International Journal of Cooperative Information Systems
Vol. 23, No. 2 (2014) 1441001 (29 pages)

DOI: 10.1142/S0218843014410019

World Scientific
www.worldscientific.com

A Perspective of Evolution After Five Years: A Large-Scale Study of Web Spam Evolution*

De Wang†, Danesh Irani and Calton Pu
College of Computing, Georgia Institute of Technology
Atlanta, Georgia 30332-0765, United States
†*wang6@gatech.edu*

Received 5 March 2013
Accepted 26 March 2014
Published 21 April 2014

Identifying and detecting web spam is an ongoing battle between spam-researchers and spammers which has been going on since search engines allowed searching of web pages to the modern sharing of web links via social networks. A common challenge faced by spam-researchers is the fact that new techniques depend on requiring a corpus of legitimate and spam web pages. Although large corpora of legitimate web pages are available to researchers, the same cannot be said about web spam or spam web pages. In this paper, we introduce the Webb Spam Corpus 2011 — a corpus of approximately 330,000 spam web pages — which we make available to researchers in the fight against spam. By having a standard corpus available, researchers can collaborate better on developing and reporting results of spam filtering techniques. The corpus contains web pages crawled from links found in over 6.3 million spam emails. We analyze multiple aspects of this corpus including redirection, HTTP headers, web page content, and classification evaluation. We also provide insights into changes in web spam since the last Webb Spam Corpus was released in 2006. These insights include: (1) spammers manipulate social media in spreading spam; (2) HTTP headers and content also change over time; (3) spammers have evolved and adopted new techniques to avoid the detection based on HTTP header information.

Keywords: Web spam; evolution; spam corpus.

1. Introduction

Web spam is defined as web pages that are created to manipulate search engines and deceive web users.[1,2] Email has long been the primary method to spread web spam, although spammers are evolving with the times and quickly employing new techniques to spread web spam. One clear trend is the move towards social media due to the ease of sharing information providing more efficient and numerous channels for the growth of web spam. For example, web spam links in friend requests, inbox messages, and news feeds, are redirecting users to advertisement web sites or

*This paper is an extended version of the article originally published in CollaborateCom2012.
†Corresponding author.

other types of malicious web sites. Further, social media sites have redefined the way links are shared with a tendency to share links using URL shortener.[3]

Apart from evolution of web and applications on the web being one of the reasons driving change in web spam, there is a constant evolution of spam as a reaction to defensive techniques introduced by researchers.[4,5] Improvements in defensive techniques used in web spam are enabled by researchers having access to corpora of web spam and being able to collaborate on developing and reporting results on web spam filtering techniques.

Previous studies[5–7] have introduced and studied first large-scale web spam corpus — Webb Spam Corpus 2006 through content and HTTP session analysis. The Webb Spam Corpus 2006[5] is a collection of nearly 350,000 spam web pages, crawled from spam links found in email messages between November 2002 to January 2006. For legitimate web pages, the Standford WebBase project[8] provides topic focused snapshots of web sites, in which the resulting archives are available to the public via fast download streams. After performing classification on those datasets, the experiments results demonstrated that good performance and high efficiency of classification using HTTP session information.

In this paper, we introduce the Webb Spam Corpus 2011, a new corpus of approximately 330,000 spam web pages. We compare this corpus with the previous, and first of its kind, web spam corpus[5] released in 2006. More concretely, we make the following contributions:

First, we create a new large-scale Web spam corpus — Webb Spam Corpus 2011 — which is a collection of approximately 330,000 spam web pages. Web spam links are extracted from spam email messages received between May 2010 to November 2010. Additionally, we also perform data cleansing to remove legitimate pages which may have been inadvertently collected (similar to the data cleansing performed in the prior Webb Spam Corpus by Webb *et al.*[6]).

Second, we analyze the Webb Spam Corpus 2011 from various perspectives. For example, we evaluate the new corpus on three main aspects: redirections, HTTP session information and content. Based on these aspects, we also make insightful observations. For example, when investigating legitimate web link attack in data cleansing, we found that social networks and search engines have become major targets of attacks.

Lastly, we studied the evolution of web spam by comparing Webb Spam Corpus 2011 with Webb Spam Corpus 2006. For redirections, Webb Spam Corpus 2011 has less redirection. Specifically, it has less "302 Found" redirections and location redirection but more iFrame redirections. The host names in redirection chains have new category — social networks sites, which indicates that social media have been manipulated to spread Web spam through hosting profiles, like plug-ins, and widgets. For HTTP session information, the percentages of hosting IP addresses for web spam in the ranges of 63.* –69.* and 204.* –216.* have changed from 45.4% and 38.6% in Webb Spam Corpus to 28.1% and 21.7%, respectively. Additionally, we compared the top 10 HTTP headers in the datasets. In terms of content, there are

few exact content duplications between the datasets. We also compared the contents of the datasets from other content aspects: most popular words, top words based on information gain, and n-gram (n is from 2 to 3) sequences based on frequency. To evaluate the classification performance and feature change over time, we also performed new classification experiments on new dataset.

The remaining of the paper is organized as follows. We motivate the problem further in Sec. 2. Section 3 introduces corpus including the data collection and cleansing methods. Section 4 compares Webb Spam Corpus 2011 with Webb Spam Corpus. Section 5 performs classification comparison on two datasets. We discuss related work in Sec. 6 and conclude the paper in Sec. 7.

2. Motivation

Web spam has received a lot of attention with search engines constantly adjusting techniques to identify web spam[9] and social networks trying to prevent web spam propagating through their networks.[10] With web links being one of the most popular and easiest ways to share information on the web, web spam will remain a problem.

One of the most common technique to fight web spam is using machine learning, more specifically supervised learning techniques, to build classifiers for web spam using headers, content, or link features. As a prerequisite to using such techniques or researching new ones, having access to a large amount of labeled web spam is important and thus we collect, cleanse, and release a corpus of web spam as an enabler for researchers to improve and develop new web spam techniques. A standard corpus released for any number of researchers to use, as is the case with our corpus, allows and encourages collaboration between researchers to share and improve on each others results.

Although the release of the previous Webb Spam Corpus achieved this a number of years ago, we found that web spam has changed significantly enough to warrant an update to the Webb Spam Corpus. Namely, as detection techniques improve, spammers evolve and introduce new techniques to avoid detection. A concrete example of this is popular tools such as URL shortener (which reduce the length of a URL by mapping an identifier on a standard web link to a long URL) were quickly picked-up by spammers as a cheap method of obfuscation and redirection. Further, looking back at the year of 2006, social networks such as Facebook do not exist or are in early stage of startup or microblog sites such as Twitter. Thus, not only do we release the Webb Spam Corpus 2011, with real-time data collection, we also provide an analysis of evolution and major changes we have observed between the 2006 and 2011 version of the Webb Spam Corpus.

3. Webb Spam Corpus 2011

In this section, we introduce the data collection method, as well as the data cleansing process for the Webb Spam Corpus 2011.

3.1. *Data collection*

3.1.1. *Collection method*

We introduce the Webb Spam Corpus 2011 which is available for download for collaborative research investigation and reporting as an .arff file (Weka file format[a]) at the Webb Spam Corpus' home page — http://www.cc.gatech.edu/projects/doi/WebbSpamCorpus.html. The two main parts involved in creating the Webb Spam Corpus 2011 are data collection and data cleansing. These steps are detailed below and a high-level overview of the process is provided in Fig. 1.

3.1.2. *Source URL and actual URL*

We distinguish URL links into two groups: source URLs and actual URLs. Here, source URLs are the original URLs extracted from email messages and are typically what the end user will see in the email message. Actual URLs are the final URLs or the URL of the web page that the user finally sees in their browsers. That is, this is the final URL after all redirects (HTTP redirects, Javascript redirects, meta-tag redirects, and more) have been followed. If a web page does not redirect a user, the actual URL could be the same as the source URL. To clarify this, the relationship between source URL and actual URL is shown in Fig. 2:

The relationship between source URL and actual URL has the following characteristics:

(a) One redirection chain leads from source URL to actual URL.
(b) Many source URLs may redirect/map to a single actual URL.

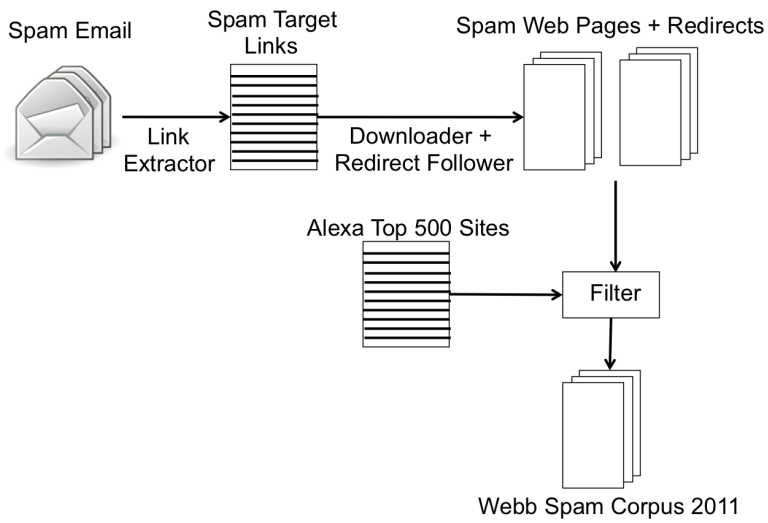

Fig. 1. Illustration of data collection and data cleansing process.

