Propagating Trust and Distrust to Demote Web Spam

Baoning Wu Vinay Goel Brian D. Davison
Department of Computer Science & Engineering
Lehigh University
Bethlehem, PA 18015 USA
{baw4,vig204,davison}@cse.lehigh.edu

ABSTRACT

Web spamming describes behavior that attempts to deceive search engine's ranking algorithms. TrustRank is a recent algorithm that can combat web spam by propagating trust among web pages. However, TrustRank propagates trust among web pages based on the number of outgoing links, which is also how PageRank propagates authority scores among Web pages. This type of propagation may be suited for propagating authority, but it is not optimal for calculating trust scores for demoting spam sites.

In this paper, we propose several alternative methods to propagate trust on the web. With experiments on a real web data set, we show that these methods can greatly decrease the number of web spam sites within the top portion of the trust ranking. In addition, we investigate the possibility of propagating distrust among web pages. Experiments show that combining trust and distrust values can demote more spam sites than the sole use of trust values.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval

General Terms

Algorithms, Performance

Keywords

Web spam, Trust, Distrust, PageRank, TrustRank

1. INTRODUCTION

In today's Web, a link between two pages can be considered to be an implicit conveyance of trust from the source page to the target page. In this case, trust implies that the author of the source page believes that the target page provides some content value.

With the increasing commercial interest of being ranked high in search engine results, content providers resort to techniques that manipulate these results. This behavior is usually termed Web spam, or search engine spam. Many kinds of spam have been discovered [24, 12, 5]. Henzinger et al. [15] mention that Web spam is one of the major challenges faced by search engines. There is no universal method that can detect all kinds of spam at the same time.

Trust can be used to combat Web spam. Gyöngyi et al. [13] present the TrustRank algorithm based on this idea. This technique assumes that a link between two pages on the Web signifies trust between them; i.e., a link from page A to page B is a conveyance of trust from page A to page B. In this technique, human experts, initially, select a list of seed sites that are well-known and trustworthy on the Web. Each of these seed sites is assigned an initial trust score. A biased PageRank [23] algorithm is then used to propagate these trust scores to the descendants of these sites. The authors observed that on applying this technique, good sites had relatively high trust scores, while spam sites had low trust scores.

TrustRank shows that the idea of propagating trust from a set of highly trusted seed sites helps a great deal in the demotion of Web spam. But TrustRank is just one implementation of this idea. This approach makes certain assumptions with regard to how trust is propagated from a parent page to a child page. For example, the authors claim that the possibility of a page pointing to a spam page increases with the number of links the pointing page has. Because of this, they proposed the idea that the trust score of a parent page be equally split amongst its children pages.

This assumption is open to argument. Why should two equally trusted pages propagate different trust scores to their children just because one made more recommendations than the other? Also, with respect to the accumulation of trust scores from multiple parents, TrustRank puts forth just one solution, that of simple summation. Clearly, there are other alternatives.

A natural extension of the idea of the conveyance of trust between links is that of the conveyance of distrust. Here, distrust has a different meaning to that in the context of social networks. In social networks, distrust between two nodes A and B usually means that A shows distrust explicitly to B. In contrast, in our system, distrust is a penalty awarded to the source page for linking to an untrustworthy page. Hence, this distrust is an indication that we don't trust some web pages, not an indication that one page doesn't trust another page on the web. Actually, the trust score of a page can also be interpreted as how much we trust this page.

In general, spam pages can be considered to be one type of untrustworthy pages. To elaborate on this idea, consider that a page links to another page and hence according to the above definition of trust, this page expresses trust towards the target page. But if this target page is known to be a spam page, then clearly the trust judgment of the source page is not valid. The source page needs to be penalized

for trusting an untrustworthy page. It is likely that the source page itself is a spam page, or is a page that we believe should not be ranked highly for its negligence in linking to an untrustworthy page.

In this paper, we explore the different issues present in the problem of propagating trust on the Web. We also study the application of propagating distrust on the Web. Additionally, we present techniques to combine trust and distrust scores to improve the overall performance in demoting Web spam.

The rest of this paper is organized as follows: the background and related work will be introduced in Section 2 and Section 3 respectively. The motivation of this work will be introduced in Section 4. The details of our technique are given in Section 5. The experiments and results will be shown in Section 7. We finish with discussion and conclusion in Sections 8 and 9.

