: Domains w/most spam occurrences[5]

spammed the most relative to their ham quantity. This ODP mapping, however, is too coarse. Thus, manual investigation was performed on 1,000 random edits from C_{spam}. While a taxonomy of behaviors is beyond the scope of this work, Tab. 2 shows the prevalence of several characteristics.

Profitability is central to any successful spam model [33]. Thus, it is surprising to see that only 15% of destinations have product(s) immediately for sale. However, profit need not be monetary, nor need it be earned directly. For example, sites could earn referral commissions, monetize via ad revenue, or fulfill some political/narcissistic agenda.

Nearly three-quarters of the corpus is unclassified in Tab. 2 because it is difficult to pinpoint the intention of a link/site. A common tactic observed was "information adjacent services." For example, a site will provide encyclopedic information (*e.g.,* the science of LASIK eye surgery) but also sell a related service (*e.g.,* the surgery itself). This combination of self-interest and encyclopedic-interest complicates our categorization. Similarly, such ambiguity would apply to the Wikipedia editors who review these links.

The takeaway from this analysis is: (1) Wikipedia link spam is categorically diverse, and (2) spammers may be attempting to mask their malicious intentions. In this manner (*i.e.,* by being subtle), they hope their edits will not draw attention and can become persistent on the page.

3.2.2 Link Presentation

Link placement is also analyzed (see Tab. 2). First, we find the vast majority of spam links are topic-similar to the Wikipedia articles on which they were placed. There is no evidence of "blanket spamming." Second, we find that spammers are overwhelmingly adherent to link style conventions. Guidelines require sources be placed in an "external links" section near the article bottom or specially handled as an in-line citation. In less than 1% of cases was this violated (*e.g.,* links atop an article). Equally rare were attempts to manipulate link appearance (*e.g.,* using a prominent font).

This adherence suggests spammers are attempting to gain persistent links. After all, if spammers had short-term goals, they would abandon subtlety and use prominently placed and/or styled links to solicit reader attention and increase click-through rates (as we propose in Sec. 4.1.3).

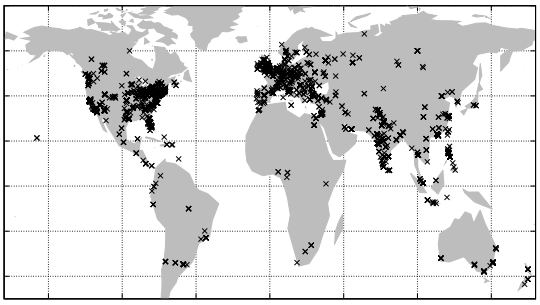

Figure 3: Geographical dist. of spamming IPs

3.2.3 Most Abusive URLs/Domains

It is intuitive to examine URLs/domains that appear multiple times in the spam corpus. Of the 4.7k members of C_{spam}, there are 3.8k (80%) unique URLs and 2.3k (50%) unique domains. Some domains are heavily linked, so a raw count of spam occurrences can be misleading. For example, `www.billboard.com` (an authoritative source in the music industry) had 35 spam occurrences, 5^{th} most among domains, but 1340 C_{ham} occurrences. Clearly, it is more helpful to examine spam-to-ham ratios. Tab. 3 shows raw counts and these ratios for domains with the most spam incidents[5].

Adding multiple links to a domain is an obvious way to increase link exposure (and therefore, efficiency). However, this behavior happens rarely – only 14 domains appear 10+ times in C_{spam} and 71% of domains appear just once. Further, few of these domains have maximized their utility – just 2 of the 25 "worst" domains are blacklisted.

3.2.4 Blacklist Correlations

Wikipedia uses purely internal mitigation mechanisms. Therefore, to maximize utility, a malicious user might reuse domains in multiple environments. To quantify this, the correlation between C_{spam} and two URL blacklists was measured in a real-time fashion: (1) Spamhaus' DBL [9] (domains in spam email bodies) and (2) Google's Safe Browsing lists [2, 45] (malware and phishing URLs).

Correlation was extremely poor. Of the 4.7k corpus links, just 9 (0.19%) were DBL listed. The Safe Browsing lists had 5 link matches (0.11%), all "malware." The lack of correlation is an indication that those conducting Wikipedia spamming are unique from those involved in email spam.

3.3 Spam Perpetrators

We now analyze the editors that placed C_{spam} links. The users' registration status and geographic location are examined, before highlighting the most abusive individuals.

3.3.1 Registration Status & Location

57% of C_{spam} links were made anonymously (compared to 25% of C_{ham}). Clearly, a majority of spammers are unaware of the benefits extended to registered users (see Sec. 4.1.2).

IP-based spammers[6] can be geo-located, as visualized in Fig. 3. This distribution is consistent with both English-speaking populations and regions commonly associated with spam activity [34]. Thus, more fine-grained analysis of the user-base is necessary before drawing conclusions.

[5]Some domains are not made explicit because: (1) they are not well known, (2) they should not be discredited (it is impossible to know *who* spammed them), and (3) to avoid additional exposure.
[6]Wikipedia does not disclose the IPs of registered users.

3.3.2 Repeat Offenders

Users adding multiple C_{spam} links are an intuitive point of focus. For the 4.7k spam links there are 2.6k unique editors, with 1.8k (70%) adding just one link and 47 users (1.8%) adding 10+ spam instances. The single worst contributor added 41 spam links, with little other account activity.