[a]Weka: http://www.cs.waikato.ac.nz/ml/weka/.

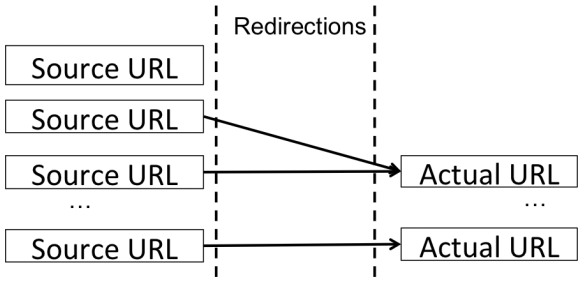

Fig. 2. The relationship between source URL and actual URL.

(c) Source URLs which were successfully accessed without resulting in a redirect is actual URL.

3.1.3. *Source URL links*

We start with a set of source URLs extracted from 6.3 million spam emails collected between May 2010 to November 2010 to a moderately sized email service provider. We only extract HTTP and HTTPS URLs (although HTTPS links make up only 0.2% of all the spam links we extracted), using Perl's URI::Find::Schemeless and Html::LinkExtor modules to extract URLs from text and HTML, respectively. We end up with 30.7 million web links (15.1 million unique links). Figure 3 shows the distribution of URL links in months. We also investigate the top level domains in source URLs and list top 10 TLDs in Table 1. "RU" is top level domain for Russian Federation and "DE" is top level domain for Federal Republic of Germany. In this study, we focus on English language web pages only, which are about 1.7 million web pages (before cleansing) which were crawled in March 2011.

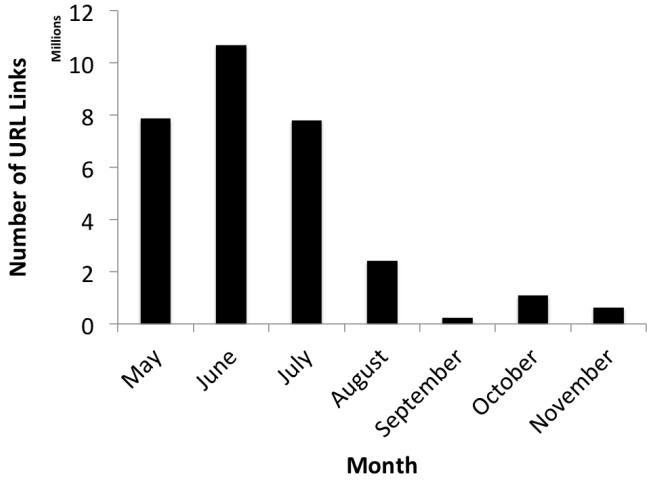

Fig. 3. Distribution of source URL links in months.

Table 1. List of top level domains in source URLs.

Top Level Domain	Number of Unique Source URLs
RU	10,052,443
COM	3,063,766
NET	205,311
UK	191,583
INFO	168,192
DE	125,472
NL	117,099
PL	106,287
IP Addresses	13,263
Other	1,061,023

3.1.4. *Web spam download*

Once we have a set of source URLs, we proceed to download all the web pages. We use a custom crawler written using Perl's LWP::Parallel::UserAgent module to download corresponding web pages. We then follow any iFrame-redirects, http-redirects, Javascript redirects (using Mozilla's Rhino), or meta-tag redirects. More details can be found in Ref. 5 which uses similar techniques. We keep the raw headers and HTML content of the page, and do not crawl or spider links from it. We downloaded a total of 1.7 million pages (including redirections) and in-total collected over 1 GB of data.

3.2. *Data cleansing*

Data cleansing on Webb Spam Corpus 2011 is split into two parts:

3.2.1. *Removing false-positives*

False-positives in corpus include legitimate URLs and error pages. Spammers often include legitimate URLs in spam emails to avoid spam rules or to appear legitimate.[6] Using Alexa's top 500 site list,[b] we list top 10 legitimate actual URLs in Webb Spam Corpus 2011 shown in Fig. 4.

Figure 4 shows that four social networks websites (www.facebook.com, support.twitter.com, www.in.com and www.myspace.com), five search engines (groups.yahoo.co.jp, explore.live.com, dogandcatanswers.yahoo.com, www.about.com and www.indeed.com) and one information portal (www.msn.com) are in the top 10 list. It indicates that spammers are using popular social networks and search engines in legitimate URL attack. We removed 6,175 legitimate actual URLs and 6,494 legitimate source URLs in this process.

Besides legitimate URL links in spam emails, the downloaded web pages also contain other false-positives. Although these actual URLs may have been spam URLs, due to the delay in setting up our downloading and cleansing system, the

[b]http://www.alexa.com/topsites.

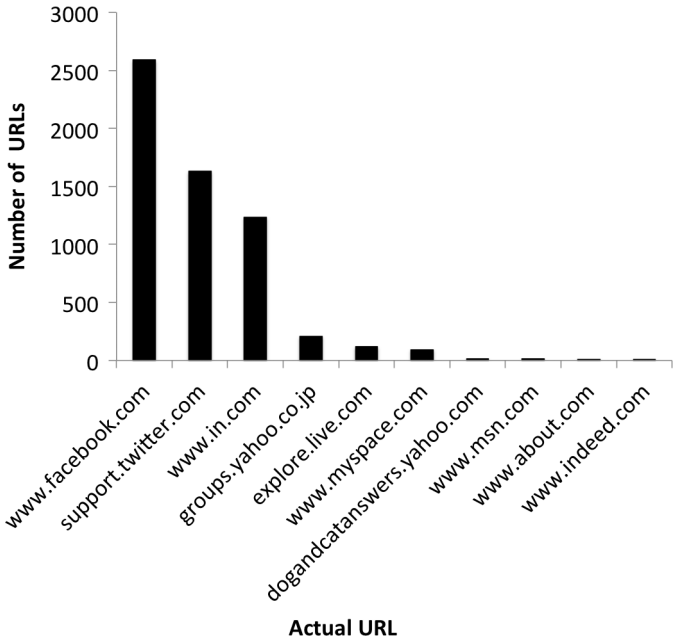

Fig. 4. Top 10 legitimate actual URLs in Webb Spam Corpus 2011.

spam URLs were crawled a few months after the source URLs were extracted. This resulted in a number of 404 HTTP errors or custom served "404 error web pages".

We eliminate such pages as well as previously mentioned false-positives leaving us with 673,489 spam web pages in the corpus.

3.2.2. *Removing non-textual web pages*

Approximately 98% of web pages identify their "Content-type" as text/html. After cleansing false-positives in corpus, we discard non-text/html pages. By removing non-textual web pages based on the attribute "Content-Type" in HTTP header information, we kept 673,313 web pages including 342,478 redirections.

3.3. *Data statistics*

After finishing downloading all web pages, we investigate the distribution of top level domains and HTTP status codes. The purpose is to find which top level domain hosts the most web spam and the most common HTTP responses when we click through those spam URL links.

To obtain popular top level domains, we process the dataset in the following steps. First, we collect all top level domains from IANA Data,[c] which contains 313 top level domains (last updated June 20, 2012). By matching all the source URLs

[c]http://data.iana.org/TLD/tlds-alpha-by-domain.txt.

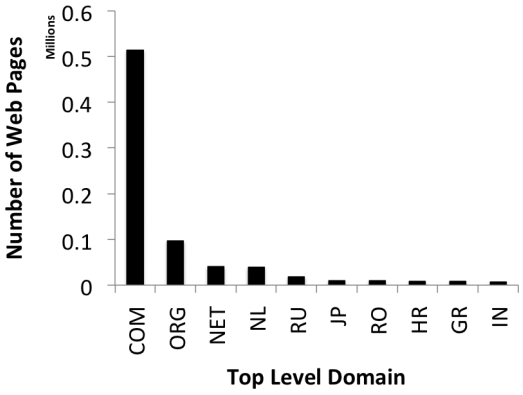

Fig. 5. Top 10 top level domains.