2. BACKGROUND

2.1 Matrix Definition

The web can be represented by a directed graph, given web pages as the nodes and hyperlinks among web pages as the directed links among the nodes. The adjacency matrix M of the web graph is: $M[i, j]$ equals 1 if there is a hyperlink from page i to page j, or 0 otherwise. Suppose we use $I(i)$ to represent the in-degree of node i and $O(i)$ as the out-degree of node i, the definition of the transition matrix T is:

$$T[i, j] = M[j, i]/O(j) \tag{1}$$

and the definition of the reverse transition matrix R is:

$$R[i, j] = M[i, j]/I(j) \tag{2}$$

2.2 TrustRank and BadRank

Gyöngyi et al. [13] introduce TrustRank. It is based on the idea that good sites seldom point to spam sites and people trust these good sites. This trust can be propagated through the link structure on the Web. So, a list of highly trustworthy sites are selected to form the seed set and each of these sites is assigned a non-zero initial trust score, while all the other sites on the Web have initial values of 0. Then a biased PageRank algorithm is used to propagate these initial trust scores to their outgoing sites. After convergence, good sites will get a decent trust score, while spam sites are likely to get lower trust scores. The formula of TrustRank is:

$$t = (1 - \alpha) \times T \times t + \alpha \times s \tag{3}$$

where t is the TrustRank score vector, α is the jump probability, T is the transition matrix and s is the normalized trust score vector for the seed set. Before calculation, t is initialized with the value of s. Gyöngyi et al. iterated the above equation 20 times with α set to 0.15.

In many SEO discussion boards, participants discuss the latest ranking and spam-finding techniques employed by commercial search engines. One approach, called BadRank[1], is believed by some to be used by a commercial engine to combat link farms.[2] BadRank is based on propagating negative value among pages. The idea of BadRank is that a page will get high BadRank value if it points to some pages with high BadRank value. This idea is similar in spirit to our mechanism of propagating distrust in this paper.

[1]One description of BadRank can be found at [1].

[2]See, for example `http://www.webmasterworld.com/forum3/20281-22-15.htm`.

3. RELATED WORK

While the idea of a focused or custom PageRank vector has existed from the beginning [23], Haveliwala [14] was the first to propose the idea of bringing topical information into PageRank calculation. In his technique, pages listed in DMOZ [22] are used as the seed set to calculate the biased PageRank values for each of the top categories. Then a similarity value of a query to each of these categories is calculated. A unified score is then calculated for each page containing the given query term(s). Finally, pages are ranked by this unified score. Experiments show that Topic-sensitive PageRank has better performance than PageRank in generating better response lists to a given query.

Jeh and Widom [17] specialize the global notion of importance that PageRank provides to create personalized views of importance by introducing the idea of preference sets. The rankings of results can then be biased according to this personalized notion. For this, they used the biased PageRank formula.

Several researchers have done some work to combat different kind of Web spam. Fetterly et al. propose using statistical analysis to detect spam [7]. Acharya et al. [2] first publicly propose using historical data to identify link spam pages. Wu and Davison [26] proposed using the intersection of the incoming and outgoing link sets plus a propagation step to detect link farms. Mishne et al. [20] used a language model to detect comment spam. Drost and Scheffer [6] proposed using a machine learning method to detect link spam. Recently, Fetterly et al. [8] describe methods to detect a special kind of spam that provides pages by stitching together sentences from a repository.

Benczur et al. proposed SpamRank in [4]. For each page, they check the PageRank distribution of all its incoming links. If the distribution doesn't follow a normal pattern, the page will be penalized and used as seed page. They also adopt the idea that spam values are propagated backward and finally spam pages will have high SpamRank values. Compared to SpamRank, we use labeled spam pages as our seed set.

In prior work, we [27] pointed out that TrustRank has a bias towards better represented communities in the seed set. In order to neutralize this bias, we proposed "Topical TrustRank", which uses topics to partition the seed set and different mechanisms to combine trust scores from each partition. We showed that this algorithm can perform better than TrustRank in reducing the number of highly ranked spam sites. Compared with that paper, we do not consider partitions for the seed set here. Instead, we show that different mechanisms for propagating trust can also help to demote more top ranked spam sites. The methods proposed in this paper can generate better performance than Topical TrustRank.

Guha et al. [11] study how to propagate trust scores among a connected network of people. Different propagation schemes for both trust score and distrust score are studied based on a network from a real social community website. Compared with their ideas, our definition of distrust is not exactly same. Their goal is to predict whether two people

will show trust (or distrust) to the other, but our goal is to use trust and distrust to demote Web spam, especially top ranked spam pages or sites.

Massa and Hayes [19] review several current proposals for extending the link mechanism to incorporate extra semantic information, primarily those that allow the authors of a web page to describe their opinion on pages they link to. They argue that any change to the hyperlink facility must be easily understood by the ordinary users of the Web, but the more expressive linking structure would produce a richer semantic network from which more precise information can be mined. They used a real world data set from Epinions.com as a proxy for the Web with the analogy that web pages are Epinions users and links are trust and distrust statements. They show that this additional link information would allow the PageRank algorithm to identify highly trusted web sites.