Poor contributors tend to spam exclusively one domain, and the worst users map well onto the worst domains in Tab. 3. Often, usernames are indicative of a conflict-of-interest (COI) (*i.e.,* the username matches the domain name, verbatim). Such COIs make explicit an intuitive notion: spam links are motivated by self-interest.

These repeat offenders present some of the strongest indications of efficient behaviors. Many repeat offenders are dedicated spam accounts, most of which maximized their utility by spamming until blocked. However, as shown in the next section, existing accounts still fail to optimize the objective function: link exposures. In contrast to the efficient accounts of our spam model, existing spam accounts do not appear to be mechanized (see Sec. 4.1.2), as they add links at speeds consistent with human operation.

3.4 Measuring Spam Impact

As our corpus demonstrates, some quantity of spam links are added to Wikipedia. However, quantity is a poor measure of impact. Instead we measure: (1) spam survival time, to gauge mitigation latency, and (2) spam exposures, to measure the efficiency of link spam efforts.

3.4.1 Spam Survival Time

When talking about the lifespan of a spam link (or damage), it is the *active duration* which is of interest – the interval when the link was visible in the most recent version of an article. Fig. 4 visualizes the CDF of lifespans for both C_{spam} and C_{damage} edits. Spam edits have a median active duration of ≈19 minutes (1164 seconds) which is an order of magnitude larger than damaged edits, at 85 seconds (recall that C_{damage} contains all forms of poor behavior).

This gap in detection times is not unexpected. One generally inspects a revision via its `diff`. While text `diffs` may capture obscenities and other forms of damage, only the URL and hypertext are shown for external links. To make a spam/ham distinction one must generally visit the URL destination, a step which users might omit for reasons of laziness or due to fears of malware/obscenity.

3.4.2 Spam Exposures

Link lifespan is not the ideal metric by which to measure spam impact [44]. Links that reside on rarely visited pages are of little worth, no matter their lifetime. Instead, the quantity of *link exposures* is measured, a factor of: (1) the link's lifespan, and (2) the popularity of the article on which it resided. Using hourly per-article statistics [11], we assume uniform hourly distributions to produce view estimates.

Fig. 4 visualizes link exposure quantity. In the median case, spam edits are viewed by 6.05 readers and damaged edits receive 1.47 views. While a difference of over 3× between types, neither figure is significant. Moreover, attempts to isolate "effective strategies" (*e.g.,* the most abusive users) produced similar results.

We also note that the semantics of link exposure calculation overstate the efficiency of *status quo* techniques. The calculation assumes that an article view is equivalent to a link exposure, when in fact there is no guarantee that:

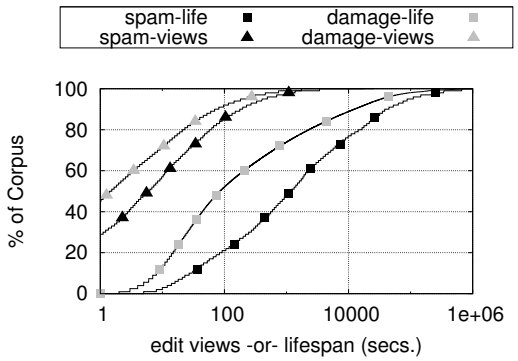

Figure 4: CDF of views and lifespan for corpora. Note the logarithmic scale for the x axis.

(1) the link graphically renders, or (2) the reader notices the link. Recall that spam links tend to follow placement/style conventions (Sec. 3.2.2). Thus, spam links are likely to be at the page bottom (requiring scrolling) and appear adjacent to other identically styled links (not drawing attention).

In contrast, our spam model proposes an aggressive link placement/styling strategy (see Fig. 7 and Sec. 4.1.3), which nearly ensures that an article view corresponds to reader focus on the link. Although difficult to quantify, these differences in strategy would presumably widen the efficiency gap between the current and proposed models.

3.5 Expanding the Investigation's Scope

In the past sections, many aspects of C_{spam} and related corpora have been examined. However, our corpus encompasses only a two-month period, and we cannot be certain that it is representative. Therefore, we now explore alternative sources for Wikipedia spam information.

3.5.1 On-wiki Archives

The archived Wikipedia/Wikimedia blacklist discussion pages [12] and a special task force [14] are the best source regarding historical Wikipedia spam behaviors. Much as corpus analysis revealed, problematic domains and accounts are not unusual – but more aggressive strategies are. There is occasional mention of automated *spambots*, but most incidents occurred before current protection settings.

3.5.2 Media Coverage

One would expect that an attack of sufficient scale would capture the attention of the security community. Though Wikipedia has been an element (phished) in attacks, no publication makes mention of direct spam events.

3.5.3 Deleted Revisions

The `RevDelete` [7] software tool is used to remove revisions from public view (including histories). There are legitimate reasons for deletions, copyright violations and libelous content among them. We were motivated to investigate if spam attacks might be hidden from public view in this manner, as this might prevent copy-cat attacks.

For 6 weeks in late 2010, we stored `diffs` for *every* revision to Wikipedia. This enabled us to view revisions that later appeared on a public block log or exhaustive API [5] requests revealed to be "suppressed" (a stronger, non-logged form of removal). Manual inspection of 2.6k deletions revealed no evidence of spam behaviors (see [52] for what *was* found).