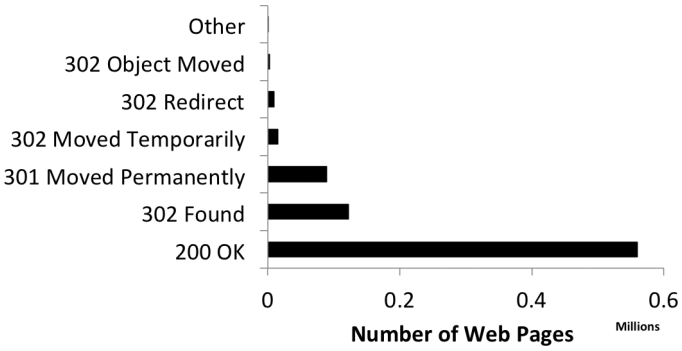

Fig. 6. Distribution of HTTP status codes.

in downloaded files with the top level domains list, we aggregated the count of web pages in the same top level domain. The 10 most popular top level domains are shown in Fig. 5. We see that the three most popular top level domains COM, ORG, and NET almost represent more than 80% of the TLDs. Especially, the percentage of web pages which are belonging to top level domain COM is over 60%.

For HTTP status codes, we aggregate all status codes based on the number of web pages and list the distribution of status codes shown in Figure 6. It shows that "200 OK" is the most common of status code in Webb Spam Corpus 2011 — over 70%. Also other status codes which are primarily used in redirection, such as "302 Found", "301 Moved Permanently", and "302 Moved Temporarily", are quite popular.

4. Comparison Between Two Datasets

We compare the Webb Spam Corpus 2011 with Webb Spam Corpus 2006 in three dimensions: redirections, HTTP session information, and content.

4.1. *Redirections*

Redirections are normally used by spammers to camouflage the actual spam URL links and avoid being blocked by URL blacklists. We look into redirections returned by source URLs in the Webb Spam Corpus 2011 shown in Table 2.

To compare fairly with redirections in the Webb Spam Corpus 2006, we compute the percentage of source URLs versus number of redirections shown in Fig. 7. It shows that Webb Spam Corpus 2011 has more source URLs returning no redirects (more source URLs which are also the actual URLs). The possible reasons are as follows: (a) spammers are using less redirections for camouflaging actual spam URLs; (b) Webb Spam Corpus 2011 has more URL links than Webb Spam Corpus 2006; (c) there may exist false positives in Webb Spam Corpus 2011 before data cleansing.

Table 2. Number of redirects returned by source URLs.

Number of Redirects	Number of Source URLs
0	254,315
1	15,075
2	2,880
3	387
4	1,361
5	86
6	58
7	46
8	31
9	27
10	26
11	19
12	13
13	15

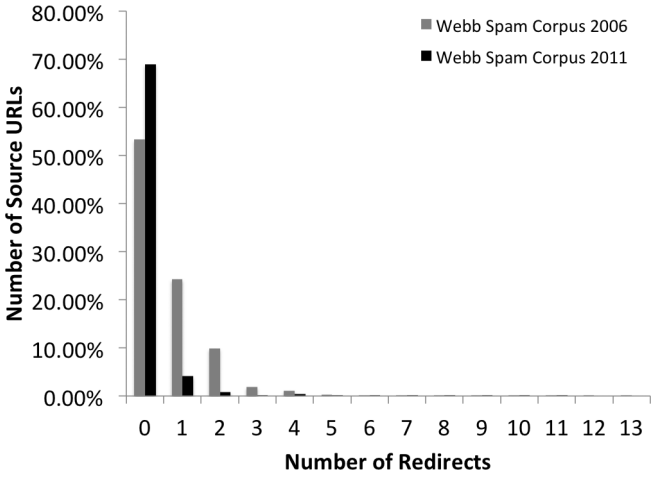

Fig. 7. Comparison based on percentage of source URLs versus number of redirections.

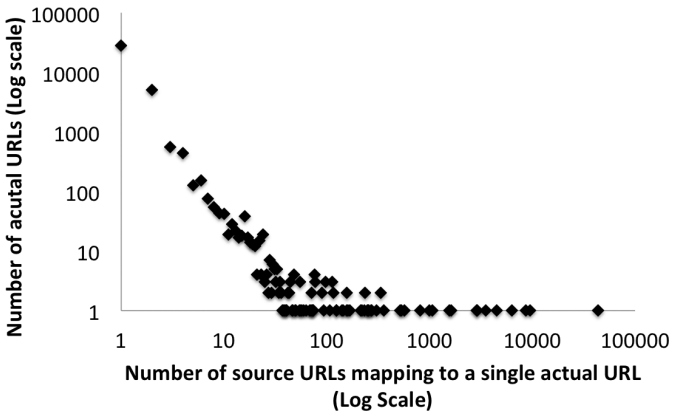

Fig. 8. Distribution of the number of source URLs that point to the same actual URL.

We also aggregate source URLs based on the actual URLs they are mapping to and generate the distribution of number of source URLs that point to the same actual URL shown in Fig. 8. It shows similar trend as the distribution of the number of source URLs that point to the same actual URL in Webb Spam Corpus 2006.

Redirections have different categories including HTTP redirect, frame redirect, iFrame redirect, meta-refresh redirect and location redirect.[6] For HTTP redirect, it also has some subcategories based on response status such as "301 Moved" HTTP redirect and "302 Found" HTTP redirect. We compare the redirection distribution of two datasets which is shown in Fig. 9.

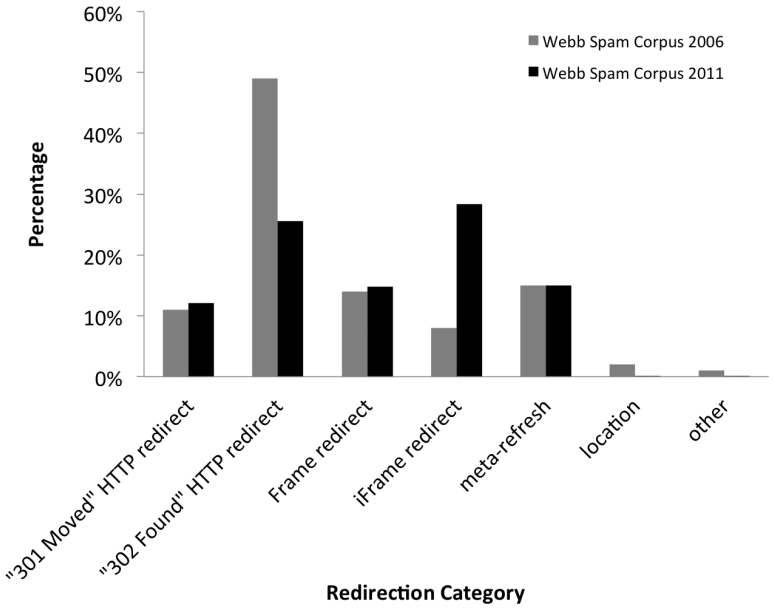

Fig. 9. Comparison between redirection distributions of the two datasets.

Figure 9 shows HTTP redirect in Webb Spam Corpus 2011 still occupies the majority of redirections, accounting for 41.7% of the redirections (25.6% for "Found" redirects, 12.1% for "Moved Permanently" redirects, and 4.0% for other HTTP redirects). HTML frame and HTML iFrame redirects account for 14.8% and 28.4%, respectively. Redirection using meta-refresh tags account for 15.0% and location redirect accounts for less than 1% of all redirects.

We observe that Webb Spam Corpus 2011 has fewer "302 Found" redirections and location redirection. But it has more iFrame redirections. Meanwhile, we found that Webb Spam Corpus 2011 has other HTTP redirects which occupies 4% redirections. The response status examples of other HTTP redirects includes: (a) "302 Object moved"; (b) "302 Moved Temporarily"; (c) "302 Redirect".

Besides showing the distribution of redirections, we also look into the common host names in redirection chains which will tell us what kinds of websites have been taken advantage of by the spammers. The most common host names in redirection chains including HTTP redirection, frame redirection, iFrame redirection, and meta-refresh redirection are shown in Table 3.

Table 3. Most common host names in redirection chains.

Host Name	Count
Top 5 host names in redirection chain	
domdex.com	59,004
www.facebook.com	37,580
domains.googlesyndication.com	9,934
bodisparking.com	9,530
potentbusy.com	9,431
Top 5 host names of HTTP redirection	
mrs45.hosteur.com	9,046
home.wanadoo.nl	8,624
arpitjain.in	6,054
sharepoint.microsoft.com	4,596
www.in.com	4,336
Top 5 host names of frame redirection	
bodisparking.com	9,530
potentbusy.com	9,430
www.ndparking.com	7,192
www.sedoparking.com	1,306
searchportal.information.com	1,209
Top 5 host names of iFrame redirection	
domdex.com	59,004
www.facebook.com	14,960
ad.doubleclick.net	2,649
areasnap.com	2,219
bullishcasino.com	1,672
Top 5 host names of meta refresh redirection	
www.facebook.com	19,931
domains.googlesyndication.com	9,875
www.lawtw.com	6,736
www2.searchresultsdirect.com	1,838
www.sedoparking.com	1,472

From Table 3, we investigated all the host names and found that there are three major categories: domain parking websites, social networks websites, and advertiser websites. For example, bodisparking.com and sedoparking.com are domain parking websites. facebook.com and in.com are social networks websites. ad.doubleclick.net is advertiser websites. The first set of counts represent the view of all of the HTTP, HTML, and JavaScript redirection techniques. This list consists of three domain parking services, one advertiser and one social networks. The top five HTTP redirect host names consist of one domain parking service, three advertisers and one social networks. The top five frame redirect host names consist of three domain parking services, two advertisers. The top five iFrame redirect host names consist of one domain parking services, three advertisers and one social networks. The top five meta refresh redirect host names consist of three domain parking services, one advertiser and one social networks.

