Spam-Resilient Web Rankings via Influence Throttling

James Caverlee, Steve Webb, and Ling Liu
Georgia Institute of Technology, College of Computing
Atlanta, GA 30332 USA
{caverlee, webb, lingliu}@cc.gatech.edu

Abstract

Web search is one of the most critical applications for managing the massive amount of distributed Web content. Due to the overwhelming reliance on Web search, there is a rise in efforts to manipulate (or spam) Web search engines. In this paper, we develop a spam-resilient ranking model that promotes a source-based view of the Web. One of the most salient features of our spam-resilient ranking algorithm is the concept of influence throttling. We show how to utilize influence throttling to counter Web spam that aims at manipulating link-based ranking systems, especially PageRank-like systems. Through formal analysis and experimental evaluation, we show the effectiveness and robustness of our spam-resilient ranking model in comparison with existing Web algorithms such as PageRank.

1. Introduction

The Web is arguably the most massive and successful distributed computing application today. Millions of Web servers support the autonomous sharing of billions of Web pages. Web search provides some of the most critical functionality for distributing, sharing, and managing the growing amount of content. As Web search becomes more and more popular for being the first and last resort of information, more and more incidents of Web spam are observed, experienced, and reported [1, 17, 21, 26].

The rise in efforts to manipulate (or spam) how users view and interact with the Web degrades the quality of information on the Web and places the user at great risk for malicious exploitation by a Web spammer. For example, Web spammers often construct illegitimate copies of legitimate Web sites (like eBay) to support identity theft. Spammer-controlled Web sites often host various types of malware – be it adware, spyware, or keyloggers – on sites intended to have a "look-and-feel" that is similar to legitimate sites [29]. Recent studies suggest that 8% of pages [17] and 18% of sites [22] are specifically engineered to manipulate the underlying algorithms that drive Web search engines.

In this paper, we focus on three prominent types of link-based vulnerabilities we have identified in Web ranking systems: *hijacking*, *honeypots*, and *collusion*. Each of these link-based vulnerabilities subverts the credibility of traditional link-based ranking approaches and undermines the quality of information offered through ranking systems. For example, in January 2006, a reputable computer science department's web page for new PhD students was *hijacked* by a Web spammer, and over 50 links to pornography-related Web sites were added to the page. This type of link-based vulnerability corrupts link-based ranking algorithms like HITS [24] and PageRank [28] by making it appear that a reputable page is endorsing the Web spam target pages.

To defend against these important types of link-based vulnerabilities, we introduce a new ranking model that promotes a source-level view of the Web and a novel notion of influence throttling for countering the influence of prominent attacks used by spammers to manipulate link-based ranking systems. We present the *Spam-Resilient Source-Rank* algorithm for assessing the quality of Web sources through a random walk over Web sources. Analytically, we provide a formal discussion on the effectiveness of the ranking model against link-based vulnerabilities. We show how it provides strong resistance to manipulation and raises the cost of rank manipulation to a Web spammer. Experimentally, we study the spam resilience of the ranking model over three large real-world datasets. We show how Spam-Resilient SourceRank counters the vulnerabilities inherent in PageRank, making it harder for adversaries to abuse.

2. Background

Link-based ranking approaches like HITS [24] and PageRank [28] are core algorithms used by search engines to assess the relative importance of Web pages by analyzing the inherent hyperlink structure of the Web. In the rest of this section, we discuss the Web graph model, briefly illustrate

This work is partially supported by grants from NSF CSR, NSF ITR, NSF CyberTrust, AFOSR, an IBM Faculty Award, and an IBM SUR grant.

1-4244-0910-1/07/$20.00 ©2007 IEEE.

link-based ranking through the popular PageRank approach, and identify several link-based vulnerabilities.

Web Graph Model: Link-based algorithms consider the Web as a graph $\mathcal{G}_\mathcal{P} = \langle \mathcal{P}, \mathcal{L}_\mathcal{P} \rangle$, where the vertexes in $\mathcal{P}$ correspond to Web pages and the directed edges in $\mathcal{L}_\mathcal{P}$ correspond to hyperlinks between pages. A hyperlink from page p_i to page p_j is denoted as the directed edge $(p_i, p_j) \in \mathcal{L}_\mathcal{P}$, where $p_i, p_j \in \mathcal{P}$. We denote the number of hyperlinks pointing out from page p_i as $o(p_i)$. $\mathcal{G}_\mathcal{P}$ can be represented by a $|\mathcal{P}| \times |\mathcal{P}|$ transition matrix $\mathbf{M}$ where a non-zero ij^{th} entry indicates a link from page p_i to page p_j:

$$M_{ij} = \begin{cases} \dfrac{1}{o(p_i)} & \text{if } (p_i, p_j) \in \mathcal{L}_\mathcal{P} \\ 0 & \text{otherwise} \end{cases}$$

PageRank: PageRank provides a global authority score to each page on the Web based on the linkage structure of the entire Web. PageRank assesses the importance of a page by recursively considering the authority of the pages that point to it via hyperlinks. For n Web pages we can denote the PageRank authority scores as the vector $\boldsymbol{\pi} = (\pi_1, \pi_2, ..., \pi_n)$. The PageRank calculation considers the page-level transition matrix $\mathbf{M}$ as well as an n-length static score vector $\mathbf{e}$, which is typically taken to be the uniform vector $\mathbf{e} = \left(\frac{1}{n}, \frac{1}{n}, ..., \frac{1}{n}\right)$. We can write the PageRank equation as a combination of these two factors according to a mixing parameter α:

$$\boldsymbol{\pi} = \alpha \mathbf{M}^T \boldsymbol{\pi} + (1 - \alpha)\mathbf{e} \qquad (1)$$

which can be solved using a stationary iterative method like Jacobi iterations [18].

Link-Based Vulnerabilities: We illustrate three categories of vulnerabilities in link-based ranking algorithms. These vulnerabilities can be exploited by spammers to subvert link-based ranking algorithms like PageRank, which typically rely on a fundamental assumption that a link from one page to another is an authentic conferral of authority by the pointing page to the target page. The three types are:

1. Hijacking, whereby spammers insert links into legitimate pages that point to a spammer-controlled page. There are a number of avenues for hijacking legitimate pages, including the insertion of spam-links into public message boards, openly editable wikis, and legitimate weblogs.

2. Honeypots, whereby spammers create quality sites to collect legitimate links that are then passed on to spammer-controlled pages. Rather than risking exposure by hijacking a link, a *honeypot* induces links, so that it can pass along its accumulated authority by linking to a spam target page.