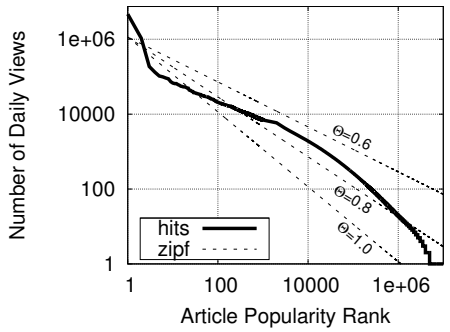

Figure 5: log-log plot of reader distribution

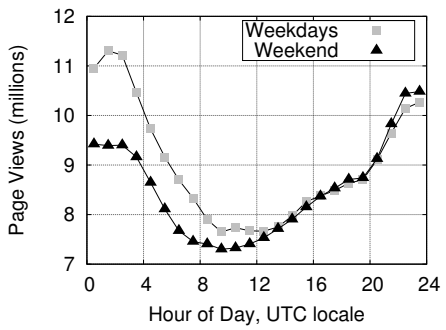

Figure 6: View distribution by UTC hour-of-day

PV_d	TITLE
106k	Wiki
93k	United States
61k	2010 Deaths
60k	Justin Bieber
58k	YouTube

(a)

PV_s	TITLE	DATE
154	The Who	02-07
73	Hockey ... 2010	02-24
55	Dennis Gabor	06-05
51	Pete Townshend	02-07
49	Corey Haim	03-10

(b)

Table 4: (a) Most avg. page-views/day (PV_d). (b) Peak page-views/sec. (PV_s), hour buckets. Both calculated for Jan. through Aug. 2010

4. SPAM MODEL & ATTACK VECTORS

With the last section providing no evidence of efficient or aggressive link spam behaviors, it could be the case that existing anti-spam mechanisms are a sufficient deterrent to attacks. While subtle and persistence-seeking behaviors may not prove viable against these protections, we hypothesize that vulnerabilities are present and can be exploited. In this section, we describe a novel and *efficient* spam model we estimate can significantly outperform *status quo* techniques.

The responsibility of spam mitigation on Wikipedia falls primarily to human users. While Wikipedia's editor community may be adept at detecting subtle link spam strategies, they do have a weakness: *inherent human latency*. Even under ideal conditions it takes several seconds for a human to scan an edit `diff`, visit a link URL, make a spam determination, and complete the revert process. Such short link durations are unhelpful when using subtle spam strategies. However, by embracing these finite windows of opportunity one can deploy an *aggressive* spam model capable of producing a large number of exposures. Moreover, latency will compound over multiple link additions before blacklist or account-level protections can be enacted.

We first describe the attack vectors (Sec. 4.1), before estimating their practical effectiveness (Sec. 4.2). Then, ethical issues of our presentation are discussed (Sec. 4.3).

4.1 Attack Vectors

There are four attack vectors which define our model: First, high-traffic pages are targeted at opportune times (Sec. 4.1.1). Second, privileged accounts are attained and operated autonomously (Sec. 4.1.2). Third, link additions are styled/placed to increase click-through (Sec. 4.1.3). Finally, use of distributed hosts enables evasion (Sec. 4.1.4).

We recognize that these vectors have been seen previously in other domains (*e.g.,* email, social networks [47]). Unique is the mapping of these strategies into a *wiki* setting, with its functional peculiarities and community-driven mitigation.

4.1.1 Targeting Popular Entities

Spam links are often quickly detected (per Fig. 4). This detection speed suggests patrollers discover them. Thus, regardless of where the link was placed, it likely would have been discovered equally as quickly. To maximize link exposures in a fixed duration, one should target: (1) the most popular articles; (2) during peak-traffic.

Popular Articles: As of this writing, Wikipedia averages 15 million hourly views over its 3.4 million articles [15]. If these views were uniformly distributed, Wikipedia would be a less than ideal spamming environment. Instead, as Fig. 5 shows, the distribution is approximately Zipfian.

Thus, large numbers of readers can be reached by targeting few articles. Tab. 4a displays the most popular pages on average, revealing several articles that consistently receive 50k+ hits per day. Tab. 4b shows peak traffic events. Often, these spikes are tied to cultural events. For example, musical act "The Who" and member "Pete Townshend" played the 2010 Super Bowl halftime show; exactly when the associated articles were receiving 200+ views *a second*.

These high view rates are at the core of the proposed attack. As Fig. 5 and Tab. 4 show, several seconds could be sufficient to accumulate a large number of link exposures, *per edit*. When multiple edits are made in the course of a campaign their sum could be immense. It should be noted that many popular pages are placed under "protection" [17] on Wikipedia. However, the account-level vulnerabilities described in Sec. 4.1.2 render these protections ineffective[7].

Peak Traffic Periods: An intelligent attacker will not just target the most popular pages, he/she will target them at the most opportune times. Fig. 6 visualizes Wikipedia's readership by hour-of-day. On the average, peak traffic is achieved on weekdays around 2:00 UTC.

4.1.2 Account Privileges & Autonomy

Here, we describe the illicit advantages of using privileged accounts and how one can autonomously attain these privileges. Having such accounts, we discuss how one could maximize their utility using mechanized operation.