Domain parking for idle domains is used to display advertisements and earn money. It is easy to understand that spammers are using these domains for monetary benefit. Advertisers are similar to domain parking services on displaying advertisements which may not be useful for users. For social networks websites, we studied in detail about Facebook URLs in Webb Spam Corpus 2011. We found that the majority of redirections from Facebook belongs to iFrame redirection, meta-refresh redirection and HTTP redirection. In iFrame redirection, there are three types of URL redirections based on the sub path of URL links: "connect", "plugins" and "widgets", which accounts for 72.6%, 24.4% and 3% respectively. Also the "connect" URL link redirects users to the profiles hosted Facebook. In our dataset, 10,820 "connect" URL link redirects to "t35.com" profile hosting in Facebook. "t35.com" is a domain parking services website. For 3,655 "plugins" URL links, 3,379 of them are "like" box plug-in and 140 of them are "activity" plugin. Normally, if you click on "like" box plugin, you will become a fan of events, products, or profiles so that you will be kept updated with news feeds and status changes. For "activity" plug-in, you will join the activity if you click on it. "Widgets" URL links are similar to "plugins" URL links. 444 "widgets" URL links provide "like" button for users to click. Therefore, we can conclude that spammers are using the power of social networks to spread spam information.

4.2. *HTTP session information*

Webb Spam Corpus 2011 also contains the HTTP session information that was obtained from the servers that were hosting those pages. In this section, we compare two datasets focusing on the most common server IP addresses and session header values.

4.2.1. *Hosting IP addresses*

Hosting IP address is the IP address that hosts a given web spam page. Figure 10 shows the distribution of all of the hosting IP addresses over network number in

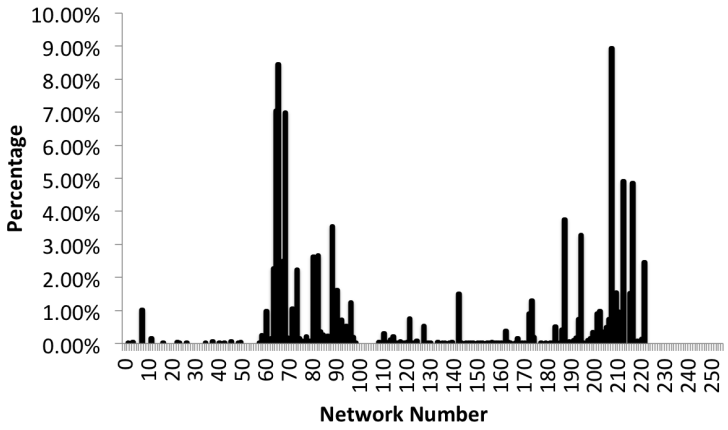

Fig. 10. Distribution of hosting IP address.

Webb Spam Corpus 2011. Here network number is the first 8 bits of IPV4 address. Previous study[6] said that the 63.* –69.* and 204.* –216.* IP address ranges account for 45.4% and 38.6% of the hosting IP addresses respectively in Webb Spam Corpus. While in Webb Spam Corpus 2011, the percentages of IP addresses in those two ranges change to 28.1% and 21.7%, respectively. Another two IP address ranges 70.*–100.* and 170.* –203.* account for 21.3% and 14.0% of the hosting IP addresses respectively in Webb Spam Corpus 2011.

It implies that spammers are comprising more various hosting IP addresses to spread web spam. The reason may be the IP blacklists used in popular anti-spam filters which force spammers to use new IP addresses for hosting web spam. To investigate most popular hosting IP addresses in Webb Spam Corpus 2011, we list top 10 hosting IP addresses based on the count of web pages. Meanwhile, through whois service, we obtain the server location and ISP (Internet service provider) for every hosting IP address.

Table 4 shows 4 IP addresses from 63.* –69.*, 2 IP addresses from 204.* –216.*, and four IP addresses from other ranges. Also, it shows that four IP addresses from US servers, two IP addresses from France servers and four IP addresses from

Table 4. Top 10 hosting IP addresses.

Hosting IP Address	Count	Server Location	ISP
208.073.210.029	23,785	Los Angeles, CA in United States	Oversee.net
065.055.011.238	21,205	Redmond, WA in United States	Microsoft Hosting
213.186.033.019	17,543	France	Ovh Systems
066.196.085.048	16,542	Sunnyvale, CA in United States	Inktomi Corporation
069.043.160.174	13,289	Beaumaris, Victoria In Australia	Castle Access
066.045.237.214	10,834	Secaucus, NJ in United States	Interserver
222.122.053.065	9,090	Seoul, Republic of Korea	Korea Telecom
217.016.006.170	9,073	France	AB Connect
195.189.117.037	8,624	Nijmegen, Gelderland in Netherlands	Bluedome Internet Application Services BV
188.040.054.131	8,538	Germany	Hetzner Online AG

other countries (Australia, Korea, Netherlands and Germany). We can see that all servers are legitimate servers which does not mean those legitimate servers are the spammers. It only means the web services provided by those servers are used by the spammers for the spamming purpose.

4.2.2. *HTTP session header*

Previous study[7] has shown that HTTP session information is used for predicting web spam efficiently. As the evolution of web spam, we intend to see whether HTTP session information of web spam has changed over time. To obtain most popular HTTP session information, we rank out top 10 HTTP session headers based on the count of web spam which those headers are associated with, shown in Table 5.

Compared with top 10 HTTP session headers, Table 5 shows some changes as follows: (a) new header P3P appears in top 10 list and old header PRAGMA has been removed from the list; (b) the most popular values for the header SERVER and CONTENT-LENGTH have changed from "microsoft-iis/6.0" to "Apache" and from 1,470 to 77, respectively; (c) the order of the header CONTENT-LENGTH moves before X-POWERED-BY but the others keep the same relative order. Also, we find that 79.1% of the web spam pages with a SERVER header were hosted by "Apache" (60.5%) or "Microsoft IIS" (18.6%). In Webb Spam Corpus, 94.2% of the web spam pages with a SERVER header were hosted by "Apache" (63.9%) or "Microsoft IIS" (30.3%). Most popular value for the header CONTENT-LENGTH is not able to show the trend of content length so we also obtain the distribution of content length shown in Fig. 11.

Figure 11 shows that the average value of content-length is between 1,000 and 10,000 although the most popular value is 77 bytes. As more multimedia used in web spam, the content length of web spam text gradually becomes shorter. Another thing we also need to check is whether the content of web spam also evolve over time.

Table 5. Top 10 HTTP session headers.

Header	Total Count	Unique Count	Most Popular Value (Count)
CONTENT-TYPE	379,721	120	text/html(147,428)
SERVER	369,985	919	Apache(82,004)
CONNECTION	359,786	5	close(312,186)
CONTENT-LENGTH	271,654	12,004	77(22,039)
X-POWERED-BY	148,944	191	ASP.NET(70,088)
CACHE-CONTROL	141,062	585	private(70,712)
SET-COOKIE	134,063	116,522	parkinglot=1;domain=.potentbusy.com; path=/;(3931)
LINK	122,352	5,012	http://l.yimg.com/d/lib/yg/css/ dynamic_200602130000.css;rel= "stylesheet"; type="text/css" (15,446)
P3P	92,591	248	policyref="http://www.dsnextgen.com/ w3c/p3p.xml"(24,180)
EXPIRES	90,915	7,668	Mon, 26 Jul 1997 05:00:00 GMT(25,641)

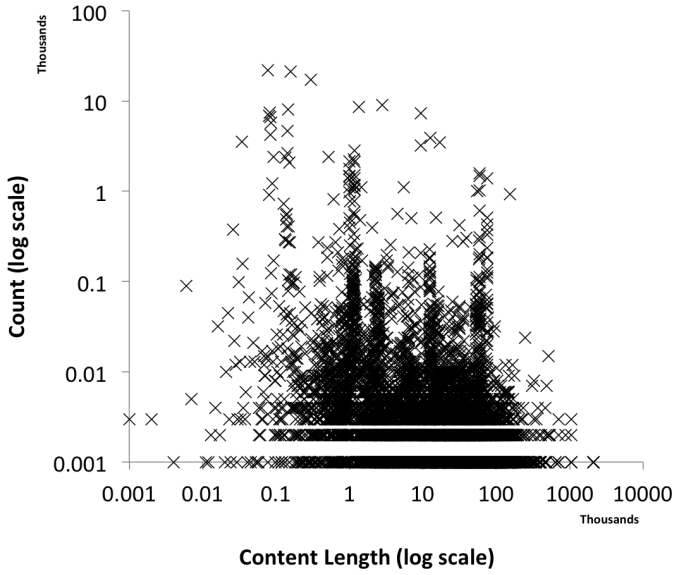

Fig. 11. Distribution of content-length.