3. Collusion, whereby a spammer constructs specialized linking structures across one or more spammer-controlled pages. In a *link exchange*, multiple spammers trade links to pool their collective resources for mutual page promotion. Another example is a *link farm*, in which a Web spammer generates a large number of colluding pages that point to a single target page.

In practice, Web spammers rely on combinations of these basic strategies to create more complex attacks on link-based ranking systems. This complexity can make the total attack both more effective (since multiple attack vectors are combined) and more difficult to detect (since simple pattern-based arrangements are masked).

3. Spam-Resilient SourceRank

To counter the strong influence of link-based manipulation, we introduce a spam-resilient ranking model called Spam-Resilient SourceRank. In the following sections we present the detailed derivation of the proposed ranking model, including three key components: (1) the source view of the Web; (2) source-based influence flow; and (3) influence throttling. In concert, these three components provide strong resistance to manipulation and significantly raise the costs of link-based manipulation.

3.1. Source View of the Web

The first component of Spam-Resilient SourceRank is a hierarchical source view of the Web that departs from PageRank's fundamentally flat view of the Web, in which all pages are treated as equal nodes in a Web graph. In this complementary view, pages are grouped into logical collections of Web pages that we call *sources*. The source view is motivated by recent work that has studied the link-locality structure of the Web (e.g., [7, 13, 14, 23]) to organize Web pages into logical groups of pages based on the strong tendency of pages within a source to link to other pages within the same source. Intuitively, a source could be defined using the host or domain information associated with each Web page, or it could be augmented with expert knowledge [11]. We use the term *source edge* to refer to the notion of source-based citation. A source s_1 has a source edge to another source s_2 if one page in s_1 has a hyperlink to a page in s_2.

Just as the page graph $\mathcal{G}_\mathcal{P} = \langle \mathcal{P}, \mathcal{L}_\mathcal{P} \rangle$ models the Web as a directed graph where the nodes correspond to Web pages $\mathcal{P}$ and the set of directed edges $\mathcal{L}_\mathcal{P}$ correspond to hyperlinks between pages, the *source graph* $\mathcal{G}_\mathcal{S} = \langle \mathcal{S}, \mathcal{L}_\mathcal{S} \rangle$ is a directed graph where the nodes of the graph correspond to Web sources in $\mathcal{S}$ and the set of directed edges $\mathcal{L}_\mathcal{S}$ corresponds to source edges. We can represent the source graph $\mathcal{G}_\mathcal{S}$ with an $|\mathcal{S}| \times |\mathcal{S}|$ transition matrix $\mathbf{T}$ where the ij^{th} entry indicates the edge strength for an edge from source s_i to source s_j:

$$T_{ij} = \begin{cases} \dfrac{1}{o(s_i)} & \text{if } (s_i, s_j) \in \mathcal{L}_\mathcal{S} \\ 0 & \text{otherwise} \end{cases}$$

where we denote the number of edges pointing out from source s_i as $o(s_i)$, meaning that this initial transition matrix relies on uniform transition probabilities.

It would be natural to determine the rank of each source using a PageRank-style iterative calculation over the Web source transition matrix, much like in Equation 1 (and suggested in [4, 16]). In such a ranking model, the source view provides a first step towards mitigating the influence of a Web spammer. In the ideal scenario, all of the pages under the control of a Web spammer would be mapped to a single source (and all legitimate pages would be mapped to their appropriate source, as well), meaning that collusion among Web spammers could be muted entirely by discounting the links within each source. If spam source i is assigned a transition probability $T_{ii} = 0$, then all link exchanges and link farms within the source will have no influence over the rank of the source. In practice, spammers can never be perfectly identified, and they can still rely on hijacking and honeypots to collect links from legitimate pages. Hence, we next introduce the second layer of defense.

3.2. Source-Based Influence Flow

The second component of Spam-Resilient SourceRank is *source-based influence flow* for determining the edge strength from one source to another. Given the directed source graph $\mathcal{G}_{\mathcal{S}} = \langle \mathcal{S}, \mathcal{L}_{\mathcal{S}} \rangle$, let $w(s_i, s_j)$ denote the weight assigned to the source edge $(s_i, s_j) \in \mathcal{L}_{\mathcal{S}}$. Unlike the straightforward notion of linkage in the page graph, source edges are derived from the page edges in the underlying page graph. Rather than a simple uniform edge strength for each outgoing edge, we propose a *source consensus* edge weighting that counts the number of unique pages within an originating source that point to a target source:

$$w(s_i, s_j) = \sum_{p_i | s(p_i) = s_i} \left(\bigvee_{p_j | s(p_j) = s_j} \mathcal{I}\left[(p_i, p_j) \in \mathcal{L}_{\mathcal{P}}\right] \right)$$

Since, source-based influence flow is a scalar value in the range $[0, 1]$, where the outgoing edge weights for any source sum to 1, we require an additional normalization of the raw edge weights. We can represent the source graph $\mathcal{G}_{\mathcal{S}}$ with an $|\mathcal{S}| \times |\mathcal{S}|$ transition matrix $\mathbf{T}'$ where the entries indicate source consensus edge weights:

$$T'_{ij} = \begin{cases} w(s_i, s_j) & \text{if } (s_i, s_j) \in \mathcal{L}_{\mathcal{S}} \\ 0 & \text{otherwise} \end{cases}$$

If we consider a PageRank-style calculation over the source transition matrix $\mathbf{T}'$, we observe a key spam-resilience characteristic. The source consensus edge weighting scheme places the burden on the hijacker (or honeypot) to capture *many* pages within a legitimate source to exert any influence over the spam target pages. Hijacking a few pages in source i will have little impact over the source-level influence flow to a spammer source j; that is $w(s_i, s_j)$ is less subject to manipulation in the presence of many other pages within a source, since it is aggregated over the link characteristics of all pages in the source.

3.3. Influence Throttling

The source view of the Web and the source-based influence flow provide a foundation towards mitigating the influence of link-based manipulation, but there are still open vulnerabilities. First, a spammer may control pages in multiple colluding sources, meaning that the spammer can construct a linking arrangement to ensure any arbitrary edge weight between colluding sources. Second, although the proposed spam-resilient components have some benefit, they are still subject to hijacking and honeypot attacks by a determined Web spammer (e.g., a spammer may have to hijack many more pages than in the page-level ranking model, but there is still room for manipulation in the source-level ranking model). As a result, we next present the final component of Spam-Resilient SourceRank for managing the impact of spammer-controlled links – *influence throttling* – so that a spammer cannot take unfair advantage of the ranking system, even in the presence of large-scale link manipulation.