Privileged Benefits: While anonymous editing (identified only by IP address) has a low barrier-to-entry, registered accounts offer both speed and economic benefits to a spammer. Two access-levels [18] are of interest, `anonymous` and `autoconfirmed`, which are compared in Tab. 5.

[7]Accounts cannot edit "fully-protected" pages, which are very rare (*i.e.,* the "Main Page"). We exclude such pages from analysis.

ACCT.	RATE	CAPTCHA	CRITERIA
`anonymous`	≈ 8 edits per min.	At *every* link add.	None
`auto-confirmed`	70+ edits/min.	*One* at acct. creation	Acct. 4+ days old; has 10+ edits

Table 5: Comparing account criteria/privileges

CAPTCHA solves inhibit spammer behavior by: (1) limiting their speed of operation and/or (2) generating cost for solutions [39]. Notice that an `autoconfirmed` user must solve only one CAPTCHA and can do at an insensitive time.

Experiments in Wikipedia's sandbox also revealed that `autoconfirmed` accounts have advantageous rate-limits. While `anonymous` users are limited to ≈8 edits/minute, rates of 70+ edits/minute were achieved with `autoconfirmed` accounts. Further, this does not appear to be a hard limit, but the result of network bottlenecks/latency.

Finally, edits made by registered users are viewed less suspiciously. For example, software inspection tools [3, 49] tend to give edits made by `anonymous` users higher review priority. Further, Goldman [30] writes about how IP users are treated as "second-class citizens."

Becoming Autoconfirmed: While attaining the `autoconfirmed` privilege has benefits, one must reach this status by having 10+ edits and being 4+ days old. When these criteria are met, software will automatically grant the privilege. While the time interval is trivial to overcome, not just *any* 10 edits will suffice, as multiple poor edits could lead to blockage (see Sec. 2.3.2). However, this process can be simplified or evaded in multiple ways.

First, few namespaces are heavily vetted for quality. Discussion pages, user profiles, and sandboxes receive little-to-none of the monitoring of encyclopedic content. Nonetheless, edits in these spaces still count towards an account's total and it is not difficult to imagine how automated scripts might be able to accumulate edits in such settings. Second, even if Wikipedia were to constrain the namespaces for such counts, one could imagine malicious users authoring *helpful* Wikipedia bots to bring malicious accounts to `autoconfirmed` status. Such bots might perform menial yet constructive tasks (*e.g.,* spelling correction).

Mechanized Editing: The ability to launch attacks in a mechanical fashion is a powerful one. This is a core vulnerability: Wikipedia relies on humans for mitigation but edits can be placed at greater-than-human speeds. In particular, once a spam *campaign* is initiated, it is logical for it to edit as quickly as possible until some resource is exhausted (*i.e.,* the account blocked, or the URL blacklisted).

Sec. 2.3.2 discussed that an account is generally blocked after some quantity, x, of damaging edits are detected. An account could methodically place x spam links and then await blockage. Alternatively, by editing rapidly, one exploits the human latency of the detection process. In the time it takes x spam instances to be located and a block enacted, some additional quantity of links can be added.

Mechanization is enabled via the MediaWiki API [5]. Further, anti-bot protections (Sec. 2.3.4) were found insufficient against mechanized attacks. The protections failed in both sandbox experiments and in tests against an identically configured local MediaWiki [6] installation.

Figure 7: Mock-up of link placement/style

4.1.3 Content Placement & Styling

Wikipedia policy states that hyperlinks should be placed in an "external links" section near the bottom of an article and use the default font. This placement and styling is not ideal to solicit reader attention. Instead, links could be prominently located (*i.e.,* atop the article) and typeset (*i.e.,* large and colorful), as demonstrated in Fig. 7.

Intuitively, such blatant presentation should improve click-through rates. Since the attack model assumes that the initial patroller(s) will undo spam edits – and only intends to exploit their latency, not deceive/delay them – it is unlikely such techniques will hasten link removal. While the rendered content is obviously abusive (see Fig. 7), the *wiki* syntax generating it is less straightforward:

```
<p style="font-size:5em;font-weight:bolder">
 [http://www.example.com Example link]</p>.
```

Since patrollers inspect text `diffs`, this markup could be difficult for them to interpret (especially non-technical users). Further, CSS is widely used in Wikipedia and does not appear anomalous. Moreover, the ability to add/modify content at arbitrary positions is fundamental to the *wiki* model.

4.1.4 Distributed Evasion

In order to sustain an attack, a large number of accounts would be necessary. If accounts all operate from the same IP address/range, then the `checkuser` tool (Sec. 2.3.3) could discover this correlation. As a result, account registration could be prohibited and accounts proactively blocked.

The need for a large and diverse pool of IP hosts is not unique to our model (*e.g.,* consider email spam and DOS attacks). Similarly, the methods of acquiring and using such addresses need not be novel. Wikipedia has protections against anonymity networks (*e.g.,* Tor) and open proxies, leaving *botnets* as a likely distribution agent.

4.2 Estimating Model Effectiveness

Since Sec. 3 showed the proposed vectors were not in active use, it is impossible to passively gleam exposure statistics. Thus, we proceed via estimation, a non-trivial task given the model's dependence on human reactions. Provided this, it is not our intention to arrive at exacting predictions. Instead, we seek only to show that the proposal: (1) would outperform *status quo* models, (2) poses a legitimate threat, and (3) could prove economically viable.