4.3. *Content*

In this section, we compare two datasets on duplications and syntax changes between them. For duplications, we try to find the overlap between them based on MD5 hash values of content of web spam. For syntax changes, we intend to obtain the evolution of web spam syntax by comparing information gain of words and n-gram phrases.

4.3.1. *Duplications*

We compute MD5 hashes on the content of HTML web pages when we crawl the URL links. After evaluating these results, we find that there are 122,618 unique MD5 values in Webb Spam Corpus 2011. Thus, 247,367 of the web spam pages (66.9%) have the same HTML content as one of 122,618 unique web spam pages. The percentage of exact content duplicates is much higher than the percentage (42%) in Webb Spam Corpus 2006.[5] One possible reason is more URL duplications in the Webb Spam Corpus 2011.

To check the duplications between the two datasets, we iteratively compared MD5 codes of every web spam in Webb Spam Corpus 2011 and Webb Spam Corpus. The result of comparison is that 7,257 web spam in Webb Spam Corpus 2011 are overlapped with 2,834 web spam in Webb Spam Corpus 2006. The percentages of duplications between two datasets are 2.0% and 1.3% in Webb Spam Corpus 2011 and Webb Spam Corpus 2006, respectively. Therefore, there are very few exact content duplicates existing between the two datasets.

4.3.2. *Syntax analysis*

We analyze syntax of Webb Spam Corpus 2011 by computing the information gain of words in the content of web pages. Information gain, which is also called Kullback–Leibler divergence[11] in information theory, is calculated based on entropy as follows:

$$IG(T, a) = H(T) - H(T \mid a). \tag{1}$$

Here, T denotes a set of training examples and a presents the ath attribute of instance. $H(T)$ is the entropy of T and $H(T|a)$ is the conditional entropy of T with knowing the value of a.

Taking every web page as document, we adopt a bag of words model[12] to generate document instances in binary features. First, we need to tokenize the documents. Tokenization is the process of splitting the document up into words, phrases, symbols, or other meaningful elements called tokens. The features are the tokens in all documents and the value of feature is false if the token appears in the document or true if not.

For the words in web pages, we first list top 20 most popular words in Webb Spam Corpus and their appearance as a percentage of documents that contain them, shown in Fig. 12.

Figure 12 shows that some words in the top 20 list appear less than in Webb Spam Corpus 2011 such as "free", "web", "home", "search", and "software". Some words appear more frequently than in Webb Spam Corpus 2011 such as "information", "online", "internet" and "price". It indicates the trend of spammy words and changes over time.

Besides most popular words, we also look into the discriminative words which distinguish two datasets. We ranked them by the value of their information gain according to the formula and used different labels to mark the instances in two datasets. The result of top 10 words based on information gain is shown in Fig. 13.

Figure 13 shows top 10 words based on information gain. We further found that all words except "playlist" appear in Webb Spam Corpus 2006 while only four words including "playlist", "vault", "cio" and "advertisement" present in Webb Spam Corpus 2011. Since we transformed all words into lower case format, words such as "cio" and "itworld" should be "CIO" and "ITworld". Word "playlist" normally appears in multimedia section of social media. For example, user profile has the embedded radio player which has a playlist for visitors. Moreover, we compared n-gram (n is from 2 to 3) sequences in the two datasets. After using Perl's Text::Ngrams module,[d] we list top 20 n-gram (n is in the range of from 2 to 3) in two datasets based on frequency shown in Table 6.

In Table 6, $\langle N \rangle$ denotes any number sequence. Also we have removed redirections and the grams which only contain number sequences. Webb Spam Corpus 2011 has

[d]http://search.cpan.org/dist/Text-Ngrams/Ngrams.pm.

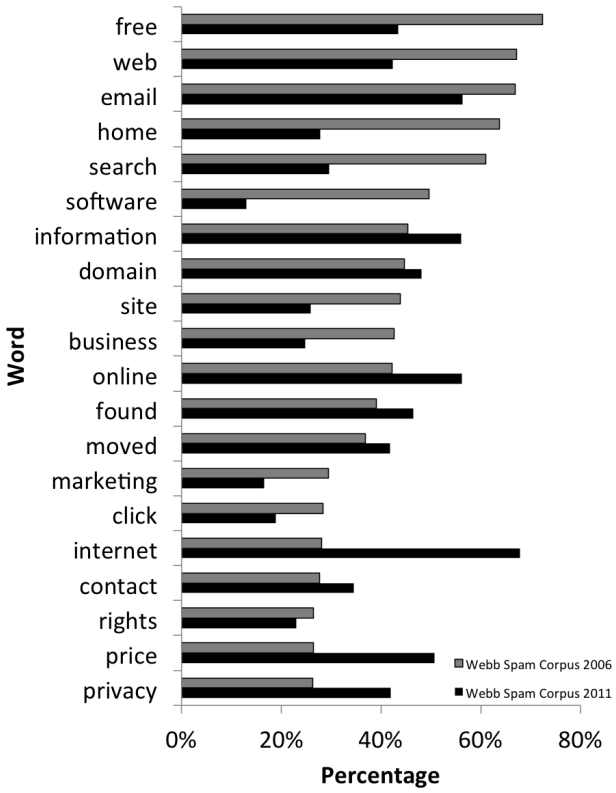

Fig. 12. Top 20 most popular words in Webb Spam Corpus (2006/2011) versus percentage of documents that contain them in two datasets.

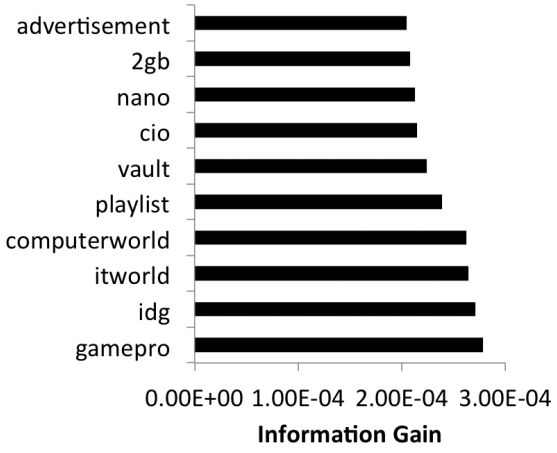

Fig. 13. Top 10 words based on information gain.

Table 6. Top 20 n-gram (n is from 2 to 3) sequences based on frequency in the two datasets (first 20 rows for Webb Spam Corpus 2006).

Two-Gram	Frequency	Three-Gram	Frequency
of the	149,029	just a few	26,585
in the	88,505	⟨N⟩ x ⟨N⟩	26,488
V ⟨N⟩	77,254	is just a	26,016
to the	77,050	the links below	25,910
on the	72,948	links below to	25,834
⟨N⟩ A	71,207	for your favorite	25,801
v ⟨N⟩	66,725	a few clicks	25,799
X ⟨N⟩	64,701	the search box	25,750
a ⟨N⟩	63,490	the Web for	25,723
⟨N⟩ x	63,019	looking for is	25,705
⟨N⟩ D	60,603	to search the	25,689
B ⟨N⟩	59,164	search the Web	25,658
x ⟨N⟩	58,455	below to search	25,636
A ⟨N⟩	57,568	few clicks away	25,633
may be	57,522	Use the search	25,632
⟨N⟩ GB	56,328	search box above	25,632
⟨N⟩ a	55,437	above or the	25,625
Price ⟨N⟩	55,424	Whatever you re	25,623
⟨N⟩ B	55,153	or hte links	25,622
⟨N⟩ s	53,330	Web for your	25,619
[4pt] of the	212,626	w ⟨N⟩ org	138,162
http www	169,180	http www w	126,770
w ⟨N⟩	140,524	www w ⟨N⟩	126,770
⟨N⟩ org	138,247	⟨N⟩ org ⟨N⟩	92,935
Price ⟨N⟩	127,091	org ⟨N⟩ ⟨N⟩	91,219
www w	126,770	mg x ⟨N⟩	73,110
in the	126,273	⟨N⟩ mg x	73,110
USD ⟨N⟩	117,108	⟨N⟩ ⟨N⟩ xmlenc	69,898
Related Searches	103,259	⟨N⟩ USD ⟨N⟩	63,904
Save ⟨N⟩	100,710	Found The doument	58,506
x ⟨N⟩	99,327	Found Found The	58,424
org ⟨N⟩	93,803	⟨N⟩ Found Found	58,424
Privacy Policy	93,328	You Save ⟨N⟩	58,127
to the	91,951	⟨N⟩ You Save	58,103
hair loss	77,774	Price ⟨N⟩ You	56,065
Internet Bellen	77,544	Admin Page Insights	54,726
⟨N⟩ mg	74,103	Retail Price ⟨N⟩	54,400
mg x	73,110	⟨N⟩ Retail Price	54,058
on the	71,891	Download Price ⟨N⟩	54,049
for the	70,177	Price ⟨N⟩ Retail	53,986

22,894,416 two-gram sequences and 14,223,621 three-gram sequences, compared with 17,049,809 two-gram sequences and 6,488,343 three-gram sequences in Webb Spam Corpus 2006. Table 6 shows that there are more numeric sequences appearing in two-gram sequences in Webb Spam Corpus 2006 than in Webb Spam Corpus 2011. three-gram sequences in Webb Spam Corpus 2006 are more related to links and search while those in Webb Spam Corpus 2011 are more related to price and money.