Ziegler and Lausen [28] introduce the Appleseed algorithm, a proposal for local group trust computation. The basic intuition of the approach is motivated by spreading activation strategies. The idea of spreading activation is the propagation of energy in a network. Also, the edges between the nodes are weighted based on the type of the edges. This idea of energy flow is tailored for trust propagation. In contrast, our algorithm doesn't consider a weighted graph.

Gray et al. [9] proposed a trust-based security framework for ad hoc networks. The trust value among two nodes connected by a path is the average of the weighted sum of trust values of all nodes in the path. No experimental results are shown.

4. MOTIVATION

The original TrustRank paper proposed that trust should be reduced as we move further and further away from the seed set of trusted pages. To achieve this attenuation of trust, the authors propose two techniques, trust dampening and trust splitting. With trust dampening, a page gets the trust score of its parent page dampened by a factor less than 1. With trust splitting, a parent's trust score is equally divided amongst its children. A child's overall trust score is given by the sum of the shares of the trust scores obtained from its parents.

In the case of trust splitting, we raise a question: Given two equally trusted friends, why should the recommendations made by one friend be weighted less than the other, simply because the first made more recommendations? A similar argument has been made by Guha [10].

It is observed that a spam page often points to other spam pages for the purposes of boosting their PageRank value and manipulating search engine results [26]. Motivated by the idea of trust propagation, we believe that propagating distrust given a labeled spam seed set, will help to penalize other spam pages.

Hence, given a set of labeled spam seed set, we can propagate distrust from this set to the pages that point to members of this set. The idea is that a page pointing to a spam page is likely to be spam itself. But sometimes, good pages may unintentionally point to spam pages. In this case, these pages are penalized for not being careful with regard to creating or maintaining links (as suggested by [3]).

In doing so, each page on the Web is assigned two scores, a trust score and a distrust score. In the combined model, a link on the Web can then propagate these two scores. As shown in Figure 1, suppose there is a link from Page A to Page B, then trust is propagated from Page A to Page B, while distrust is propagated from Page B to Page A.

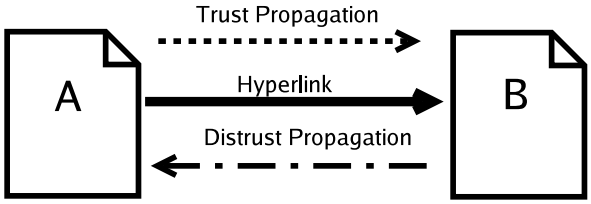

Figure 1: A link on the Web can propagate both trust and distrust.

We explore different techniques for the handling of propagation of trust and distrust from the respective seed sets to other pages on the Web.

5. ALGORITHM DETAILS

In this section, we present details of our ideas on propagating trust and distrust among web pages.

5.1 Propagating Trust

TrustRank propagates trust among web pages in the same manner as the PageRank algorithm propagates authority among web pages. The basic idea is that during each iteration, a parent's trust score is divided by the number of its outgoing links and each of its children gets an equal share. Then a child's overall trust score is the sum of the shares from all its parents.

Two key steps in the technique described above may be explored. One is, for each parent, how to divide its score amongst its children; we name this the "splitting" step. The other is, for each child, how to calculate the overall scores given the shares from all its parents; we name this the "accumulation" step.

For the splitting step, we study three choices:

- **Equal Splitting:** a node i with $O(i)$ outgoing links and trust score $TR(i)$ will give $d \times \frac{TR(i)}{O(i)}$ to each child. d is a constant with $0 < d < 1$;
- **Constant Splitting:** a node i with trust score $TR(i)$ will give $d \times TR(i)$ to each child;
- **Logarithm Splitting:** a node i with $O(i)$ outgoing links and trust score $TR(i)$ will give $d \times \frac{TR(i)}{log(1+O(i))}$ to each child.

We term d to be the decay factor, which determines how much of the parents' score is propagated to its children. In fact, if d equals 1, then the above "Equal Splitting" is the same as the method used in TrustRank. As discussed in the Section 4, why should equally trusted pages propagate different trust scores just because they have different number of children? With "Constant Splitting", each parent will give a constant portion of its trust value to all of its children irrespective of the number of its children. Thus for a child, if two of its parents have identical trust values but different number of children, then the child will get the same value from both of these parents. The third choice, "Logarithm Splitting" does not eliminate the effect of the number of children that a page has but can decrease it.

Since "Equal Splitting" is the choice already being employed in TrustRank, we will focus on "Constant Splitting" and "Logarithm Splitting" in our experiments.

For the accumulation step, we study three choices.

- **Simple Summation:** Sum the trust values from each parent.
- **Maximum Share:** Use the maximum of the trust values sent by the parents.
- **Maximum Parent:** Sum the trust values in such a way as to never exceed the trust score of the most-trusted parent.