Here, we proceed by first estimating administrative responses to the spam model (Sec. 4.2.1), before handling those of casual readers (Sec. 4.2.2). Then, we discuss the model's economic implications (Sec. 4.2.3).

4.2.1 Administrative Response

We conduct analysis at account granularity (*i.e.,* a *campaign* of edits) and are concerned primarily with *active duration*. That is, how long do links last? And how long do accounts/URLs survive before blockage/blacklisting?

Conservatively assume an account and all of its links are removed exactly 1 minute after the campaign is initiated. In that 1 minute, rate-limits would permit 70 links to be placed on popular Wikipedia pages. Using Fig. 5, we estimate that ≈1300 readers would see a link during this 1 minute (assuming the last link added would be visible for only a fraction of a second). In practice, even after a campaign is blocked, its links may remain active on the site. Although editors may work rapidly to undo this damage, they must contend with the same network latency as the attacker. If 70 links were to last for 1 minute *each*, ≈2100 active views would occur.

We believe that campaign durations on the order of 1–2 minutes are a reasonable (albeit cautious) estimation. Fig. 4 showed that "damaging edits" – often blatant acts of vandalism – survived 85 seconds in the median case. Further, these are only single links, not an entire campaign of edits with a necessary administrative response: while anyone can remove a link, only privileged users can actually halt link additions.

Clearly the proposed model outperforms the 6 viewer median (per link) observed in the *status quo* corpus (Sec. 3). Just 2–3 seconds on the most popular pages would equal this total. One minute of exposure would produce an average of 30 views for 70 links/pages, not to mention the higher click-through rates due to prominent placement/style.

4.2.2 Viewer Response

We expect that most exposures would occur during a link's active duration. However, there are alternative ways link views could occur. As [30] observes, Wikipedia content is frequently scraped or used in mashup-like applications. If an article version containing a spam link were obtained in such a fashion, it could become a source of exposure/traffic.

Probably more significant are *watchlisters*, who receive notification whenever a page of interest is edited. While a link may no longer be "active", watchlists will point to changes in version histories. Such "inactive" exposures could prove non-trivial. For perspective, the "Wiki" article has some 2,000 watchlisters. Also realize that to garner watchlist traffic, a link edit need not survive for any meaningful duration.

From the attacker's perspective, it is important to convert viewers into landing site visitors. Initially, one would expect high click-through rates due to novelty. Sustained attacks, however, would desensitize readers and click-through rates might converge to those in familiar domains. For example, [33] reported a click-through rate of 0.71% in their economic study of email spam. Using this, we estimate that campaigns could *consistently* produce *at least* 15–20 landing site visits.

4.2.3 Economic Considerations

The low-barriers to entry of Wikipedia invite a diversity of spam types (Sec. 3.3). Thus, it should be emphasized that the notion of "profiting" from a spam campaign need not be monetary. However, for the sake of economic discussion, we now make precisely that assumption. We begin by considering the costs of mounting campaigns. As Kanich *et al.* [33] note, dedicated spammers are likely to participate in an affiliate program. This removes the burden of business costs (inventory, shipping, *etc.*) and web hosting, while typically returning commissions of 40-50%.

If we assume a spammer already has a distribution infrastructure available (*i.e.,* botnet), this leaves three expenses: (1) A CAPTCHA must be solved at account creation, whose third-party solutions cost as little as $1 per thousand [39]. (2) Domain names must be purchased and replaced if blacklisting occurs, at a cost of $1–$2 each (depending on TLD). (3) Attack scripts must be coded, incurring labor costs. Using available API frameworks, a straightforward implementation can be encoded in < 100 lines of code.

Thus, *marginal* costs could be as little as $1 per campaign (assuming domains cannot be re-used). The larger question is whether these costs can be recouped to produce a positive return-on-investment (ROI). Extrapolating from a measurement study of a "male enhancement pharmacy" [33], the 20 anticipated landing site visitors would have an expected revenue of $5.20. Even after affiliate profit-sharing, this sum exceeds the marginal cost, yielding a profit and non-trivial ROI. These anecdotal figures should be interpreted only as an indication that this threat *could* be viable.

Given this finding, why is the proposed spam model not already in active use? First, there is the possibility that we have introduced previously unknown vectors. This would make proactive mitigation techniques especially pertinent (Sec. 5). More likely, the profits are simply less than those found in other domains. Consider that a single botnet can send 1 *billion* spam emails *per day* [33]. In comparison, English Wikipedia has only had half-a-billion edits in its 10 year history. The massive scale at which decentralized email spam models operate render them far more profitable (in an absolute sense) than possible in a targeted and centralized environment. However, we do not believe this renders our proposal irrelevant. The minimal startup costs and technical simplicity of our model may attract certain attackers.

4.3 Ethical Considerations

It is in no way this research's intention to facilitate damage to Wikipedia or any *wiki* host. The vulnerabilities discussed in this section have been disclosed to Wikipedia's parent organization, the Wikimedia Foundation (WMF). Further, the WMF was notified regarding the publication schedule of this document and offered technical assistance.