First, we augment the original source graph $\mathcal{G}_{\mathcal{S}} = \langle \mathcal{S}, \mathcal{L}_{\mathcal{S}} \rangle$ to require that all sources have a self-edge, regardless of the characteristics of the underlying page graph, i.e., $\forall s_i \in \mathcal{S}, (s_i, s_i) \in \mathcal{L}_{\mathcal{S}}$ holds. Including self-edges in the source graph is a sharp departure from the classic PageRank perspective and may initially seem counter-intuitive – since it allows a source to have a direct influence over its own rank – but we will see how it is a critical feature of the Spam-Resilient SourceRank approach.

For each source $s_i \in \mathcal{S}$, we associate a throttling factor $\kappa_i \in [0, 1]$, such that the self-edge weight $w(s_i, s_i) \geq \kappa_i$. We refer to this $|\mathcal{S}|$-length vector $\boldsymbol{\kappa}$ as the *throttling vector*. By requiring a source to direct some minimum amount of influence (κ_i) on itself, we throttle the influence it can pass along to other sources. In the extreme, a source's influence is completely throttled when $\kappa_i = 1$, meaning that all edges to other sources are completely ignored (and hence, the throttled source's influence on other sources is diminished). Conversely, a source's influence is not throttled at all when $\kappa_i = 0$. Based on the throttling vector $\boldsymbol{\kappa}$, we can construct a new influence-throttled transition matrix $\mathbf{T}''$ where the transition probabilities in $\mathbf{T}'$ are:

$$T''_{ij} = \begin{cases} \kappa_i & \text{if } T'_{ij} < \kappa_i \\ & \text{and } i = j \\ \dfrac{T'_{ij}}{\sum_{i \neq k} T'_{ik}} \cdot (1 - \kappa_i) & \text{if } T'_{ij} < \kappa_i \\ & \text{and } i \neq j \\ T'_{ij} & \text{otherwise} \end{cases}$$

For a source that does not meet its minimum throttling threshold (i.e., $T'_{ii} < \kappa_i$), the self-edge weight in the transformed transition matrix is tuned upward (i.e., $T''_{ii} = \kappa_i$), and the remaining edge weights are re-scaled such that $\sum_{i \neq j} T''_{ij} = 1 - \kappa_i$.

Many factors may impact the specific choice of $\boldsymbol{\kappa}$, including the size of the Web dataset, the number of pages considered, the link density, and other link characteristics. In Section 5 we discuss one approach based on *spam-proximity*. The key insight is to tune κ_i higher for known spam sources and those sources that link to known spam sources (e.g., through hijacking, honeypots, or collusion).

3.4. Putting it All Together

Given the three key components introduced in the previous sections, we can now construct the final Spam-Resilient SourceRank ranking model. Similar in spirit to the "random surfer" model often used to describe PageRank, we adopt a source-based random walk model. Intuitively, each source's "authority" (or "importance") is determined by the long-term probability of a random walker visiting each source. Unlike the PageRank random walker, the Spam-Resilient SourceRank random walker can be interpreted as a *selective random walk*, whereby a random walker arrives at a source, and flips a source-specific biased coin. The random walk proceeds as follows. For source $s_i \in \mathcal{S}$:

- With probability $\alpha\kappa_i$, the random walker follows source s_i's self-edge;
- With probability $\alpha(1-\kappa_i)$, the random walker follows one of source s_i's out-edges;
- With probability $1-\alpha$, the random walker teleports to a randomly selected source.

Such a random walk may be modelled by a Markov Chain and written in terms of the transition matrix $\hat{\mathbf{T}}$:

$$\hat{\mathbf{T}} = \alpha \cdot \mathbf{T}'' + (1-\alpha) \cdot \mathbf{1} \cdot \mathbf{c}^T \qquad (2)$$

The teleportation factor is included as a "fix" to guarantee that the transition matrix associated with the Markov chain be both aperiodic and irreducible, which ensures convergence to a stationary distribution. The stationary distribution of the random walk corresponds to the principal eigenvector of $\hat{\mathbf{T}}$, and it encodes the long-term probability of a random walker being at each particular source. The stationary distribution is exactly the Spam-Resilient SourceRank vector $\boldsymbol{\sigma}$. The eigenvector version:

$$\boldsymbol{\sigma}^T = \boldsymbol{\sigma}^T(\alpha \cdot \mathbf{T}'' + (1-\alpha) \cdot \mathbf{1} \cdot \mathbf{c}^T)$$

may be rewritten in a convenient linear form as:

$$\boldsymbol{\sigma}^T = \alpha \cdot \boldsymbol{\sigma}^T \cdot \mathbf{T}'' + (1-\alpha) \cdot \mathbf{c}^T \qquad (3)$$

Coupled with the normalization $\boldsymbol{\sigma}/||\boldsymbol{\sigma}||$, this linear formulation results in exactly the same Spam-Resilient SourceRank vector as the eigenvector problem, but with the added property that the score for any source may be written as a linear combination of the scores of the sources that point to it. For more discussion of this linear formulation, we refer the reader to two recent studies: [8, 25].

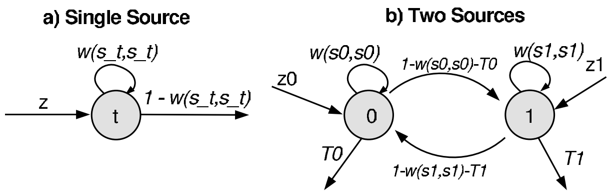


Figure 1: What is the optimal source configuration?

4. Spam Resilience Analysis

In this section, we analyze the spam resilience properties of the proposed ranking model and compare it to PageRank. We consider a Web spammer whose goal is to maximize his influence over a single *target source* through the manipulation of links (both from within the source and from other sources), which corresponds to the vulnerabilities identified in Section 2.

4.1. Link Manipulation Within a Source

We begin by studying link manipulation that is confined to a single source, which would correspond to collusive arrangements among spammer-controlled Web pages like link farms and link exchanges. In the source view of the Web, all intra-source page links are reflected in a single self-edge to the source, and all page links to others sources are reflected in source edges to external sources.