The first choice is the same as in PageRank and TrustRank; using the sum of trust scores from all parents as the child's trust score. For "Maximum Share", the maximum value among the trust values inherited from all the parents is used as the child's trust score. For "Maximum Parent", first the sum of trust values from each parent is calculated and this sum is compared with the largest trust score among each of its parents, the smaller of these two values is used as the child's trust score.

By using the above choices, the equation for calculating trust score is different from Equation 3. For example, if using "Constant Splitting" and "Simple Summation", the equation will become:

$$t = (1-\alpha) \times d \times M^T \times t + \alpha \times s \tag{4}$$

where t is the trust score vector, α is the jump probability, d is the constant discussed in the above splitting choices, M is the web matrix shown in Section 2.1 and s is the normalized trust score vector for the seed set.

5.2 Propagating Distrust

The trust score of a page is an indication of how trustworthy the page is on the Web. In the case of web spam, the trust score can be seen as a measure of the likelihood that a page is not a spam page.

Similarly, we introduce the concept of distrust to penalize the pages that point to untrustworthy pages. Now, it is possible that pages unintentionally point to spam pages. In these cases, we argue that the (otherwise good) page should be penalized to some extent for not being careful in its linking behavior.

Distrust propagation makes sense when spam sites are used as the distrusted seed set and distrust is propagated from a child to its parent. So, based on this idea, one link can represent two propagation processes, i.e., the trust score is propagated from the parent to the children while the distrust score is propagated from the children to the parent.

In this technique, some known spam pages are selected as the distrusted seeds and assigned some initial distrust scores. During each iteration, the distrust score is propagated from children pages to parent pages iteratively. After convergence, a higher distrust score indicates that this page is more likely to be a spam page.

A direct method of calculating distrust score for each page is to follow the same idea as TrustRank. The calculation can be represented by Equation 5.

$$n = (1-\alpha) \times R \times n + \alpha \times r \tag{5}$$

where n is the distrust score vector, α is the jump probability, R is the reverse transition matrix shown in Equation 2 and r is the normalized distrust score vector for the distrusted seed set. Before calculation, n is initialized with the value of r.

However, as discussed in Section 5.1, the propagation mechanism of TrustRank may not be optimal to propagate trust or distrust for the purpose of demoting spam pages. We propose that the same choices to propagate trust, discussed in Section 5.1, can be taken to propagate distrust.

Suppose we use $DIS\_TR(i)$ to represent the distrust score for node i. For the splitting step, we have three choices:

- **Equal Splitting:** a node i with $I(i)$ incoming links and $DIS\_TR(i)$ will give $d_D \times \frac{DIS\_TR(i)}{I(i)}$ to each parent. where $0 < d_D < 1$;
- **Constant Splitting:** a node i with $DIS\_TR(i)$ will give $d_D \times DIS\_TR(i)$ to each parent;
- **Logarithm Splitting:** a node i with $I(i)$ incoming links and $DIS\_TR(i)$ will give $d_D \times \frac{DIS\_TR(i)}{log(1+I(i))}$ to each parent.

The "Equal Splitting" choice is quite similar to that in the case of trust propagation in TrustRank. Intuitively, this kind of splitting may raise problems when the purpose of propagating distrust is to demote spam. For a simple example, by "Equal Splitting", a spam site with more parents will propagate smaller distrust to its parents, while spam sites with fewer parents will propagate bigger distrust to its parents. Obviously, this policy supports popular spam sites and this is clearly not desirable for the purpose of demoting spam. In comparison, "Constant Splitting" and "Logarithm Splitting" present better choices.

For the accumulation step, we also have three choices:

- **Simple Summation:** Sum the distrust values from each child.
- **Maximum Share:** Use the maximum of the distrust values sent by the children;
- **Maximum Parent:** Sum the distrust values in such a way as to never exceed the distrust score of the most-distrusted child.

Different choices will employ different equations during the calculation. For example, if using "Constant Splitting" and "Simple Summation", the equation of calculating distrust score is:

$$n = (1-\alpha) \times d_D \times M \times n + \alpha \times r \tag{6}$$

where n is the distrust score vector, α is the jump probability, d is the constant discussed in the above splitting choices, M is the web matrix shown in Section 2.1 and r is the normalized distrust score vector for the distrusted seed set.

5.3 Combining Trust and Distrust

On propagating trust and distrust to the pages on the web, each page will be assigned two scores, a trust score and a distrust score. Then comes the question of combining them to generate a unified ranking of pages that is indicative of their trustworthiness.

Our goal of propagating trust and distrust is to demote spam sites in the ranking. Since the trust score is an indication of how unlikely it is that the page is a spam page,

while the distrust score is an indication of how likely it is that the page is a spam page, a direct solution is to simply calculate the difference of these two scores and use this value to represent the overall trustworthiness of the Web page.