In Sec. 5 mitigation strategies are discussed. Straightforward configuration changes and refinement of the privilege-granting system would significantly improve defenses. Moreover, we have independently developed the most technically challenging of our suggestions, a signature-based machine-learning framework [50]. A live implementation of this technique is currently operational on English Wikipedia.

5. PREVENTION & MITIGATION

Having exposed potentially viable attack vectors against *wikis* and Wikipedia, we believe it prudent to propose solutions. Three proposals are discussed: the notion of explicit edit approval (Sec. 5.1), refinement of account privileges (Sec. 5.2), and signature-based detection (Sec. 5.3).

While the first suggestion artificially increases attack latency so that human editors can vet contributions, the latter two proposals aim to remove this burden from humans entirely. Machine-driven detection seems especially pertinent given Goldman's [30] claims of a dwindling labor-force.

However, Wikipedia's community is hesitant of changes that threaten usability and the open-editing model. Therefore, the community's willingness to integrate security mechanisms with its philosophies may prove critical.

5.1 Explicit Approval

Given that the spam model targets human latency, an obvious suggestion is to delay contributions from going live until patrollers can inspect them. Indeed, this is a valid proposal to prevent all damaging contributions, and is called "Pending Changes" or "Flagged Revisions" [24].

The proposal is a controversial one [8], with concerns regarding: (1) usability ("why don't my edits appear?"), (2) the creation of class hierarchy, (3) manpower [30], and (4) coverage. Implementing such treatment for `autoconfirmed` users seems highly unlikely. It is more realistic that only suspicious link-adding edits on the most popular pages (see Fig. 5 and Sec. 5.3) could be handled in this fashion.

5.2 Privilege Configuration

As Sec. 4.1.2 showed, `autoconfirmed` accounts have expansive permissions yet are easy to obtain. An obvious defense focus is refining the privilege-granting system.

Simply increasing the thresholds for `autoconfirmed` status (*i.e.,* requiring >10 edits) seems unwise. Given that certain namespaces are not well-monitored, such a change would be of little consequence. Even if edit counts were constrained to well-vetted content (*i.e.,* the article namespace), it was discussed that malicious users could still autonomously appear human-like by making trivial but helpful edits (Sec. 4.1.2). Alternatively, edit count could be replaced by more fine-grained and quantitative measures [27].

More simply, it would seem helpful to modify the permissions of `autoconfirmed` users. It is excessive that accounts can edit at rates exceeding 70 edits/min. – inconsistent with human operating speeds. Constructive "bots" have a dedicated permission and approvals process. Human-users expedited by software tools could undergo a similar, manually-delegated procedure. Consider that if `autoconfirmed` users were limited to 5 edits/min., it would render the attack 15× less useful than under the current configuration.

5.3 Signature-Based Detection

Given that the spam model is blatant and aggressive in nature, signature-based detection is a logical mitigation choice. Such a system is described and implemented in our recent work [50]. Summarily, a machine-learning framework is used to score the "spam potential" of edits in real-time. Poorly scoring edits can be undone automatically (per community set false-positive tolerances), used to prioritize patrollers' efforts [49], or require explicit approval (as in Sec. 5.1). The technique has been brought live for English Wikipedia and could be easily ported to other languages/projects.

The feature set includes: (1) Wikipedia metadata processing, (2) HTML landing site analysis (including the quantification of "commercial intent" [25, 41]), and (3) third-party data (*e.g.,* from web crawlers [1]). Most importantly, the attack vectors described herein are quantified as features. However, because the attack vectors are not in active use (and not captured in the training corpus), static rules are installed to bring these features into force.

Over subtle *status quo* behaviors our system detected 64% of spam at a 0.5% FP-rate [50]. It is intuitive that the easiest spam instances to detect are those employing blatant strategies – meaning that the spam model proposed herein could be detected with even greater confidence. Moving forward, system development intends to target deeper URL/domain analysis (see [26, 37]), given that the need for non-blacklisted domains forms a significant portion of marginal attack cost.

6. RELATED WORK

Related literature is best divided into two categories: (1) damage to Wikipedia (Sec. 6.1), and (2) spam in non-*wiki* collaborative applications (Sec. 6.2).

6.1 Damage to Wikipedia

While Wikipedia link spam has long been considered a subset of "damaging edits" [44], researchers have tended to ignore link spam when analyzing and building damage detection systems [21, 22, 42, 43, 51]. As a result, link spam mitigation relies heavily on human-driven detection.

While less prevalent in the *status quo*, link spam edits are more interesting than other damaging edits because of their potential financial motives. The potential for profit has not gone unnoticed [30, 36]. Similarly, SEO proponents have described how one might reap these benefits, publishing guides [4] on how to attain persistent Wikipedia links.

6.2 Spam in Collaborative Applications

Wikipedia is the archetype of the *wiki* model, the most generalized example of a collaborative application. Recent history has seen the emergence of *collaborative functionality* and *user-generated content* in web environments. As Heymann *et al.* [32] survey, these features lead to a significant amount of link spam in Web forums [40], blog comments [20], and social networks [28]. Similarly, there are varied proposals on how to mitigate spam in such domains [20, 31, 38]. However, *wiki* environments are distinct from these systems. For example, *wikis* allow arbitrary content placement/deletion and have community mitigation.