How should the Web spammer configure the target source s_t to maximize its Spam-Resilient SourceRank score, which in turn will have the greatest impact on the target source's rank relative to all other sources? In Figure 1(a), we consider a generic source configuration for s_t. The target has a self-edge weight of $w(s_t, s_t)$, leaving $1 - w(s_t, s_t)$ for all source edges to external sources. Let z denote the aggregate incoming score to the target source from sources beyond the control of the Web spammer. Here, the Web spammer has direct influence over its own links (reflected in $w(s_t, s_t)$) but no influence over the incoming links from other sources. As prescribed in Equation 3, we can write the target source's score:

$$\sigma_t = \alpha z + \alpha \cdot w(s_t, s_t) \cdot \sigma_t + \frac{1-\alpha}{|\mathcal{S}|}$$

$$\sigma_t = \frac{\alpha z + \frac{1-\alpha}{|\mathcal{S}|}}{1 - \alpha \cdot w(s_t, s_t)}$$

which is maximized when $w(s_t, s_t) = 1$. The optimal configuration is for the source to *eliminate all out edges* and retain only a self-edge.Hence, the optimal σ_t^* is:

$$\sigma_t^* = \frac{\alpha z + \frac{1-\alpha}{|\mathcal{S}|}}{1-\alpha} \qquad (4)$$

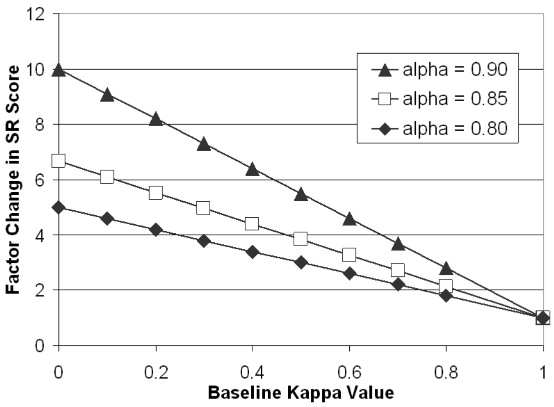


Figure 2: Change in Spam-Resilient SourceRank Score By Tuning κ from a baseline value to 1

Impact of Influence Throttling: Given that the target source has an initial throttling factor $\kappa < 1$ and that $w(s_t, s_t) = \kappa$, the next question to consider is by how much may a source improve its score by adopting a self-edge weight even higher than its throttling factor (i.e., by increasing $w(s_t, s_t)$ beyond the mandated minimum κ throttling value)? Examining the relative SourceRank score for s_t, we have:

$$\frac{\sigma_t^*}{\sigma_t} = \frac{\frac{\alpha z + \frac{1-\alpha}{|S|}}{1-\alpha}}{\frac{\alpha z + \frac{1-\alpha}{|S|}}{1-\alpha\kappa}} = \frac{1-\alpha\kappa}{1-\alpha}$$

For a source with an initial baseline throttling value of $\kappa = 0$, a source may increase its SourceRank score by $\frac{1}{1-\alpha}$ by increasing its $w(s_t, s_t)$ to 1. For typical values of α – from 0.80 to 0.90 – this means a source may increase its score from 5 to 10 times. For sources that are more throttled there is less room for manipulation. In Figure 2, we show for increasing values of a baseline κ, the maximum factor change in Spam-Resilient SourceRank score by tuning the κ value closer to 1. A highly-throttled source may tune its SourceRank score upward by a factor of 2 for an initial $\kappa =$ 0.80, a factor of 1.57 times for $\kappa = 0.90$, and not at all for a fully-throttled source.

By including self-edges in the source graph and the throttling factor κ, we allow a Web spammer some room for manipulating the score of its sources; however, the manipulation is for a *one-time* increase only and it may be limited by tuning the κ throttling factor higher. No such limit is provided under PageRank, meaning that a Web spammer may arbitrarily increase the score of a series of target pages by a factor even larger than we see for SourceRank.

4.2. Cross-Source Link Manipulation

We now study link manipulation across two or more sources, which corresponds to hijacking and honeypots scenarios, as well as collusive arrangements that span multiple sources. For this analysis, the spammer wishes to maximize the score for the single target source by manipulating the links available in one or more *colluding* sources.

In Figure 1(b), we show a generic source configuration for a single target source s_0 and single colluding source s_1. If we let θ_0 and θ_1 denote the edge weighting for each source to sources outside the sphere of influence of the Web spammer. Hence, source s_0 has $1 - w(s_0, s_0) - \theta_0$ edge weighting available for the edge to source s_1. The corresponding edge weight holds for the edge from s_1 to s_0. Let z_0 and z_1 denote the incoming score to each source, respectively, from other sources beyond the control of the Web spammer. Hence, we may describe the Spam-Resilient SourceRank for the two sources with a system of equations, where the Web spammer may manipulate $w(s_0, s_0)$, $w(s_1, s_1)$, θ_0, and θ_1:

$$\begin{aligned}\sigma_0 &= \alpha z_0 + \alpha w(s_0, s_0)\sigma_0 + \frac{1-\alpha}{|S|}\\ &\quad + \alpha(1 - w(s_1, s_1) - \theta_1)\sigma_1\\ \sigma_1 &= \alpha z_1 + \alpha w(s_1, s_1)\sigma_1 + \frac{1-\alpha}{|S|}\\ &\quad + \alpha(1 - w(s_0, s_0) - \theta_0)\sigma_0\end{aligned}$$

Solving and taking the partial derivative with respect to the four parameters, we find that the optimal scenario for a Web spammer who wishes to maximize the SourceRank score for source s_0 is to set $\theta_0 = \theta_1 = 0$, meaning that there are no source edges to sources outside of the Web spammer's sphere of influence; $w(s_0, s_0) = 1$, meaning that the target source points only to itself and not at all to the colluding source; $w(s_1, s_1) = 0$, meaning that the colluding source points only to the target source. With the κ_1 throttling factor requirement this means that the best the colluding source can do is meet the minimum requirement $w(s_1, s_1) = \kappa_1$ and direct the rest $(1 - \kappa_1)$ to the target.