5. Classification Comparison

Previous research[7] shows that web page spam could be detected using HTTP header information efficiently. To further investigate the difference between two datasets, we compare them in terms of classification features and performance. Through the comparison, we try to find out whether the HTTP header information features still are discriminative and how well those classifiers perform on new dataset. To use the results in previous research as control, we adopted similar feature selection and experimental setup in our experiments.

5.1. *Feature generation and selection*

In our experiments, we adopt the traditional vector space model[13] (or "bag of words" model) to represent data in a consistent format. Also, this model has been quite effective in previous information retrieval and machine learning research. In the model, it uses a feature vector f of n features: $\langle f_1, f_2, \ldots, f_n \rangle$ to represent each data instance. Since all of our features are Boolean, we obtain that the feature is present in a given instance if $f_i = 1$; otherwise, the feature is absent. Meanwhile, we borrow three types of feature representations (phrases, n-grams, and tokens) from previous research[7] for each unique HTTP information header value. The feature generation process is: (1) we keep the header value as an uninterrupted phrase; (2) we tokenize the header value using whitespace and punctuation characters as delimiters; (3) we perform n-gram (n is from 1 to 3) generation from the tokens; (4) we prepend the header name to each of feature values. One example of feature representations is illustrated in Table 7.[7]

The feature selection process is based on a well-know information theoretic measure called information Gain.[14,15] Information Gain is defined as follows:

$$IG(f_i, c_j) = \sum_{c \in \{c_j, \bar{c_j}\}} \sum_{f \in \{f_i, \bar{f}_i\}} p(f, c) \cdot \log \frac{p(f, c)}{p(f) \cdot p(c)}, \tag{2}$$

where f_i is a feature in the feature vector, c_j is one of the classes (i.e. spam or legitimate), $p(f)$ is the probability that f occurs in the training set, $p(c)$ is the

Table 7. Feature representations.

Representation	Feature
Phrase	$server\_apache/2.0.52(fedora)$
N-grams	$server\_apache/2052$
	$server\_052fedora$
	$server\_apache/20$
	$server\_052$
	$server\_52fedora$
Tokens	$server\_apache/2$
	$server\_0$
	$server\_52$
	$server\_fedora$

probability that c occurs in the training set, and $p(f, c)$ is the joint probability that f and c occur in the training set.[7]

Information Gain quantifies the predictive power of features. If feature has a higher Information Gain value, we say that it has more predictive power. We selected and used the features which have the highest Information Gain scores to train the classifiers.

5.2. *Classifiers*

We perform the classification using the various classifiers implemented in the Weka software package.[16] Weka is an open source collection of machine learning algorithms and has become the standard tool in the machine learning community. The classifiers used in the our experiments include decision trees (e.g. C4.5, Random Forest, etc.), rule generators (e.g. RIPPER, PART, etc.), logistic regression, radial basis function (RBF) networks, HyperPipes, multilayer perceptions, K Star, SMO (an algorithm for training a support vector classifier), Simple Logistic and naïve bayes.

5.3. *Classification setup and cross validation*

We download legitimate web pages from the Standford WebBase project,[8] which categorizes its data based on crawling time. To avoid the time factor influence, we only use the web interface provided by the project to fetch web pages on March 2011 the same crawling time as Webb Spam Corpus 2011. We requested 100,000 web pages and obtained about 53,000 legitimate web pages with HTTP status code equals 200 (called WebBase 2011). In the classification, we randomly choose the same amount of legitimate web pages as spam web pages to eliminate any prior probability influence. Therefore, the training dataset used in our experiments is consisting of 53,000 legitimate web pages and 53,000 spam web pages randomly selected from Webb Spam Corpus 2011.

Based on the methodology mentioned in Sec. 5.1, we retain about 5,000 features which have the high predictive power on the basis of information gain. In addition, we employed the machine learning classifiers previously mentioned using 10-fold cross-validation model. Cross validation is a technique for protecting against overfitting in a predictive model. Specifically, the data is randomly divided into k groups and the classifier is re-estimated k times, holding back a different group each time. The overall accuracy of the classifier is the mean accuracy of the k classifiers tested.

5.4. *Classifier evaluation*

To evaluate the performance of the classifiers, we adopt the F-measure and accuracy as evaluation metrics. F-measure (also called F-score) is calculated based on precision and recall. Before introducing the details of precision and recall, we review

Table 8. The relationship between true-positive, true-negative, false-positive, and false-negative.

Actual Label	Predicted Label	
	Positive	Negative
Positive	True positive (TP)	False negative (FN)
Negative	False positive (FP)	True negative (TN)

in Table 8 the relationship between true positive, true negative, false positive and false negative.

Specifically, true positives are the number of instances that are correctly predicted as belonging to the positive class. True negatives are the number of instances that are correctly predicted as belonging to the negative class. False positives are the number of instances that are incorrectly predicted as belonging to the positive class. False negatives are the number of instances that are incorrectly predicted as belonging to the negative class. Based on these definitions, the formulas for Precision (P), Recall (R), F-measure (FM) and accuracy (A) are as follows[17]:

$$P = \frac{\mathrm{TP}}{(\mathrm{TP} + \mathrm{FP})}, \quad R = \frac{\mathrm{TP}}{(\mathrm{TP} + \mathrm{FN})}, \tag{3}$$

$$\mathrm{FM} = 2 \cdot \frac{P \cdot R}{P + R}, \quad A = \frac{(\mathrm{TP} + \mathrm{TN})}{(\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN})}. \tag{4}$$

Precision is obtained by dividing the number of the true positives by the sum of the true positives and false positives. Recall is obtained by dividing the number of the true positives by the sum of true positives and false negatives. The goal is to obtain high F-measure and accuracy values for better classification.

5.5. *Result analysis*

Previous study shows that there are five algorithms which achieved the best performance in the web spam detection using HTTP header information.[7] The algorithms include C4.5, HyperPipers, Logistic regression, RBF network, and SVM. The results on Webb Spam Corpus 2006 and WebBase 2006 are shown in Table 9.[7] For comparison, we list the performance results of those five algorithms on our new dataset in Table 10.

Table 9. Classifier performance results for Webb Spam Corpus 2006 and WebBase 2006.

Classifier	TP (%)	FP (%)	F-Measure	Accuracy (%)
C4.5	88.5	4.6	0.916	91.9
HyperPipes	88.2	0.4	0.935	93.9
Logistic Regression	88.2	2.0	0.927	93.1
RBFNetwork	87.1	0.8	0.927	93.2
SVM	89.4	2.3	0.933	93.6

Table 10. Classifier performance results for Webb Spam Corpus 2011 and WebBase 2011.

Classifier	TP (%)	FP (%)	F-Measure	Accuracy (%)
C4.5	80.1	0.0	0.890	90.0
HyperPipes	73.5	0.0	0.847	86.7
Logistic Regression	80.0	0.0	0.889	89.9
RBFNetwork	44.5	36.1	0.494	54.2
SVM	80.1	0.0	0.889	90.0

By comparing the results in two tables (Tables 9 and 10), it shows that the overall performance of the five algorithms under-performs on new dataset (Webb Spam Corpus 2011 and WebBase 2011). HyperPipes is no longer the best algorithm in terms of F-measure and accuracy. C4.5 algorithm which is one kind of decision tree algorithm outperforms others. RBFNetwork algorithm has surprising poor performance which has lowest TP rate and highest FP rate. FP rates for C4.5, HyperPipes, Logistic Regression, and SVM algorithm are nearly zero which means very few legitimate web pages have been mislabeled as spam. However, TP rates for those algorithms are all below 85%. The difference between two classification results indicates that spammers have evolved to avoid the detection based on HTTP header information over time.