Additionally, we may apply several methods for the combination. For example, we may give different weights when calculating the sum. Suppose we use $Total(i)$ to represent the difference of trust and distrust score for page i. Then we can apply the following formula:

$$Total(i) = \eta \times TR(i) - \beta \times DIS\_TR(i) \quad (7)$$

where η and β ($0 < \eta < 1$, $0 < \beta < 1$) are two coefficients to give different weights to trust and distrust scores in this formula.

6. DATA SET

The data set used in our experiments is courtesy of search.ch search engine [25]. It is a 2003 crawl of pages that are mostly from the Switzerland domain. There are about $20M$ pages within this data set and around $350K$ sites with the ".ch" domain. Since we were also provided with $3,589$ labeled sites and domains applying different spam techniques, we used the site graph for testing the ideas we propose in this paper.

In order to generate a trusted seed set, we extract all the URLs listed within the search.ch topic directory [25] of 20 different topics, which is similar to the DMOZ directory but only lists pages primarily within the Switzerland domain. Since we use the site graph in our calculation and the topic directory listed only pages, we used a simple transfer policy: if a site had a page listed in a certain topic directory, we put the site into a trusted seed set. In doing so, we marked $20,005$ unique sites to form the seed set.

For the generation of a distrusted seed set, we use the labeled spam list which contains $3,589$ sites or domains. In our experiments, we use only a portion of this list as the distrusted seed set with the rest being used to evaluate the performance.

7. EXPERIMENTS

We test all the ideas we propose in Section 5 by using the search.ch data set. Since the goal of this paper is to investigate how different mechanisms of propagating trust and distrust can help to demote top ranking spam sites, we will focus on the ranking positions of the labeled $3,589$ spam sites.

We first calculate the PageRank value for each site based on the search.ch site graph. These sites are then ranked in a descending order of their PageRank values. Based on this ranking, we divide these sites among 20 buckets, with each bucket containing sites with the sum of their PageRank values equal to $1/20th$ of the sum of the PageRank values of all sites.

We then calculate the TrustRank score for each site based on the site graph, to generate a ranking of sites sorted in the descending order of these scores. As in the case of the TrustRank paper [13], we iterated 20 times during this calculation. We then divide these sites among 20 buckets such that each TrustRank bucket has an identical number of sites to the corresponding PageRank bucket. The distribution of the 3,589 spam sites in the 20 buckets by PageRank and TrustRank is shown in Figure 2. It is clear that TrustRank is good at demoting spam sites compared to PageRank.

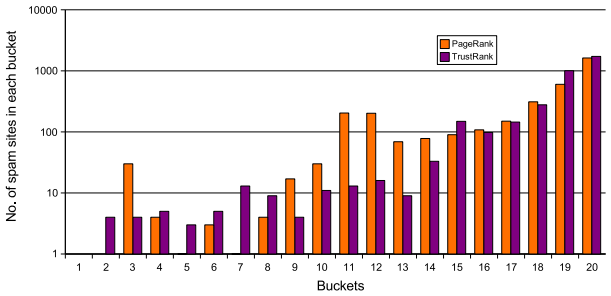

Figure 2: Number of spam sites within each bucket by PageRank and TrustRank.

In this paper, we use the number of spam sites within the top 10 buckets as the metric for measuring the performance of algorithms. This choice of choosing the top 10 buckets was arbitrary as in the case of [27]. The smaller the number of spam sites in the top 10 buckets, the better the performance of the algorithm in demoting spam sites from the top ranking positions.

The results of this metric for the PageRank and TrustRank algorithms are shown in Table 1. These results will be used as the baseline results. We can see that PageRank ranks 90 spam sites within the top ten buckets, while TrustRank ranks only 58 spam sites.

7.1 Different Jump Probabilities

In TrustRank, the jump probability α in Equation 3 is usually assigned a value of 0.15. We measure the performance of TrustRank with different values of this jump factor.

Since we use all the URLs listed in dir.search.ch as the trusted seed set, it is quite possible that some spam sites get included in this set too. On checking, we find that 35 labeled spam sites are within the trusted seed set. It is worthwhile to drop these spam sites from the seed set. We run TrustRank again with different jump probabilities after dropping these 35 labeled spam sites from the seed set.

The results with both the original seed set and the cleaned seed set are shown in Figure 3. We observe that larger jump probabilities decrease the number of spam sites from top ranking positions. Since a larger jump probability means that smaller trust values are propagated from a parent to its children, the results show that for the purpose of demoting spam sites, in TrustRank, a better approach is of relatively little trust propagation. We also observe that the dropping of spam sites from the seed set results in fewer spam sites within the top ten buckets.