If we extend this analysis to consider x colluding sources (labelled $s_1, ...s_x$) all in service to a single target source, then the system of equations is:

$$\begin{aligned}\sigma_0 &= \alpha z_0 + \alpha w(s_0, s_0)\sigma_0 + \frac{1-\alpha}{|S|}\\ &\quad + \alpha\sum_{i=1}^{x}(1 - w(s_i, s_i))\frac{\alpha z_i + \frac{1-\alpha}{|S|}}{1 - \alpha w(s_i, s_i)}\\ \sigma_i &= \alpha z_i + \alpha w(s_i, s_i)\sigma_i + \frac{1-\alpha}{|S|}\\ &\quad + \alpha(1 - w(s_0, s_0) - \theta_0)\sigma_0\end{aligned}$$

The optimal configuration is for all colluding sources to set $\theta_i = 0$, meaning that there are no source edges from colluding sources to sources outside of the Web spammer's sphere of influence; $w(s_0, s_0) = 1$, meaning that the target source points only to itself and not at all to the colluding source; $w(s_i, s_i) = \kappa_i$, meaning that the colluding source directs the minimum edge weight to itself and the remainder $(1 - \kappa_i)$ to the target source. Hence, each colluding source s_i contributes some SourceRank $\Delta_{\sigma_i}(\sigma_0)$ to the target s_0:

$$\Delta_{\sigma_i}(\sigma_0) = \frac{\alpha}{1-\alpha}\sum_{i=1}^{x}(1-\kappa_i)\frac{\alpha z_i + \frac{1-\alpha}{|S|}}{1-\alpha\kappa_i} \qquad (5)$$

Clearly, tuning the κ throttling factor for each source closer to 1 (meaning that the majority of the colluding source's edge weight is directed to itself) results in a smaller change to the score of the target source. Hence, the introduction of the self-edge and the use of the throttling factor limits the impact of inter-source link manipulation.

Impact of Influence Throttling: To further understand the importance of the κ throttling factor on muting the impact of a Web spammer across sources, we consider a scenario in which a Web spammer controls x colluding sources, that each source has the same throttling factor of κ, and that the sources are configured optimally (as described above). Now suppose the throttling factor is raised to κ' for each source, meaning that each colluding source has less influence on the target source. How many sources x' are needed to achieve the same score as in the original case? I.e., what impact does raising the throttling factor have on the Web spammer?

If we let $z_i = 0$, we may write the original Spam-Resilient SourceRank score with x colluding sources and an initial throttling factor κ as well as the Spam-Resilient SourceRank score under the higher throttling factor (κ') scenario:

$$\sigma_0(x, \kappa) = \frac{\left(\frac{\alpha(1-\kappa)x}{1-\alpha\kappa} + 1\right)\frac{1-\alpha}{|S|}}{1-\alpha}$$

$$\sigma_0(x', \kappa') = \frac{\left(\frac{\alpha(1-\kappa')x'}{1-\alpha\kappa'} + 1\right)\frac{1-\alpha}{|S|}}{1-\alpha}$$

Letting $\sigma_0(x, \kappa) = \sigma_0(x', \kappa')$, we may find a relationship between the number of original colluding sources x and the number of colluding sources x' necessary under the higher throttling factor:

$$\frac{x'}{x} = \frac{1-\alpha\kappa'}{1-\alpha\kappa} \cdot \frac{1-\kappa}{1-\kappa'}$$

In Figure 3, we plot the percentage of additional sources $(\frac{x'}{x} - 1)$ needed for a choice of κ' to equal the same influence on the score of the target page as under an initial choice $\kappa = 0$. For example, when $\alpha = 0.85$ and $\kappa' = 0.6$, there are 23% more sources necessary to achieve the same score as in the case when $\kappa = 0$. When $\kappa' = 0.8$, the Web spammer needs to add 60% more sources to achieve the same influence; for $\kappa' = 0.9$, he needs 135% more sources; and for $\kappa' = 0.99$, he needs 1485% more sources. Tuning the throttling factor higher considerably increases the cost of inter-source manipulation.

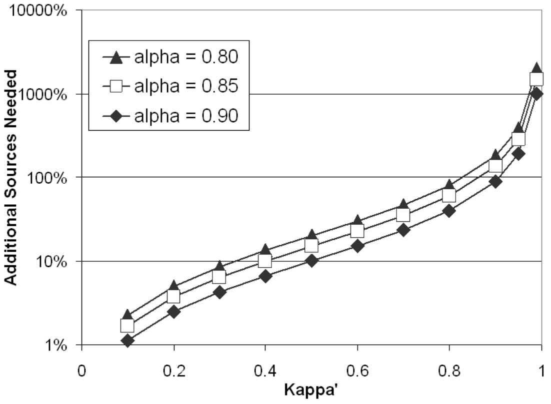


Figure 3: Additional Sources Needed Under the Throttling Factor κ' to Equal the Impact when $\kappa = 0$

4.3. Comparison with PageRank

Now that we have studied Spam-Resilient SourceRank and seen how influence throttling can be used to significantly increase the cost of manipulation to a Web spammer, we next compare Spam-Resilient SourceRank to PageRank. Since PageRank provides page-level rankings, we consider a Web spammer whose goal is to maximize his influence over a single *target page* within a target source. Extending the framework from the previous section, we consider three scenarios:

1. The target page and all colluding pages belong to the same source.

2. The target page belongs to one source, and all colluding pages belong to one colluding source.

3. The target page belongs to one source, and the colluding pages are distributed across many colluding sources.

For each scenario, the colluding pages are structured with a single link to the target page. We consider the impact of an increasing number of colluding pages (τ). Adopting a linear formulation of PageRank that is similar in spirit to Equation 3, we may denote the PageRank score π_0 for the target page in terms of the PageRank due to pages outside of the sphere of influence of the Web spammer, the PageRank due to the teleportation component, and the PageRank due to the τ colluding pages:

$$\pi_0 = z + \frac{1-\alpha}{|\mathcal{P}|} + \tau\alpha\frac{1-\alpha}{|\mathcal{P}|}$$

where α refers to the teleportation probability and $|\mathcal{P}|$ refers to the total number of pages in the page graph. The contribution of the τ colluding pages to the overall PageRank score of the target page is:

$$\Delta_\tau(\pi_0) = \tau\alpha\frac{1-\alpha}{|\mathcal{P}|}$$

For Scenario 1, the Web spammer configures the target source optimally (as we presented in Equation 4), meaning that the colluding pages' intra-source links to the target page have no impact on the Spam-Resilient SourceRank score (other than perhaps a one-time increase due to tuning the self-edge weight up from κ to 1). PageRank, however, is extremely susceptible, as illustrated in Figure 4(a), where the PageRank score (PR) of the target page jumps by a factor of nearly 100 times with only 100 colluding pages.

For Scenario 2, the Web spammer adopts the optimal (worst-case) two-source configuration discussed in the previous section. In this configuration, the target source points only to itself, and the colluding source that contains the colluding pages directs κ edge weight to itself and the rest to the target source. In Figure 4(b), we see how PageRank is again extremely susceptible to such collusion, whereas the maximum influence over Spam-Resilient SourceRank is capped at 2 times the original score for several values of κ. Since PageRank has no notion of a source, makes no effort to regulate the addition of new pages to the Web graph, and has no notion of influence throttling, all three spam scenarios under consideration will have the same extreme impact on the PageRank score of the target page.