To investigate the feature change over time, we rank the features based on information gain and list top 10 features in Table 11. Comparing with previous top 10 features shown in Table 12, we have the findings as follows: (1) new top 10 features are from the HTTP header fields including "P3P", "LINK", "SET-COOKIE", "CACHE-CONTROL" and "X-POWERED-BY", while the features in previous top 10 list are from the HTTP header fields including "ACCEPT-RANGES", "X-POWERED-BY", "CONTENT-TYPE", "EXPIRES", "IP address", "SERVER",

Table 11. Top 10 features for Webb Spam Corpus 2011 and WebBase 2011.

Rank	Feature	Rank	Feature
1	*p3p_cp =*	6	*p3p_xmlcp =*
2	*set-cookie_gmt*	7	*link_type =*
3	*link_rel =*	8	*p3p_ind*
4	*p3p_policyref =*	9	*cache-control_no-cache*
5	*p3p_xml*	10	*x-powered-by_php/5*

Table 12. Top 10 features for Webb Spam Corpus 2006 and WebBase 2006.

Rank	Feature	Rank	Feature
1	*accept-ranges_bytes*	6	*expires_0000gmt*
2	*x-powered-by_php/43*	7	64.225.154.135
3	*x-powered-by_php/4*	8	*server_fedora*
4	*content-type_text/html; charset = utf-8*	9	*pragma_no-cache*
5	*content-type_text/html; charset = iso-8859-1*	10	*p3p_cp =*

"PRAGMA" and "P3P"; (2) the features from the header "P3P" show high predictive power on new dataset; (3) only one feature "*p3p_cp* =" remains in top 10 list.

In those popular HTTP headers, "P3P" header is designed to give users more control of their personal information when browsing. "SET-COOKIE" header stores an HTTP cookie. "LINK" header is used to express a typed relationship with another resource, where the relation type is defined by RFC 5988.[18,19] It shows that spammers have changed their attack strategy using new and advanced techniques such as cross-site scripting (XSS)[20] and cookie spoofing, which arise up a serious protection issue to user privacy on the Internet.

5.6. *Computational costs*

Besides our evaluations on the effectiveness of HTTP session classification, we also consider the computational cost of HTTP session classification on new dataset since the timing requirements of the real-time spam detection system needs efficient classifier.

We perform timing experiments using the five classifiers mentioned in Sec. 5.5 to investigate the computational cost of HTTP session classification on Webb Spam Corpus 2011 and WebBase 2011. For each classifier, we compute the training time and per instance classification time to perform a stratified tenfold cross-validation evaluation. The experiments are conducted on a eight process (Intel Xeon 2.67 GHZ) system with 96 GB of memory, and the results for the five classifiers are shown in Table 13.

Table 13 shows that HyperPipes is still the most efficient classifier in terms of training time (10.85 s). The training times of C4.5 (727.75 s), Logistic Regression (1,688.7 s) and SVM (499.96 s) are all more than one order of magnitude larger than the corresponding training time for HyperPipes. RBFNetwork still shows efficiency in terms of training time (275.44 s) but it has poor performance in classification on new dataset.

5.7. *Discussion*

The finding of our classifications shows that the discriminative power of features in spam detection changes over time and the performance of classifiers differs between two datasets, which inspires our researchers to learn new strategies being adopted

Table 13. Classifier training and per instance classification times.

Classifier	Training Time (s)	Classify Time Per Instance (s)
C4.5	727.75	0.0068
HyperPipes	10.85	0.0001
Logistic Regression	1,688.7	0.0159
RBFNetwork	275.44	0.0026
SVM	499.96	0.0047

by spammers and possible countermeasures. Here, we listed several possible attacks from spammers and countermeasures for those attacks.

5.7.1. *Possible attacks from spammers*

One possible attack is to camouflage the links using short URLs and hidden links. Spammers take advantage of the popularity and fuzzy property of short URLs to lead vulnerable users to spam web pages. As social media are having an enormous growth, short URL spam becomes more serious problem. Also, hidden links behind the text or in the Javascript codes post real challenges to spam detection system. For example, the text for link is the URL to some websites you know well such as Facebook. However, the real destination redirected by hidden links may be a phishing website in the end.

Another possible attack is to embed the URL links in non-text medium such as image or video. It is also the reason why spam web pages show less text in our dataset. Spam URL link may appear in one or many images which is not covered by common text-based spam detection system. Moreover, spam URL links in images require the spam detection system to do image processing first that results in high cost and low accuracy as well. Those image spam also are spreading in different channels such as user profile in social media and multimedia messages in mobile phones. Spreading spam in video streaming is similar as the network bandwidth is not the bottleneck any more today.

Third one is XSS and cookie spoofing attacks. Spammers inject their malicious codes into web pages viewed by other users in cross-site scripting attack which may help spammers to bypass access controls. Browser cookie is a small piece of data storing the users' previous activity such as logging in or clicking particular buttons. After spammers obtain cookies, they could steal legitimate users' identities for spreading spam which could help them to earn clicks through the trust built among friends.

Of course, we cannot predict all possible attacks from spammers since the spamming strategies are always involving.

5.7.2. *Countermeasures*

To deal with attacks from spammers mentioned above, we may have the following countermeasures:

For camouflage attacks, we need to obtain the redirections and the destination URL through the short URL and hidden links. Meanwhile, other kinds of analysis such as behavior analysis could be used in social media. Spammers may have collective behavior pattern which differs from normal users. For example, spammers repeatedly send the same URL links to popular trending topics using a group of user accounts in Twitter.

For non-text spam attacks, we need to process those non-text media first to extract the embedded text out of them. But it becomes harder and harder to

eliminate the noise in the process. Another way to defend against those attacks is simply to disable image showing and let the users make the decisions. However, it is still an open problem to classify those non-text spam.

For XSS and cookie spoofing attacks, we need to do source code checking and prevent back-doors in our browsers. Also, if possible, we should use SSL instead of normal HTTP requests. Meanwhile, disabling JavaScript may help you prevent those attacks with the cost of not being able to run normal JavaScript programs.

To sum up, due to evolving spamming strategies and growing Internet in terms of size and complexity, we need to spend more research efforts in exploring security vulnerabilities and preventing web spam.

6. Related Work

Webb *et al.*[5] introduced the first large-scale dataset — the Webb Spam Corpus which is a collection of approximately 330,000 web spam pages. It addressed the challenge of the lack of publicly available corpora in previous web spam research.[21–25] Further, they conducted intensive experimental study of web spam through content and HTTP session analysis on it.[6] They categorized web spam into five groups: Ad Farms, Parked Domains, Advertisements, Pornography and Redirection. Besides, they performed HTTP session analysis and obtained several interesting findings. After that, Webb *et al.*[7] presented a predicative approach to web spam classification using HTTP session information (i.e. hosting IP addresses and HTTP session headers). They found that HTTP session classifier effectively detected 88.2% of the web spam pages with a low false positive rate of 0.4%. Our work is to further experimental study on evolution of web spam through content and HTTP session analysis on new Webb Spam Corpus. By comparing the two large-scale datasets in different time ranges, we obtained the trend of web spam and behavior changes of spammers. Also, we perform the classification experiments on Webb Spam Corpus 2011 and WebBase 2011 and evaluate the performance and computational cost of classifiers.

Fetterly *et al.*[26] presented their work on a large-scale study of the evolution of web pages through measuring the rate and degree of web page changes over a significant period of time. They focused on statistical analysis on the degree of change of different classes of pages. Youngjoo Chung[4] studied the evolution and emergence of web spam in three-yearly large-scale of Japanese Web archives which contains 83 million links. His work focus on the evolution of web spam based on sizes, topics and host-names of link farms, including hijacked sites which are continuously attacked by spammers and spam link generators which will generate link to spam pages in the future. Irani *et al.*[27] studied the evolution of phishing email messages and they classified phishing messages into flash attacks and non-flash attacks and analyzed transitory features and pervasive features. In our paper, we also studied the evolution of web spam but there are two important ways which are different from his work: First, we focus on redirection techniques, HTTP session information

and content not link farms. Second, the majority of the datasets we study on is in English language not in Japanese. It may have common features between them but our datasets are more representative than his dataset in terms of the popularity of web spam in English language.

In previous research, we proposed a social spam detection framework for social networks.[28,29] We studied three popular objects in social networks including profile, message, and web page objects. The classification of web page model shows promising results for associative learning. The classification results of web page model also improve the classification of other objects using the relationship between web page model and other models.

We collected new web spam corpus and studied the evolution of web spam. Our work addresses the lack of publicly available dataset for research and shows the trend of web spam in social media. In addition, we investigated the feature and performance change in web page spam classification over time.

7. Conclusions

We introduced new large-scale web spam corpus — Webb Spam Corpus 2011 which is a collection of approximately 330,000 web spam pages. Adopting the automatic web spam collection method,[5] we crawled the Internet through more than one million URL links in spam email messages during the time range between May 2010 and November 2010. In data cleansing of new dataset, we found that legitimate URL attacks by spammers are using more URLs in social media and search engine domains.