Algorithm	No. of Spam sites in top 10 buckets
PageRank	90
TrustRank	58

Table 1: Baseline results for search.ch data set.

Algorithm	Constant Splitting				Logarithm Splitting			
d value	d=0.1	d=0.3	d=0.7	d=0.9	d=0.1	d=0.3	d=0.7	d=0.9
Simple summation	364	364	364	364	364	364	364	364
Maximum Share	34	34	34	34	13	12	20	18
Maximum Parent	27	32	33	33	372	27	29	32

Table 2: Results for the combination of different methods of propagating trust. Experiments are done with different values for d. Only trust score is used in this table.

7.2 Different Trust Propagation Methods

As introduced in Section 5, we explore two choices in the splitting step: "Constant Splitting" ($d \times TR(i)$) and "Logarithm Splitting" ($d\frac{TR(i)}{log(1+O(i))}$), while we have three choices in the accumulation step: "Simple Summation", "Maximum Share" and "Maximum Parent".

The number of different combinations of the above choices is six. For each combination we try using different values of d ranging from 0.1 to 0.9. The results of these six combinations with different values of d are shown in Table 2.

From the results in Table 2, we can tell that "Simple Summation" always generates the worst performance, which is worse than TrustRank and even PageRank. A lot of spam sites are raised in the ranking. Intuitively, this "Simple Summation" will boost the rankings of sites with multiple parents. In general, it is likely a spam site that has a large number of incoming links will be able to accumulate a fairly large value of trust. Hence, spam sites may be benefited by this "Simple Summation" method.

We also observe that, in most cases, both "Maximum Share" and "Maximum Parent" methods generate much better performance than TrustRank and the "Simple Summation" method. With regard to the splitting methods, we observe that in most cases, "Logarithm Splitting" performs better than "Constant Splitting".

The results clearly demonstrate that for the purpose of demoting web spam, propagating trust based on the idea of "Equal Splitting" and "Simple Summation" which is used by TrustRank, is not the optimal solution.

Gyöngyi et al. [13] mentioned that there are different possibilities for splitting trust scores; the reason that they chose the method similar to PageRank is that only minor changes are needed for calculating TrustRank by using existing efficient methods for computing PageRank [18]. We argue that if different choices of splitting and accumulating trust can greatly demote spam sites, it is worthwhile to implement these choices. In Table 2, our best result is 12 spam sites in the top ten buckets, which is a much greater improvement when compared to the baseline results of 58 spam sites in Table 1.

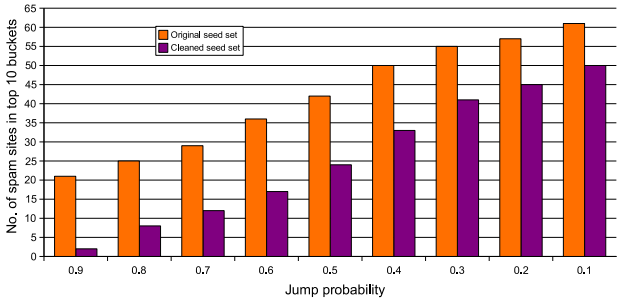

Figure 3: Number of top ranked spam sites with different jump probabilities for TrustRank.

It is worth mentioning that by introducing the above ideas of splitting and accumulating trust, we notice, in some cases, long ties in the trust scores. For example, the top several thousand of sites may have identical trust scores. This is different from the values by PageRank or TrustRank. We think this tie is still reasonable as long as few spam sites are in the ties close to the top. Since there are $3,823$ sites in the top ten buckets by PageRank, we consider the ties that have rankings around this position still within top ten buckets, thus all the spam sites before or within this tie will still be counted within top ten buckets.

Actually, we find that for most cases, these ties can help to demote more spam sites. But some small d may cause a strong tie with more than $10,000$ sites and thus raise the number of spam sites within top ten buckets. One example is that there are 372 spam sites within top ten buckets when combining "Maximum Parent" and "Logarithm Splitting" with d set to 0.1.

7.3 Introducing Distrust

Trust can be propagated from trusted seed set to the children pages iteratively. Similarly, distrust can be propagated from a distrusted seed set to the parent pages iteratively. While our distrusted seed set was provided to us, in general a search engine will maintain a spamming blacklist, using both manual and automatic methods (perhaps, e.g., [7, 8, 26, 21]).

In order to investigate whether introducing distrust can help to improve the performance in demoting spam sites, we randomly select a portion of labeled spam sites as the distrusted seed set and calculate distrust values for each site. The ranking positions of the remaining spam sites will be used to evaluate the performance.

7.3.1 *Basic Propagation of Distrust*

As described, there are several different choices of propagating distrust among web pages, we first use the method shown in Equation 5.