In Scenario 3, the Web spammer adopts the optimal configuration for x colluding sources (as we established in the previous section). Figure 4(c) plots the extreme impact on PageRank. As the influence throttling factor is tuned higher (up to 0.99), the Spam-Resilient SourceRank score of the target source is less easily manipulated.

5. Spam-Proximity Throttling

Determining the level of influence throttling for each source is an important component of Spam-Resilient SourceRank. In this section, we discuss one alternative for determining $\boldsymbol{\kappa}$ using the notion of *spam proximity*. Spam proximity is intended to reflect the "closeness" of a source to other spam sources in the source graph. A source is "close" to spam sources if it is a spam source itself; if it directly links to a spam source; or if the sources it directly links to link to spam sources, and so on (recursively).

Given a small seed of known spam sources, we adopt a propagation approach that relies on an inverse PageRank-style model to assign a spam proximity value to *every* source in the Web graph, similar to the BadRank [30] approach for assigning in essence a "negative" PageRank value to spam. First, we reverse the links in the original source graph $\mathcal{G}_\mathcal{S} = \langle \mathcal{S}, \mathcal{L}_\mathcal{S} \rangle$ so that we have a new *inverted* source graph $\mathcal{G}'_\mathcal{S} = \langle \mathcal{S}, \mathcal{L}'_\mathcal{S} \rangle$, where the source edge $(s_i, s_j) \in \mathcal{L}_\mathcal{S} \Rightarrow (s_j, s_i) \in \mathcal{L}'_\mathcal{S}$. A source that is *pointed to* by many other sources in the original graph will now itself point to those sources in the inverted graph. We replace the original transition matrix $\hat{\mathbf{T}}$ with the inverse transition matrix $\hat{\mathbf{U}}$:

$$\hat{\mathbf{U}} = \beta \cdot \mathbf{U} + (1-\beta) \cdot \mathbf{1} \cdot \mathbf{d}^T \qquad (6)$$

where $\mathbf{U}$ is the transition matrix associated with the reversed source graph $\mathcal{G}'_\mathcal{S}$, β is a mixing factor, and $\mathbf{d}$ is a teleportation probability distribution derived from the set of pre-labeled spam sources. An element in $\mathbf{d}$ is 1 if the corresponding source has been labeled as spam, and 0 otherwise. By including the pre-labeled spam sources, the resulting spam-proximity vector is biased towards spam and sources "close" to spam.

Based on these scores, we can assign a throttling value to each source, such that sources that are "closer" to spam sources are throttled more than more distant sources. There are a number of possible ways to assign these throttling values. In this paper, we choose a simple heuristic such that sources with a spam-proximity score in the top-k are throttled completely (i.e., $\kappa_i = 1$ for all s_i in the top-k), and all other sources are not throttled at all. We are exploring this topic in our ongoing research.

6. Experimental Evaluation

In this section, we provide experimental validation for the Spam-Resilient SourceRank approach over a selection of Web spam scenarios to supplement the more exhaustive analysis of the previous section.

6.1. Data and Setup

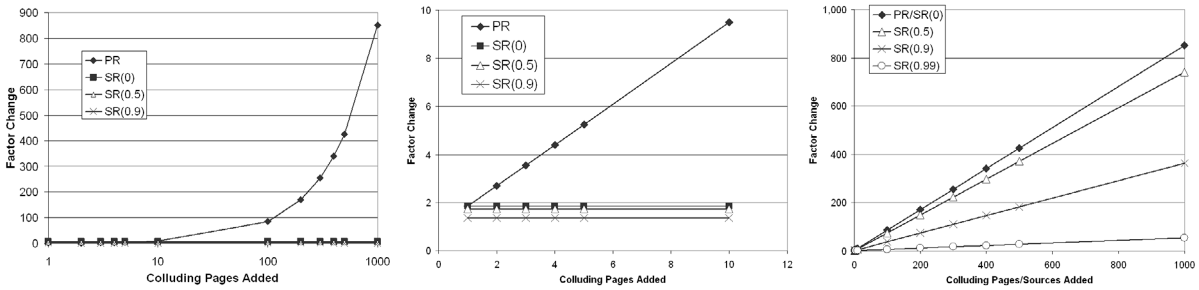
We rely on three real-world Web datasets. The first dataset – **WB2001** – was originally collected by the Stanford WebBase project [2] in 2001 and includes over 118 million pages from a wide variety of top-level-domains. The second dataset – **UK2002** – is derived from a 2002 crawl of the `.uk` top-level-domain by UbiCrawler [9] and consists of over 18 million pages. The last dataset – **IT2004** – is derived from a 2004 crawl of the `.it` top-level-domain, again by UbiCrawler, and consists of over 40 million pages.

For each dataset we extracted the host information for each page URL and assigned pages to sources based on this host information. In Table 1, we present summary information for each of the source graphs.

Table 1: Source Summary

Dataset	**Sources**	**Edges**
UK2002	98,221	1,625,097
IT2004	141,103	2,862,460
WB2001	738,626	12,554,332

All of the Spam-Resilient SourceRank code was written in Java. The data management component was based



(a) (b) (c)

Figure 4: Comparison with PageRank: (a) Scenario 1; (b) Scenario 2; (c) Scenario 3

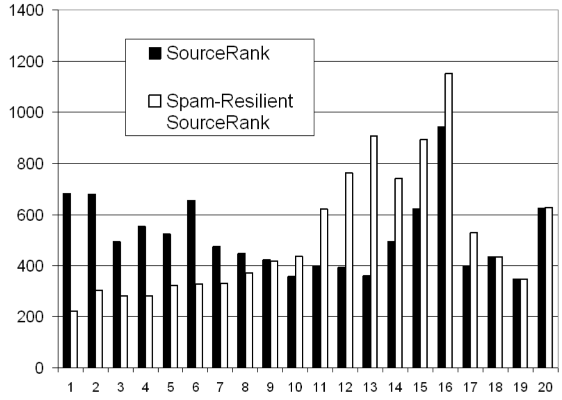


Figure 5: Rank Distribution of All Spam Sources

on the WebGraph compression framework described in [10] for managing large Web graphs in memory. We measured the convergence for PageRank and SourceRank calculations using the L2-distance of successive iterations of the Power Method and terminated the PageRank and SourceRank calculations once the L2-distance dropped below a threshold of 10e-9. For all calculations, we used a mixing parameter of 0.85 (which is typical in the literature, e.g., [28]).

