Web Spam Taxonomy

Zoltán Gyöngyi

Stanford University
Computer Science Department
Stanford, CA 94305
zoltan@cs.stanford.edu

Hector Garcia-Molina

Stanford University
Computer Science Department
Stanford, CA 94305
hector@cs.stanford.edu

March 14, 2004

Abstract

Web spamming refers to actions intended to mislead search engines and give some pages higher ranking than they deserve. Recently, the amount of web spam has increased dramatically, leading to a degradation of search results. This paper presents a comprehensive taxonomy of current spamming techniques, which we believe can help in developing appropriate countermeasures.

1 Introduction

Today, more and more people rely on the wealth of information available on the World Wide Web, and thus, increased exposure on the web may yield significant financial gains for organizations. Often, search engines are the entryways to the web. That is why some people try to mislead search engines, so that their pages rank high in search results, and thus, capture user attention.

Hence, just as with emails, we can talk about attempts of *spamming* the content of the web. The outcome is that the quality of search results decreases. For instance, for the query "kaiser pharmacy," the top 10 results returned by a major search engine (on March 12, 2004) contained 7 pages that had nothing to do with pharmacies related to the Kaiser Permanente health delivery system. The top result page actually directed the user to sites of questionable value, selling "cheap" diet drugs and "discount" male potency products. Some other results tried to lure users with pharmacy job offers, or take them to senior citizen humor pages.

To provide quality services, it is critical for search engines to address web spam. Search engines currently fight spam with a variety of often manual techniques, but as far as we know, they still lack a fully effective set of tools for combating it. We believe that the first step in combating spam is *understanding* it, that is, analyzing the techniques the spammers use to mislead search engines. A proper understanding of spamming can then guide the development of appropriate countermeasures.

To that end, in this paper we organize web spamming techniques into a taxonomy that can provide a framework for combating spam. There have been brief discussions of spam in the scientific literature [4]. One can also find details for several specific techniques on the web itself (e.g., [9]). Nevertheless, we believe that this paper offers the first comprehensive taxonomy of all important spamming techniques known to date. To build our taxonomy, we worked closely with experts at one of the major search engine companies, relying on their experience, while at the same time investigating numerous spam instances on our own.

Some readers might question the wisdom of revealing spamming secrets, concerned that this might encourage additional spamming. We assure readers that nothing in this paper is secret to the spammers; it is only most of the web users who are unfamiliar with the techniques presented here. We believe that by publicizing these spamming techniques we will encourage researchers to develop appropriate countermeasures.

2 Definition

The objective of a search engine is to provide high-quality results by correctly identifying all web pages that are *relevant* for a specific query, and presenting the user with the most *important* of those relevant pages. Relevance refers to the textual similarity between the query and a page. Pages can be given a query-specific, numeric relevance score; the higher the number, the more relevant the page is to the query. Importance refers to the global (query-independent) popularity of a page, as often inferred from the link structure (e.g., pages with many inlinks are more important), or perhaps other indicators. In practice, search engines usually combine relevance and importance, computing a combined *rank* score that is used to order query results presented to the user.

We use the term *spamming* (also, *spamdexing*) to refer to any deliberate human action that is meant to trigger an unjustifiably favorable relevance or importance for some web page, considering the page's true value. We will use the adjective *spam* to mark all those web objects (page content items or links) that are the result of some form of spamming. People who perform spamming are called *spammers.*

One can locate on the World Wide Web a handful of other definitions of web spamming. For instance, some of the definitions (see, for instance, [10]) are close to ours, stating that any modification done to a page solely because search engines exist is spamming. Specific organizations or web user groups (e.g., [7]) define spamming by enumerating some of the techniques that we present in Sections 3 and 4. An important voice in the web spam area is that of *search engine optimizers* (SEOs), such as SEO Inc. (www.seoinc.com) or Bruce Clay (www.bruceclay.com). Most SEOs claim that spamming is only increasing relevance for queries not related to the topic(s) of the page. At the same time, many SEOs endorse and practice techniques that have an impact on importance scores to achieve what they call "ethical" web page positioning or optimization. Please note that according to our definition, all types of actions intended to boost ranking, without improving the true value of a page, are considered spamming.