We randomly select 200 spam sites from the $3,589$ labeled spam sites as the distrusted seed set to calculate distrust score. Then we calculate the sum of this distrust score and the trust score generated by TrustRank. By using the sum for ranking, we count the number of spam sites (m) in the top ten buckets as in the case of previous experiments.

But we can not compare the above number m directly with the results shown in Table 1. The reason is that some top ranked spam sites may have been selected in the distrusted seed and they will get demoted as an effect of their selection, not as an effect of our algorithm. Thus, in order to be fair,

Algorithm	Constant Splitting				Logarithm Splitting			
d_D value	d_D=0.1	d_D=0.3	d_D=0.7	d_D=0.9	d_D=0.1	d_D=0.3	d_D=0.7	d_D=0.9
Simple Summation	53	53	55	55	57	53	53	53
Maximum Share	53	53	53	53	59	53	52	52
Maximum Parent	53	53	53	53	57	53	53	53

Table 3: Results for different methods of propagating distrust. The ranking is determined by the combination of distrust score and TrustRank.

we need to count the number of spam sites (n) that are in the top ten buckets by TrustRank which are also in the distrusted seed set. Only when the sum of m and n is smaller than 58, which is listed in Table 1, we can claim that the performance is better than that of TrustRank.

Also the random selection of distrusted seeds may still not be representative of the 3,589 spam sites. In order to neutralize this bias, we repeated the above seed selection five times for calculating distrust scores. Then we use the average results as the final results for the distrusted seed set with 200 seeds. On average, there are 54 spam sites still in the top ten buckets and 4 spam sites are in the distrusted seed set. The sum of 54 and 4 equals the number of spam sites, which is 58, in top ten TrustRank buckets; this shows that using TrustRank's mechanism (Equation 5) to propagate distrust is not helpful in demoting top ranked spam sites.

In order to verify whether introducing more distrusted seeds with this basic distrust propagation is useful, we generated distrusted seed sets of sizes ranging from 200 to 1,000. Similarly, for each seed set size, we repeated this generation five times. The average results are shown in Table 4. The results show that no matter how many seeds are selected for the distrusted seed set, the sum of the second element and third element in Table 4 is always around 60. Since this sum is quite close to the 58 spam sites in Table 1, we believe that using the same mechanism as TrustRank to propagate distrust can not help to demote top ranked spam sites.

7.3.2 Different Choices of Propagating Distrust

Since we have shown that propagating distrust by using the TrustRank mechanism may not be helpful, the next obvious step is to investigate whether the choices of propagating trust can also be applied for propagating distrust in order to demote top ranked spam sites.

Similar to the methods used for generating results in Table 2, we applied the six combinations of different choices for the splitting step and accumulation steps to the propagation of distrust. In order to evaluate the performance, for each combination, we calculate the sum of the distrust value and TrustRank value for each site. Then this sum is used for ranking. Since the TrustRank value is unchanged for each different combination, we can see how different choices of propagating distrust can affect the overall performance and thus we can tell which choice is better for propagating distrust. For simplicity, we only choose 200 spam sites to generate the distrusted seed set once. Results of six different combinations with different d values are shown in Table 3.

From the results in Table 3, we can see that some choices can help to demote more spam sites than others. For example, the combination of "Logarithm Splitting" and "Maximum Share" with d set to 0.7 or 0.9.

Number of seeds	No. of Spam sites in top 10 buckets	No. of Spam sites in seed set
200	54	4
400	55	5
600	49	12
800	48	13
1000	45	16

Table 4: Results when using same mechanism as the propagation of trust in TrustRank to propagate distrust.

7.4 Combining Trust and Distrust Values

In the above experiments, we use the sum of the trust and distrust values as the final value for ranking. As discussed in Section 5, we may use different weights to combine trust and distrust values.

In practice, we did the following experiment to show how the combination of trust and distrust values can affect performance.

- To calculate trust score, we select the choice that can generate best performance in Table 2, i.e., using "Maximum Share" for accumulation and "Logarithm Splitting" for splitting while with d set to 0.3.
- To calculate distrust score, we select the choice that can generate best performance in Table 3, i.e., using "Maximum Share" for accumulation and "Logarithm Splitting" for splitting with d_D set to 0.9.
- For combining trust and distrust values, we follow the Equation 7, with β equals $1 - \eta$. Test with different values of η.
- We test with different numbers of distrusted seeds.

The results for these experiments are shown in Figure 4. There are three lines in the figure. Each represents the results by using 200, 400, 600 spam sites as distrusted seed respectively. From these results, we can tell that an increase in the size of the distrusted seed set will result in an increase in performance.

Compared with the baseline results in Figure 1, more than 80% of spam sites disappear from the top ten buckets. This verifies our hypothesis that using different trust propagation methods together with distrust propagation can help to demote spam sites effectively.