6.2. Influence Throttling Effectiveness

We begin by considering the impact of influence throttling on the spam-resilience characteristics of Spam-Resilient SourceRank. For the WB2001 dataset we manually identified 10,315 pornography-related sources, and labeled these as spam. It is unreasonable for a spam identification algorithm (whether manual or automated) to identify all spam sources with high precision. Hence, of these 10,315 spam sources, we randomly selected just 1,000 (fewer than 10%) to use as a seed set for the spam-proximity calculation. We calculated the spam-proximity score for each source using the approach described in Section 5.

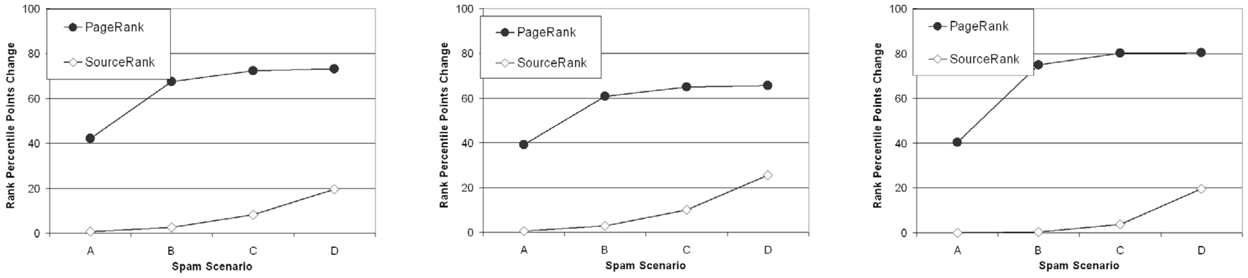
Based on these scores, we assigned an appropriate throttling value to each source, such that sources that are "closer" to spam sources are throttled more than more distant sources. These spam proximity scores are propagated to all sources in the dataset based only on the seed set of 1,000 identified spam sources. We assigned the top-20,000 spam-proximity sources a throttling value of $\kappa = 1$, meaning that their influence was completely throttled. For all other sources we assigned a throttling value of $\kappa = 0$, meaning that these sources were throttled not at all. We then computed the Spam-Resilient SourceRank ranking vector using these throttling values. As a point of comparison, we also computed the baseline SourceRank ranking vector using no throttling information.

For each of the two ranking vectors, we sorted the sources in decreasing order of scores and divided the sources into 20 buckets of equal number of sources. Along the x-axis of Figure 5 we consider these 20 buckets for the WB2001 dataset, from the bucket of top-ranked sources (bucket 1) to the bucket of the bottom-ranked sources (bucket 20). Along the y-axis, we plot the number of actual spam sources (of the 10,315 total spam sources) in each bucket. The Spam-Resilient SourceRank approach using influence throttling penalizes spam sources considerably more than the baseline SourceRank approach, even when fewer than 10% of the spam sources have been explicitly marked as spam.

6.3. Comparison with PageRank

In our next set of experiments we compare Spam-Resilient SourceRank to PageRank. As a baseline, we computed the PageRank vector (π) over each page graph using the parameters typically used in the literature (e.g., [28]). We also computed the Spam-Resilient SourceRank vector for each source graph, using the throttling values described in the previous section. Our goal is to study a spammer's impact on the rank of a target page through increasing degrees of manipulation, both for manipulation in a single source and across sources.

Link Manipulation Within a Source: We first aim to validate the analysis in Section 4.1 by considering the impact of page-level manipulation *within* a single source. We ran-



(a) UK2002

(b) IT2004

(c) WB2001

Figure 6: PageRank vs. Spam-Resilient SourceRank: Intra-Source Manipulation

domly selected five sources from the bottom 50% of all sources that have not been throttled by the spam-proximity influence throttling approach. This corresponds to a worst-case scenario for Spam-Resilient SourceRank, since these sources are essentially "in the clear". For each source, we randomly selected a target page within the source and then added a single new spam page within the same source with a link to the target page. This is case A. We repeated this setup for 10 pages (case B), 100 pages (case C), and 1,000 pages (case D). For each case, we constructed the new spammed page graph and source graph for each of the three Web datasets. We ran PageRank and Spam-Resilient SourceRank for each of the four cases.

In Figure 6, we show the influence of the Web spammer in manipulating the rank of the target page and the rank of the target source through the average ranking percentile increase. For WB2001, the PageRank of the target page jumped 80 percentile points under case C (from an average rank in the 19th percentile to the 99th percentile), whereas the Spam-Resilient SourceRank of the target source jumped only 4 percentile points (from an average rank in the 27th percentile to the 31st percentile).

We first note the dramatic increase in PageRank for the target page across all three Web datasets, which confirms the analysis about the susceptibility of PageRank to rank manipulation. Although PageRank has typically been thought to provide fairly stable rankings (e.g., [27]), we can see how link-based manipulation has a profound impact, even in cases when the spammer expends very little effort (as in cases A and B). The Spam-Resilient SourceRank does increase some, but not nearly as much as PageRank, and it requires considerably more spammer effort to yield any significant rankings change (as in case D). Even then, the spammer's impact is considerably less: $\sim$20 percentile points increase versus $\sim$70 for PageRank.

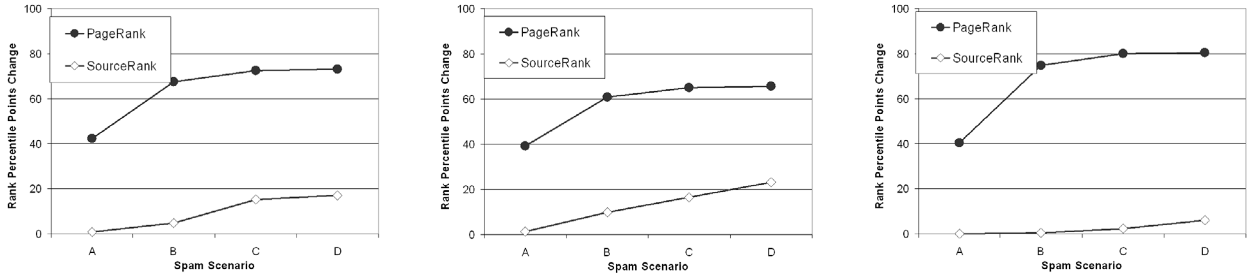
Link Manipulation Across Sources: In the second Web spam scenario, we consider the impact of manipulation *across* sources, which corresponds to the analysis in Section 4.2. For this scenario, the spam links are added to pages in a colluding source that point to the target page in a different source. We paired the randomly selected target sources from the previous experiment with a randomly selected colluding source, again from the bottom 50% of all sources. For each pair, we added a single spam page to the colluding source with a single link to the randomly selected target page within the target source. This is case A. We repeated this setup for 10 pages (case B), 100 pages (case C), and 1,000 pages (case D). For each case, we then constructed the new spammed page graph and source graph for each of the three Web datasets. We ran PageRank and Spam-Resilient SourceRank for each of the four cases.