There are two categories of techniques associated with web spam. The first category includes the boosting techniques, i.e. methods through which one seeks to achieve high relevance and/or importance for some pages. The second category includes hiding techniques, methods that by themselves do not influence the search engine's ranking algorithms, but that are used to hide the adopted boosting techniques from the eyes of human web users. The following two sections discuss each of these two categories in more detail.

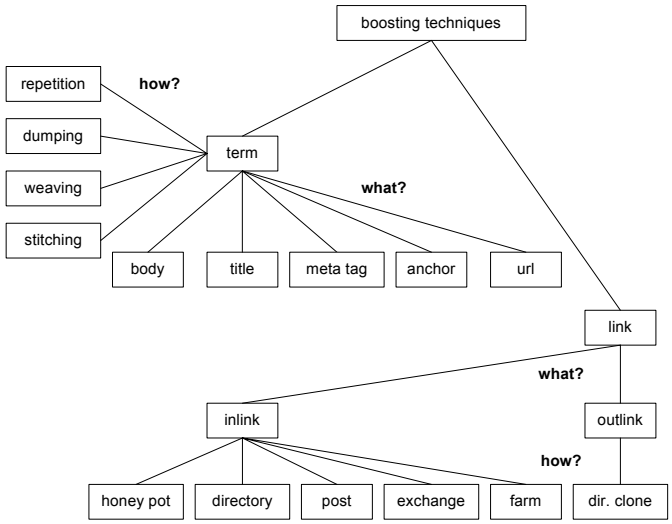

Figure 1: Boosting techniques.

3 Boosting Techniques

In this section we present spamming techniques that influence the ranking algorithms used by search engines. Figure 1 depicts our taxonomy, in order to guide our discussion.

3.1 Term Spamming

In evaluating textual relevance, search engines consider where on a web page query terms occurs. Each type of location is called a *field*. The common text fields for a page p are the document body, the title, the meta tags in the HTML header, and page p's URL. In addition, the anchor texts associated with URLs that point to p are also considered belonging to page p (anchor text field), since they often describe very well the contents of p. The terms in p's text fields are used to determine the relevance of p with respect to a specific query (a group of query terms), often with different weights given to different fields. *Term spamming* refers to techniques that tailor the contents of these text fields in order to make spam pages relevant for some queries.

3.1.1 Target Algorithms

The algorithms used by search engines to rank web pages based on their text fields use various forms of the fundamental tf-idf metric used in information retrieval [1]. Given a specific text field, for each term t that is common for the text field and a query, $tf(t)$ is the frequency of that term in the text field. For instance, if the term "apple" appears 6 times in the document body that is made up of a total of 30 terms, tf("apple") is $6/30 = 0.2$. The inverse document frequency $idf(t)$ of a term t is related to the number of documents in the collection that contain t. For instance, if "apple" appears in 4 out of the 40 documents in the collection, its idf("apple") score will be 10. The tf-idf score of a page p with respect to a query q is then computed over all common terms t:

$$tf\text{-}idf(p,q) = \sum_{t \in p, t \in q} tf(t) \cdot idf(t)$$

With *tf-idf* scores in mind, spammers can have two goals: either to make a page relevant for a large number of queries (i.e., to receive a non-zero *tf-idf* score), or to make a page very relevant for a specific query (i.e., to receive a high *tf-idf* score). The first goal can be achieved by including a large number of distinct terms in a document. The second goal can be achieved by repeating some "targeted" terms. (We can assume that spammers cannot have real control over the *idf* scores of terms. Thus, the only way to increase the *tf-idf* scores is by increasing the frequency of terms within specific text fields of a page.)

3.1.2 Techniques

Term spamming techniques can be grouped based on the text field in which the spamming occurs. Therefore, we distinguish:

- *Body spam.* In this case, the spam terms are included in the document body. This spamming technique is among the simplest and most popular ones, and it is almost as old as search engines themselves.

- *Title spam.* Today's search engines usually give a higher weight to terms that appear in the title of a document. Hence, it makes sense to include the spam terms in the document title.