XRumer [19] is blackhat SEO software that broadly targets collaborative functionality (including *wikis*) by discovering and auto-completing web forms [39, 46]). Anecdotally [35], XRumer is thought to be a significant source of web spam. However, XRumer's goal is backlink accumulation, not direct click-throughs. As a result, XRumer emphasizes link quantity, rather than targeting environments.

The work most similar to our own pertains to social networks. Much as our proposal targets high-traffic Wikipedia articles, [47] investigates social "hubs" – user's with popular profiles and high connectivity. By exploiting the lack of access-control, the researchers posted content to these profiles and attained a large number of exposures. While [47] examines DDOS and botnet C&C via this channel, it is easy to imagine how it could be re-purposed for link spam.

7. CONCLUSIONS

In this work, we conducted the first systematic study of link spam on Wikipedia. This revealed that the *status quo* strategy of spammers is the use subtle techniques in the hope of attaining persistent links. However, the diligence of Wikipedia's community mitigates this strategy, as the links receive few exposures (*i.e.,* views) before they are reverted.

While the Wikipedia community is capable of detecting even subtle spam behaviors, it has a weakness – the inherent human latency with which it mitigates spam links. In this paper, we described a novel spam model that exploits this latency by using aggressive techniques to maximize resource utility in these brief windows of opportunity. The model is characterized by the targeting of popular articles, use of mechanized accounts, attention-grabbing link placement/style, and distributed hosting.

Unable to passively evaluate the effectiveness of the model, we instead relied on statistical estimation. While precise

estimates are beyond the scope of this work, our analysis revealed that the proposed strategy would: (1) outperform *status quo* spam techniques, (2) cause significant disruption to the encyclopedia, and (3) could prove financially viable to an attacker, motivating sustained attacks at scale.

To this end, we offered solutions on how the vulnerabilities might be patched. Simple configuration changes would have significant impact. Beyond that, we have implemented a signature-driven detection engine which is autonomous and operates in real-time, eliminating the human latency so critical to the proposed spam model's success.

How collaborative and *wiki* applications embrace this result is significant. While our analysis has focused on English Wikipedia, there are other high-traffic targets which could sustain attacks. The willingness of these communities to embrace software protections over human-driven enforcement may have important ramifications for the security of these sites and the collaborative paradigm as whole.

Acknowledgements

This research is supported in part by ONR MURI N00014-07-1-0907. The authors recognize the helpful advice of Sampath Kannan (UPenn, professor), Robert Terrell (UPenn, Office of General Counsel), and Bo Adler (UCSC, Ph.D. student).