In addition to introducing new dataset, we also performed intensive study on Webb Spam Corpus 2011 through redirection, HTTP session analysis, and content. In the redirection analysis, we found that fewer redirections appear in the Webb Spam Corpus 2011 (about 70% source URLs returning no redirection). Another observation is Webb Spam Corpus 2011 has less 302 "Found" redirections and location redirection but it has more iFrame redirections. Also Webb Spam Corpus has 4% redirections which are other types of HTTP redirections. For most common host names in redirection chains, we obtained a interesting finding that social networks are used for hosting web profile spam and the widgets and plug-ins of social networks become convenient spamming traps to attract click traffic. Furthermore, we investigated the HTTP session information of Web spam in Webb Spam Corpus 2011. For hosting IP addresses, the percentages of IP addresses in ranges 63.*–69.* and 204.* –216.* have been reduced from 45.4% to 28.1% and from 38.6% to 21.7%, respectively. For HTTP session headers, new header P3P appears in top 10 list and old header PRAGMA has been removed from the list. The most popular values for the header SERVER and CONTENT-LENGTH have changed from "microsoft-iis/6.0" to "Apache" and from 1,470 to 77, respectively. Also we generated the distribution of content length of Web spam and found the content length of web spam text gradually becomes shorter. Moreover, we analyzed duplications

and syntax changes in Webb Spam Corpus 2011. 66.9% web spam pages in Webb Spam Corpus 2011 have the same HTML content as one of 122,618 unique web spam pages, which is much higher than the percentage (42%) in Webb Spam Corpus. Two datasets have very few percentage of exact content duplicates in common (2.0% for Webb Spam Corpus 2011 and 1.3% for Webb Spam Corpus). For content analysis, we listed the trend of top 20 most popular words in Webb Spam Corpus and top 10 words based on information gain to distinguish the two datasets. Also we compared n-gram (2–3) based on frequency in the two datasets.

Also, we have done classification comparison between Webb Spam Corpus 2011 and Webb Spam Corpus 2006 with respect to classifier performance, feature analysis, and computational cost three aspects. Poorer performance results of new experiments imply that spammers have evolved to avoid the detection based on HTTP header information over time. Different top 10 feature list also tells us that spammers have changed their attack strategy using new and advanced techniques. Computational cost computation shows HyperPipes classifier still has the highest efficiency but its accuracy dropped a lot in classification.

To sum up, we collected a new Webb Spam Corpus of approximately 330,000 web pages. We derive insights from this dataset as well as do an evolutionary study by intensive analysis and comparison between Webb Spam Corpus 2011 and Webb Spam Corpus 2006. Also we obtained lots of interesting findings between them.

Acknowledgments

This research has been partially funded by National Science Foundation by CNS/SAVI (1250260, 1402266), IUCRC/FRP (1127904), CISE/CNS (1138666), NetSE (0905493) programs, and gifts, grants, or contracts from DARPA/I2O, Singapore Government, Fujitsu Labs, and Georgia Tech Foundation through the John P. Imlay, Jr. Chair endowment. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation or other funding agencies and companies mentioned above.

References

1. Z. Gyöngyi and H. Garcia-Molina, Web spam taxonomy, in *Proc. 1st Int. Workshop on Adversarial Information Retrieval on the Web* (*AIRWeb*), Chiba, Japan, May 2005.
2. Z. Gyöngyi, H. Garcia-Molina and J. Pedersen, Combating web spam with trustrank, in *Proc. 30th Int. Conf. Very Large Databases* (*VLDB 04*), Toronto, Canada, August 2004.
3. D. Antoniades, I. Polakis, G. Kontaxis, E. Athanasopoulos, S. Ioannidis, E. P. Markatos and T. Karagiannis, Web: The web of short URLs, in *Proc. 20th Int. conf. World Wide Web*, WWW '11 (ACM, New York, NY, USA, 2011), pp. 715–724.
4. Y. Chung, *A Study on the Evolution and Emergence of Web Spam*, PhD thesis, University of Tokyo, Tokyo, Japan (2011).

5. S. Webb, J. Caverlee and C. Pu, Introducing the webb spam corpus: Using email spam to identify web spam automatically, in *Proc. Third Conf. Email and Anti-Spam* (*CEAS 2006*), Mountain View, CA, USA, July 2006.
6. S. Webb, J. Caverlee and C. Pu, Characterizing web spam using content and http session analysis, in *Proc. Fourth Conference on Email and Anti-Spam* (*CEAS 2007*), Mountain View, CA, USA, August 2007, pp. 84–89.
7. S. Webb, J. Caverlee and C. Pu, Predicting web spam with http session information, in *Proc. Seventeenth Conf. Information and Knowledge Management* (*CIKM 2008*), Napa Valley, CA, USA, October 2008.
8. Stanford University, The stanford webbase project, http://dbpubs.stanford.edu:8091/∼testbed/doc2/WebBase/, 2013.
9. M. Cutts, Google blog: Using data to fight webspam, http://googleblog.blogspot.com/2008/06/using-data-to-fight-webspam.html, 2013.
10. Twitter Blog, Twitter blog: Shutting down spammers, http://blog.twitter.com/2012/04/shutting-down-spammers.html, 2013.
11. S. Kullback and R. A. Leibler, On information and sufficiency, *Ann. Math. Statist.* **22**(1) (1951) 79–86.
12. D. Lewis, Naive (bayes) at forty: The independence assumption in information retrieval, in *Proc. 10th European Conf. Machine Learning* (*ECML-98*) (Springer Verlag, Heidelberg, DE, August 1998), pp. 4–15.
13. G. Salton, A. Wong and C. S. Yang, A vector space model for automatic indexing, *Commun. ACM* **18**(11) (1975) 613–620.
14. G. Forman, An extensive empirical study of feature selection metrics for text classification, *J. Mach. Learn. Res.* **3** (2003) 1289–1305.
15. Y. Yang and J. O. Pedersen, A comparative study on feature selection in text categorization, in *Proc. Fourteenth Int. Conf. Machine Learning*, ICML '97, San Francisco, CA, USA (1997), pp. 412–420.
16. M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann and I. Witten, The WEKA data mining software, *ACM SIGKDD Explorations Newslett.* **11**(1) (2009) 10–18.
17. N. Chinchor, Muc-4 evaluation metrics, in *The Fourth Message Understanding Conference* (1992), pp. 22–29.
18. Internet Engineering Task Force (IETF), RFC 5988. http://tools.ietf.org/html/rfc5988, 2013.
19. World Wide Web Consortium (W3C), http://www.w3.org/, 2013.
20. The Open Web Application Security Project (OWASP), Cross-site scripting (XSS), https://www.owasp.org/index.php/Cross-site_Scripting_(XSS), 2013.
21. E. Amitay, D. Carmel, A. Darlow, R. Lempel and A. Soffer, The connectivity sonar: Detecting site functionality by structural patterns, in *Proc. Fourteenth ACM Conf. Hypertext and Hypermedia* (ACM Press, 2003), pp. 38–47.
22. A. A. Benczur, K. Csalogany, T. Sarlos, M. Uher and M. Uher, Spamrank — fully automatic link spam detection, in *Proc. First Int. Workshop on Adversarial Information Retrieval on the Web* (*AIRWeb*) (2005).
23. K. Chandrinos, I. Androutsopoulos, G. Paliouras and C. D. Spyropoulos, Automatic web rating: Filtering obscene content on the web, in *Proc. 4th European Conf. Research and Advanced Technology for Digital Libraries*, ECDL '00, (Springer-Verlag, London, UK, 2000), pp. 403–406.
24. B. D. Davison, Recognizing nepotistic links on the web, in *AAAI-2000 Workshop on Artificial Intelligence for Web Search* (AAAI Press, 2000), pp. 23–28.

25. I. Drost and T. Scheffer, Thwarting the nigritude ultramarine: Learning to identify link spam, in *Proc. 16th European Conf. Machine Learning* (*ECML*) (2005), pp. 233–243.
26. D. Fetterly, M. Manasse, M. Najork and J. Wiener, A large-scale study of the evolution of web pages, in *Proc. 12th Int. Conf. World Wide Web*, WWW '03, New York, NY, USA (2003), pp. 669–678.
27. D. Irani, S. Webb, J. Giffin and C. Pu, Evolutionary study of phishing, *eCrime Res. Summit 2008* (2008) 1–10.
28. D. Wang, D. Irani and C. Pu, A social-spam detection framework, in *Proc. 8th Annual Collaboration, Electronic messaging, Anti-Abuse and Spam Conference* (*CEAS 11*), Perth, Australia, September 2011, pp. 46–54.
29. D. Wang, Analysis and detection of low quality information in social networks, in *Proc. Ph.D. Symp. 30th IEEE Int. Conf. Data Engineering* (*ICDE 2014*), Chicago, IL, United States (2014).