- *Meta tag spam.* The HTML meta tags that appear in the document header have always been the target of spamming. Because of the heavy spamming, search engines currently give low priority to these tags, or even ignore them completely. Here is a simple example of a spammed keywords meta tag:

  <meta name="keywords" content="buy, cheap, cameras, lens, accessories, nikon, canon">

- *Anchor text spam.* Just as with the document title, search engines assign higher weight to anchor text terms, as they are supposed to offer a summary of the pointed document. Therefore, spam terms are sometimes included in the anchor text of the HTML hyperlinks to a page. Please note that this spamming technique is different from the previous ones, in the sense that the spam terms are added not to a target page itself, but the other pages that point to the target. As anchor text gets indexed for both pages, spamming it has impact on the ranking of both the source and target pages. A simple anchor text spam is:

  free, great deals, cheap, inexpensive, cheap, free

- *URL spam.* Some search engines also break down the URL of a page into a set of terms that are used to determine the relevance of the page. To exploit this, spammers sometimes create long URLs that include sequences of spam terms. For instance, one could encounter spam URLs like:

  buy-canon-rebel-300d-lens-case.camerasx.com,
  buy-nikon-d100-d70-lens-case.camerasx.com,
  ...

Often, spamming techniques are combined. For instance, anchor text and URL spam is often encountered together with link spam, which will be discussed in Section 3.2.2.

Another way of grouping term spamming techniques is based on the type of terms that are added to the text fields. Correspondingly, we have:

- *Repetition* of one or a few specific terms. This way, spammers achieve an increased relevance for a document with respect to a small number of query terms.
- *Dumping* of a large number of unrelated terms, often even entire dictionaries. This way, spammers make a certain page relevant to many different queries. Dumping is effective against queries that include relatively rare, obscure terms: for such queries, it is probable that only a couple of pages are relevant, so even a spam page with a low relevance score would appear among the top results.
- *Weaving* of spam terms into copied contents. Sometimes spammers duplicate text corpora (e.g., news articles) available on the web and insert spam terms into them at random positions. This technique is effective if the topic of the original real text was so rare that only a small number of relevant pages exist. Weaving is also used for *dilution*, i.e., to conceal some repeated spam terms within the text, so that search engine algorithms that filters out plain repetition would be misled. A short example of spam weaving is:

  Remember not only airfare to say the right plane tickets thing in the right place, but far cheap travel more difficult still, to leave hotel rooms unsaid the wrong thing at vacation the tempting moment.

- *Phrase stitching* is also used by spammers to create content quickly. The idea is to glue together sentences or phrases, possibly from different sources; the spam page might then show up for queries on any of the topics of the original sentences. For instance, a spammer using this paper as source could come up with the following collage:

  The objective of a search engine is to provide high-quality results by correctly identifying. Unjustifiably favorable boosting techniques, i.e., methods through which one seeks relies on the identification of some common features of spam pages.

3.2 Link Spamming

Beside term-based relevance metrics, search engines also rely on link information to determine the importance of web pages. Therefore, spammers often create link structures that they hope would increase the importance of one or more of their pages.

3.2.1 Target Algorithms

For our discussion of the algorithms targeted by link spam, we will adopt the following model. For a spammer, there are three types of pages on the web:

1. *Inaccessible* pages are those that a spammer cannot modify. These are the pages out of reach; the spammer cannot influence their outgoing links. (Note that a spammer can still point to inaccessible pages.)
2. *Accessible* pages are maintained by others (presumably not affiliated with the spammer), but can still be modified in a limited way by a spammer. For example, a spammer may be able to add a message to a guest book, and that message may contain a link to a

spam site. As infiltrating accessible pages is usually not straightforward, let us say that a spammer has a limited budget of A accessible pages. For simplicity, we assume that at most one outgoing link can be added to each accessible page.

3. *Own* pages are maintained by the spammer, who thus has full control over their contents. We call the own pages a *spam farm*. A spammer's goal is to boost the importance of one or more of his or her own pages. For simplicity, say there is a single target page t. There is a certain maintenance cost (domain registration, web hosting) associated with a spammer's own pages, so we can assume that a spammer has a limited budget of O such pages, not including the target page.

With this model in mind, we discuss the two well-known algorithms used to compute importance scores based on link information.

HITS. The original HITS algorithm was introduced in [5] to rank pages on a specific topic. It is more common, however, to use the algorithm on all pages on the web to assigns global *hub* and *authority* scores to each page. According to the circular definition of HITS, important hub pages are those that *point to* many important authority pages, while important authority pages are those *pointed to* by many hubs. A search engine that uses the HITS algorithm to rank pages returns as query result a blending of the pages with the highest hub and authority scores.