In Figure 7, we show the influence of the Web spammer in manipulating the rank of the target page and the target source. Since the page-level view of the Web does not differentiate between intra-source and inter-source page links, we again see that the PageRank score dramatically increases, whereas the Spam-Resilient SourceRank score is impacted less. We witness this advantage using no additional influence throttling information for the sources under consideration, meaning that the Spam-Resilient SourceRank advantage would be even greater with the addition of more throttling information.

7. Related Work

As we have noted, several studies have identified large portions of the Web to be subject to malicious rank manipulation [17, 22], especially through the construction of specialized link structures for promoting certain Web pages. In [21], a taxonomy of Web spam is proposed that categorizes various techniques to manipulate search engine results. Several researchers have studied collusive linking arrangements with respect to PageRank, including [31] and [5]. Link farms have been studied in [3]. Separately, optimal link farms and the effectiveness of spam alliances have been studied in [20].

There have been several previous efforts to handle Web spam, beginning with Davison [12], who was the first to investigate the identification of so-called nepotistic links on the Web. Other researchers have attempted to identify spam pages based on a statistical analysis of common Web prop-



(a) UK2002 (b) IT2004 (c) WB2001

Figure 7: PageRank vs. Spam-Resilient SourceRank: Inter-Source Manipulation

erties (like page length) [17]; many outliers in their analysis were, indeed, spam Web pages. Similarly, there have been some initial efforts to learn spam classifiers to distinguish between link farms and legitimate sites [15]. In a similar vein, several researchers have suggested identifying and penalizing pages that derive a large amount of ranking benefit from spam links, e.g., [6], [19], and [30]. Rather than identify spam pages outright, the TrustRank approach propagates trust from a seed set of trusted Web pages [22]. Such a technique is still vulnerable to honeypot and hijacking vulnerabilities, in which high-value trusted pages may be especially targeted.

8. Conclusion

We have presented the Spam-Resilient SourceRank approach for Web spam resilient ranking of Web data sources, and shown how it counters link-based vulnerabilities. In our ongoing research we are developing a model of spammer behavior, including new metrics for the effectiveness of link-based manipulation. Our goal is to evaluate the relative impact on the *value* of a spammer's portfolio of sources due to link-based manipulation.

References

[1] *SEO Black Hat*. http://seoblackhat.com/.

[2] *The Stanford WebBase Project*. http://dbpubs.stanford.edu/ testbed/doc2/WebBase/.

[3] S. Adali, T. Liu, and M. Ismail. Optimal link bombs are uncoordinated. In *AIRWeb*, 2005.

[4] A. Arasu et al. Pagerank computation and the structure of the web. In *WWW*, 2002.

[5] R. Baeza-Yates, C. Castillo, and V. Lopez. Pagerank increase under different collusion topologies. In *AIRWeb*, 2005.

[6] A. A. Benczur et al. Spamrank - fully automatic link spam detection. In *AIRWeb*, 2005.

[7] K. Bharat et al. Who links to whom: Mining linkage between Web sites. In *ICDM*, 2001.

[8] M. Bianchini, M. Gori, and F. Scarselli. Inside PageRank. *ACM TOIT*, 5(1), 2005.

[9] P. Boldi et al. Ubicrawler. In *WWW*, 2002.

[10] P. Boldi and S. Vigna. The WebGraph Framework I. In *WWW*, 2004.

[11] J. Caverlee, L. Liu, and W. B. Rouse. Link-based ranking of the web with source-centric collaboration. In *CollaborateCom*, 2006.

[12] B. Davison. Recognizing nepotistic links on the Web. In *Workshop on AI for Web Search*, 2000.

[13] B. Davison. Topical locality in the web. In *SIGIR*, 2000.

[14] S. Dill et al. Self-similarity in the Web. *ACM TOIT*, 2(3), 2002.

[15] I. Drost and T. Scheffer. Learning to identify link spam. In *ECML*, 2005.

[16] N. Eiron, K. S. McCurley, and J. A. Tomlin. Ranking the Web frontier. In *WWW*, 2004.

[17] D. Fetterly, M. Manasse, and M. Najork. Spam, damn spam, and statistics. In *WebDB*, 2004.

[18] D. Gleich, L. Zhukov, and P. Berkhin. Fast parallel pagerank: A linear system approach. Technical report, Yahoo!, 2004.

[19] Z. Gyöngyi et al. Link spam detection based on mass estimation. In *VLDB*, 2006.

[20] Z. Gyöngyi and H. Garcia-Molina. Link spam alliances. In *VLDB*, 2005.

[21] Z. Gyöngyi and H. Garcia-Molina. Web spam taxonomy. In *AIRWeb*, 2005.

[22] Z. Gyöngyi, H. Garcia-Molina, and J. Pedersen. Combating Web spam with TrustRank. In *VLDB*, 2004.

[23] S. D. Kamvar et al. Exploiting the block structure of the Web for computing PageRank. Technical report, Stanford, 2003.

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

[25] A. N. Langville and C. D. Meyer. Deeper inside PageRank. *Internet Mathematics*, 1(3), 2005.

[26] C. Mann. Spam + blogs = trouble. *Wired*, 2006.

[27] A. Y. Ng, A. X. Zheng, and M. I. Jordan. Stable algorithms for link analysis. In *SIGIR*, 2001.

[28] L. Page et al. The PageRank citation ranking: Bringing order to the Web. Technical report, Stanford, 1998.

[29] S. Webb, J. Caverlee, and C. Pu. Introducing the webb spam corpus: Using email spam to identify web spam automatically. In *CEAS*, 2006.

[30] B. Wu and B. Davison. Identifying link farm spam pages. In *WWW*, 2005.

[31] H. Zhang et al. Improving eigenvector-based reputation systems against collusions. In *Algorithms and Models for the Web Graph*, 2004.