References

[1] Alexa Web Info. http://aws.amazon.com/awis/.
[2] Google Safe Browsing API. http://code.google.com/apis/safebrowsing/. (Malware/phishing lists).
[3] Huggle. http://en.wikipedia.org/wiki/WP:HG.
[4] Link building on Wikipedia. http://www.gamblingcashcow.com/link-building-on-wikipedia/. (SEO blog).
[5] MediaWiki API. http://en.wikipedia.org/w/api.php.
[6] MediaWiki (MW). http://www.mediawiki.org/.
[7] MW extensions. http://www.mediawiki.org/Extension_Matrix.
[8] Pending changes: Straw poll. http://en.wikipedia.org/wiki/Wikipedia:Pending_changes/Straw_poll.
[9] Spamhaus Project. http://www.spamhaus.org/.
[10] The Open Directory Project. http://www.dmoz.org/.
[11] Wikimedia statistics. http://dammit.lt/wikistats.
[12] Wikipedia (local) and Wikimedia (global) spam blacklists. http://en.wikipedia.org/wiki/WP:BLACKLIST.
[13] Wikipedia (WP). http://www.wikipedia.org/.
[14] WikiProject spam. http://en.wikipedia.org/wiki/WP:WPSPAM.
[15] Wikistats. http://stats.wikimedia.org/.
[16] WP: External links. http://en.wikipedia.org/wiki/WP:EXT.
[17] WP: Protection policy. http://en.wikipedia.org/wiki/WP:PP.
[18] WP: User access levels. http://en.wikipedia.org/wiki/WP:UAL.
[19] XRumer. http://www.xrumerseo.com/.
[20] S. Abu-Nimeh and T. Chen. Proliferation and detection of blog spam. *IEEE Security and Privacy*, 8:42–47, 2010.
[21] B. Adler, L. de Alfaro, S. M. Mola-Velasco, P. Rosso, and A. G. West. Wikipedia vandalism detection: Combining natural language, metadata, and reputation features. In *CICLing'11 and LNCS 6609*, pages 277–288, February 2011.
[22] B. T. Adler and L. de Alfaro. A content-driven reputation system for the Wikipedia. In *WWW'07*, May 2007.
[23] J. Antin and C. Cheshire. Readers are not free-riders: Reading as a form of participation on Wikipedia. In *CSCW'10: Conf. on Computer Supported Cooperative Work*, 2010.
[24] N. Cohen. Wikipedia to limit changes to articles on people. *New York Times*, page B1, August 25, 2009.
[25] H. Dai, Z. Nie, L. Wang, L. Zhao, J.-R. Wen, and Y. Li. Detecting online commercial intention (OCI). In *WWW'06*.
[26] M. Felegyhazi, C. Kreibich, and V. Paxson. On the potential of proactive domain blacklisting. In *LEET: Proc. of the Conf. on Large-scale Exploits and Emergent Threats*, 2010.
[27] P. K.-F. Fong and R. P. Biuk-Aghai. What did they do? Deriving high-level edit histories in wikis. In *WikiSym'10: Intl. Syposium on Wikis and Open Collaboration*, 2010.
[28] H. Gao, J. Hu, C. Wilson, Z. Li, Y. Chen, and B. Y. Zhao. Detecting and characterizing social spam campaigns. In *CCS'10: Proceedings of the Conference on Computer and Communications Security*, 2010.
[29] R. S. Geiger and D. Ribes. The work of sustaining order in Wikipedia: The banning of a vandal. In *CSCW'10: Proc. of the Conf. on Computer Supported Cooperative Work*, 2010.
[30] E. Goldman. Wikipedia's labor squeeze and its consequences. *Journal of Telecomm. and High Tech. Law*, 8, 2009.
[31] S. Han, Y. yeol Ahn, S. Moon, and H. Jeong. Collaborative blog spam filtering using adaptive percolation search. In *WWE'06: The Wkshp. on the Weblogging Ecosystem*, 2006.
[32] P. Heymann, G. Koutrika, and H. Garcia-Molina. Fighting spam on social web sites: A survey of approaches and future challenges. *IEEE Internet Comp.*, 11(6):36–45, 2007.
[33] C. Kanich, C. Kreibich, K. Levchenko, B. Enright, G. M. Voelker, V. Paxson, and S. Savage. Spamalytics: An empirical market analysis of spam marketing conversion. In *CCS'08: Conf. on Computer and Comm. Security*, 2008.
[34] Kaspersky Labs. Spam in the third quarter of 2010. http://www.securelist.com/en/analysis/204792147/Spam_in_the_Third_Quarter_of_2010.
[35] B. Krebs. Body armor for bad websites. http://krebsonsecurity.com/2010/11/body-armor-for-bad-web-sites/.
[36] C. McCarthy. Amazon adds Wikipedia to book-shopping. http://news.cnet.com/8301-13577_3-20024297-36.html, 2010.
[37] Y. min Wang, M. Ma, Y. Niu, and H. Chen. Spam double-funnel: Connecting web spammers with advertisers. In *WWW'07: Proc. of the 16th World Wide Web Conf.*, 2007.
[38] G. Mishne, D. Carmel, and R. Lempel. Blocking blog spam with language model disagreement. In *AIRWeb'05: Proc. of the Wkshp. on Adversarial Info. Retrieval on the Web*, 2005.
[39] M. Motoyama, K. Levchenko, C. Kanich, D. McCoy, G. M. Voekler, and S. Savage. Re: CAPTCHAs - Understanding CAPTCHA-solving services in an economic context. In *USENIX Security*, August 2010.
[40] Y. Niu, Y. min Wang, H. Chen, M. Ma, and F. Hsu. A quantitative study of forum spamming using context-based analysis. In *NDSS'07: Proc. of the Network and Distributed System Security Symposium*, 2007.
[41] A. Ntoulas, M. Najork, M. Manasse, and D. Fetterly. Detecting spam web pages through content analysis. In *WWW'06*.
[42] M. Potthast, B. Stein, and R. Gerling. Automatic vandalism detection in Wikipedia. In *Advances in Information Retrieval*, pages 663–668, 2008.
[43] M. Potthast, B. Stein, and T. Holfeld. Overview of the 1st Intl. competition on Wikipedia vandalism detection. In *PAN-CLEF 2010 Labs and Workshops*, 2010.
[44] R. Priedhorsky, J. Chen, S. K. Lam, K. Panciera, L. Terveen, and J. Riedl. Creating, destroying, and restoring value in Wikipedia. In *GROUP'07: Proceedings of the 2007 Intl. ACM Conference on Supporting Group Work*, 2007.
[45] N. Provos, P. Mavrommatis, M. A. Rajab, and F. Monrose. All your iFrames point to us. In *USENIX Security*, 2008.
[46] Y. Shin, M. Gupta, and S. Myers. The nuts and bolts of a forum spam automator. In *LEET: Proc. of the 4th Wkshp. on Large-Scale Exploits and Emergent Threats*, 2011.
[47] B. E. Ur and V. Ganapathy. Evaluating attack amplification in online social networks. In *W2SP'09: The Workshop on Web 2.0 Security and Privacy*, 2009.
[48] B. Vibber. http://lists.wikimedia.org/pipermail/wikien-l/2007-January/061137.html. (HTML [nofollow]) enabled).
[49] A. G. West. STiki: A vandalism detection tool for Wikipedia. http://en.wikipedia.org/wiki/Wikipedia:STiki.
[50] A. G. West, A. Agrawal, P. Baker, B. Exline, and I. Lee. Autonomous link spam detection in purely collaborative environments. In *WikiSym '11: Proc. of the 7th Intl. Symposium on Wikis and Open Collaboration*, October 2011.
[51] A. G. West, S. Kannan, and I. Lee. Detecting Wikipedia vandalism via spatio-temporal analysis of revision metadata. In *EUROSEC'10: European Wkshp. on System Security*, 2010.
[52] A. G. West and I. Lee. What Wikipedia deletes: Examining dangerous collaborative content. In *WikiSym '11: 7th Intl. Symposium on Wikis and Open Collaboration*, October 2011.