Hub scores can be easily spammed by adding outgoing links to a large number of well known, reputable pages, such as www.cnn.com or www.mit.edu. Thus, a spammer should add many outgoing links to the target page t to increase its hub score.

Obtaining a high authority score is more complicated, as it implies having many incoming links from presumably important hubs. A spammer could boost the hub scores of his O pages (once again, by adding many outgoing links to them) and then make those pages point to the target. Links from important accessible hubs could increase the target's authority score even further. Therefore, the rule here is "the more the better": within the limitations of the budget, the spammer should have all own and accessible pages point to the target. Non-target own pages should also point to as many other (known important) authorities as possible.

PageRank. PageRank, as described in [8], uses incoming link information to assign global importance scores to all pages on the web. It assumes that the number of incoming links to a page is related to that page's popularity among average web users (people would point to pages that they find important). The intuition behind the algorithm is that a web page is important if several other important web pages point to it. Correspondingly, PageRank is based on a mutual reinforcement between pages: the importance of a certain page *influences* and is *being influenced* by the importance of some other pages.

Recent analyses of the algorithm [2, 6] showed that the total PageRank score r_{total} of a group of pages (at the extreme, a single page) depends on four factors:

$$r_{total} = r_{static} + r_{in} - r_{out} - r_{sink},$$

where r_{static} is the score gained from the static score distribution (random jump); r_{in} is the score flowing into the pages through the incoming links from external pages; r_{out} is the score leaving the pages through their outgoing links to external pages; and r_{sink} is the score loss due to sink pages within the group (i.e., pages without outgoing links).

The previous formula leads us to a class of optimal link structures for our model that maximize the score of the target page. One such optimal structure is presented in Figure 2; it has the nice properties that (1) it makes all own pages reachable from the accessible ones (so

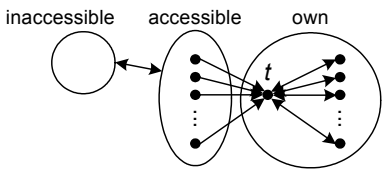

Figure 2: Optimal link structure for PageRank.

that they could be crawled by a search engine), and (2) contains a minimal number of links. For this structure, we used the following strategies to maximize the total PageRank score of the spam farm, and of page t in particular:

1. Use all available O pages in the spam farm, thus maximizing the static score r_{static}.
2. Accumulate the maximum number of A incoming links from accessible pages to the spam farm, thus maximizing the incoming score r_{in}.
3. Suppress links pointing outside the spam farm, thus setting r_{out} to zero.
4. Avoid sink pages within the farm, assuring that every page (including t) has some outlinks. This sets r_{sink} to zero.

Within the spam farm, the link structure maximizes the score of page t by abiding the following rules:

1. Make all accessible and own pages point directly to the target, thus maximizing its incoming score.
2. Add links from t to all other own pages. Without such links, t would had lost a significant part of its score being a sink, and the own pages would had been unreachable from outside the spam farm. The resulting short cycles help the score leaving t "flow back" into it. Note that it would not be wise to create similar cycles between t and the accessible pages, as those would decrease the total score of the spam farm.

As we can see in Figure 2, the "more is better" rule also applies to PageRank. Setting up sophisticated link structures within a spam farm does not improve the ranking of the target page. A spammer can achieve high PageRank by accumulating many incoming links from accessible pages, and/or by creating large spam farms with all the pages pointing to the target. The corresponding spamming techniques are presented next.

3.2.2 Techniques

We group link spamming techniques based on whether they add numerous outgoing links to popular pages or they gather many incoming links to a single target page or group of pages.

Outgoing links. A spammer might manually add a number of outgoing links to well-known pages, hoping to increase the page's hub score. At the same time, the most widespread method for creating a massive number of outgoing links is *directory cloning:* One can find on the World Wide Web a number of directory sites, some larger and better known (e.g., the DMOZ Open Directory, dmoz.org, or the Yahoo! directory, dir.yahoo.com), some others smaller and less famous (e.g., the Librarian's Index to the Internet, lii.org). These directories

organize web content around topics and subtopics, and list relevant sites for each. Spammers then often simply replicate some or all of the pages of a directory, and thus create massive outlink structures quickly.

Incoming links. In order to accumulate a number of incoming links to a single target page or set of pages, a spammer might adopt some of the following strategies:

- *Create a honey pot*, a set of pages that provide some useful resource (e.g., copies of some Unix documentation pages), but that also have (hidden) links to the target spam page(s). The honey pot then attracts people to point to it, boosting the ranking of the target page(s). Please note that the previously mentioned directory clones could act as honey pots.

- *Infiltrate a web directory.* Several web directories allow webmasters to post links to their sites under some topic in the directory. It might happen that the editors of such directories do not verify and control link additions strictly, or get misled by a skilled spammer. In these instances, spammers may be able to add to directory pages links that point to their target pages. As directories tend to have both high PageRank and hub scores, this spamming technique is useful in boosting both the PageRank and authority scores of target pages.

- *Post links to unmoderated message boards or guest books.* As mentioned earlier, spammers may include URLs to their spam pages as part of the seemingly innocent messages they post. Without a moderator to oversee the submitted messages, pages of the message board or guest book end up linking to spam.

- *Participate in link exchange.* Often times, a group of spammers set up a link exchange structure, so that their sites point to each other.

- *Create own spam farm.* These days spammers can control a large number of sites and create arbitrary link structures that would boost the ranking of some target pages. While this approach was prohibitively expensive a few years ago, today it is very common as the costs of domain registration and web hosting have declined dramatically.

4 Hiding Techniques

It is usual for spammers to conceal the telltale signs (e.g., repeated terms, long lists of links) of their activities. They use a number of techniques to hide their abuse from regular web users visiting spam pages, or from the editors at search engine companies who try to identify spam instances. This section offers an overview of the most common spam hiding techniques, also summarized in Figure 3.

4.1 Content Hiding

Spam terms or links on a page can be made invisible when the browser renders the page. One common technique is using appropriate *color schemes*: terms in the body of an HTML document are not visible if they are displayed in the same color as the background. We show a simple example next:

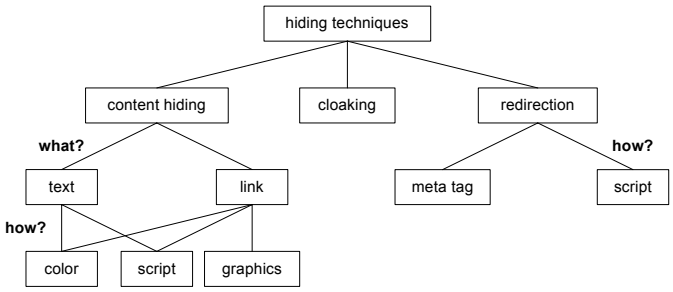

Figure 3: Spam hiding techniques.

```
<body background=white>
  <font color=white>hidden text</font>

    ...
</body>
```

In a similar fashion, spam links can be hidden by avoiding anchor text. Instead, spammers often create tiny, 1×1-pixel *anchor images* that are either transparent or background-colored:

```
<a href="target.html"><img src="tinyimage.gif"></a>
```

A spammer can also use *scripts* to hide some of the visual elements on the page, for instance, by setting the visible HTML style attribute to false.

4.2 Cloaking

If spammers can clearly identify web crawler clients, they can adopt the following strategy, called *cloaking*: given a URL, spam web servers return one specific HTML document to a regular web browser, while they return a different document to a web crawler. This way, spammers can present the ultimately intended content to the web users (without traces of spam on the page), and, at the same time, send a spammed document to the search engine for indexing.

The identification of web crawlers can be done in two ways. On one hand, some spammers maintain a list of IP addresses used by search engines, and identify web crawlers based on their matching IPs. On the other hand, a web server can identify the application requesting a document based on the user-agent field in the HTTP request message. For instance, in the following simple HTTP request message the user-agent name is that one used by the Microsoft Internet Explorer 6 browser:

```
GET /db_pages/members.html HTTP/1.0
Host: www-db.stanford.edu
User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)
```

The user-agent names are not strictly standardized, and it is really up to the requesting application what to include in the corresponding message field. Nevertheless, search engine crawlers usually identify themselves by a name distinct from the ones used by traditional web browser applications, in order to allow well-intended, legitimate optimizations. For instance, few sites serve to search engines a version of their pages that is free from navigational links,

advertisements, and other visual elements related to the presentation, but not to the content. This kind of activity is welcome by the search engines, as it helps indexing the useful information.

4.3 Redirection

Another way of hiding the spam content on a page is by automatically *redirecting* the browser to another URL as soon as the page is loaded. This way the page still gets indexed by the search engine, but the user will not ever see it—pages with redirection act as *intermediates* (or *proxies*, *doorways*) for the ultimate targets, which spammers try to serve to a user reaching their sites through search engines.

Redirection can be achieved in a number of ways. A simple approach is to take advantage of the refresh meta tag in the header of an HTML document. By setting the refresh time to zero and the refresh URL to the target page, spammers can achieve redirection as soon as the page gets loaded into the browser:

```
<meta http-equiv="refresh" content="0;url=target.html">
```

While the previous approach is not hard to implement, search engines can easily identify such redirection attempts by parsing the meta tags. More sophisticated spammers achieve redirection as part of some script on the page, as scripts are not executed by the crawlers:

```
<script language="javascript"><!- -
  location.replace("target.html")
- -></script>
```

5 Conclusions

In this paper we presented a variety of commonly used web spamming techniques, and organized them into a taxonomy. We argue that such a structured discussion of the subject is important to raise the awareness of the research community. Our spam taxonomy naturally leads to a similar taxonomy of countermeasures. Correspondingly, we outline next the two approaches that a search engine can adopt in combating spam.

On one hand, it is possible to address each of the boosting and hiding technique presented in Sections 3 and 4 separately. Accordingly, one could:

1. *Identify* instances of spam, i.e., find pages that contain specific types of spam, and stop crawling and/or indexing such pages. Search engines usually take advantage of a group of automatic or semi-automatic, proprietary spam detection algorithms and the expertise of human editors to pinpoint and remove spam pages from their indices.

2. *Prevent* spamming, that is, making specific spamming techniques impossible to use. For instance, a search engine's crawler could identify itself as a regular web browser application in order to avoid cloaking.

3. *Counterbalance* the effect of spamming. Today's search engines use variations of the fundamental ranking methods (discussed in Sections 3.1.1 and 3.2.1) that feature some degree of spam resilience.

On the other hand, it is also possible to address the problem of spamming as a whole, despite the differences among individual spamming techniques. This approach relies on the identification of some common features of spam pages. For instance, the spam detection methods presented in [3] take advantage of the *approximate isolation* of reputable, non-spam pages: reputable web pages seldom point to spam. Thus, adequate link analysis algorithms can be used to separate reputable pages from any form of spam, without dealing with each spamming technique individually.

Acknowledgement

This paper is the result of many interesting discussions with one of our collaborators at a major search engine company, who wishes to remain anonymous. We would like to thank this person for the explanations and examples that helped us shape the presented taxonomy of web spam.

References

[1] Ricardo A. Baeza-Yates and Berthier A. Ribeiro-Neto. *Modern Information Retrieval*. ACM Press / Addison-Wesley, 1999.

[2] Monica Bianchini, Marco Gori, and Franco Scarselli. Inside PageRank. Technical report, University of Siena, 2003.

[3] Zoltán Gyöngyi, Hector Garcia-Molina, and Jan Pedersen. Combating web spam with TrustRank. Technical report, Stanford University, 2004.

[4] Monika R. Henzinger, Rajeev Motwani, and Craig Silverstein. Challenges in web search engines. *SIGIR Forum*, 36(2):11–22, 2002.

[5] Jon M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5):604–632, 1999.

[6] Amy Langville and Carl Meyer. Deeper inside PageRank. Technical report, North Carolina State University, 2003.

[7] Open Directory Project. Open directory editorial guidelines: Spamming. http://dmoz.org/guidelines/spamming.html.

[8] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The PageRank citation ranking: Bringing order to the web. Technical report, Stanford University, 1998.

[9] Alan Perkins. The classification of search engine spam. http://www.ebrandmanagement.com/whitepapers/spam-classification/.

[10] Shari Thurow. The search engine spam police, 2002. http://www.searchenginewatch.com/searchday/article.php/2159061.