Actually, the results in Figure 4 are not our best results. During our experiments, we find that by using "Constant Splitting" and "Maximum Parent" for trust propagation, "Logarithm Splitting" and "Maximum Share" for distrust propagation with d, d_D and η as 0.1, we can remove all the spam sites from the top ten buckets. We believe that there may be several other combinations that generate optimal results. However, due to resource constraints, we have not enumerated every such combination.

Algorithm	Constant Splitting				Logarithm Splitting			
d value	d=0.1	d=0.3	d=0.7	d=0.9	d=0.1	d=0.3	d=0.7	d=0.9
Maximum Share	77.71	77.73	77.74	77.74	77.19	77.72	77.73	77.73
Maximum Parent	77.52	77.71	77.73	77.74	76.93	77.60	77.71	77.72

Table 5: Percentage of sites affected by combining different ideas to propagate trust.

7.5 Impact of Trust Propagation

Since the trust or distrust scores are propagated from limited number of seed pages, it is quite possible that only a part of the whole web graph can be touched by this propagation. In other words, some pages will have zero values after the algorithm is employed. We are not in a position to make trust judgments with regard to these pages. It is highly desirable to have a well performing algorithm that with a limited seed set enables us to make trust judgments about a large fraction of web pages.

Intuitively, different values for α in Equation 3 or d in "Constant Splitting" and "Logarithm Splitting" will determine how far trust and distrust are propagated. In TrustRank, smaller α means that more trust will be propagated to children pages in each iteration; thus more pages may have nonzero value after 20 iterations. In order to show this, for the same experiment shown in Figure 3, we check what percentage of sites have nonzero values according to different values of α. The results are shown in Table 6.

If more sites have nonzero values by using different choices, then we can claim that the trust scores are propagated further by these choices. Since the results obtained by using "Maximum Share" and "Maximum Parent" in Table 2 are better than TrustRank, we check the percentage of pages with nonzero values for these choices. The results are shown in Table 5.

The results in Table 5 show larger numbers when compared to the results in Table 6. This demonstrates that our choices can affect more pages as well as generate better performance in the demotion of top ranking spam sites.

8. DISCUSSION

In this paper, we investigate the possibility of using different choices to propagate trust and distrust for ranking Web pages or sites. We only focus on the demotion of spam sites. In the future, we intend to study how the propagation of trust or distrust can help raise high quality sites in the ranking positions.

We show that mechanisms such as "Logarithm Splitting" or "Maximum Share" for propagating trust and distrust can do better than TrustRank in demoting top ranked spam sites. We intend to explore other choices that can help improve the performance.

In our paper, we combine trust and distrust scores only at the final step. It is possible that this combination can be done during the calculation of trust and distrust scores. We aim to study the different choices that may be taken into this combination.

Ranking algorithms such as PageRank are used by several popular search engines for ranking Web pages to given queries. The concept of authority and trustworthiness are not identical—PageRank gives an authority value for each page, while propagating trust from seed sets tells how trustworthy a page on the web is as a source of ranking information. In this paper we have only explored the value of trust propagation for spam demotion; ultimately the goal, however, is to improve the quality of search results. We plan to investigate combinations of trust and distrust with authority to measure the effect on search results ranking (quality of results).

All of our experiments are based on the search.ch data set. This data set may have special characteristics different from the whole web. We need to test the ideas presented here on a larger data set, such as the WebBase [16] data set, in the future.

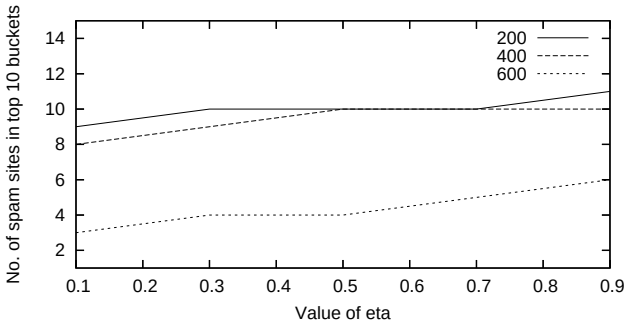

Figure 4: Number of top ranked spam sites when ranking by the combination of trust score and distrust score. Different η and different number of seeds (200, 400, 600) are used.

9. CONCLUSION

In this paper, we show that propagating trust based on the number of outgoing links is not optimal in demoting top ranked spam sites. Instead, we demonstrate that using different choices such as "Constant Splitting" or "Logarithm Splitting" in the splitting step and "Maximum Share" or "Maximum Parent" in the accumulation step for propagating trust can help to demote top ranked spam sites as well as increase the range of trust propagation.

Jump Probability	Percentage of sites with nonzero values
0.9	59.28
0.8	66.72
0.7	70.52
0.6	72.79
0.5	74.07
0.4	74.99
0.3	75.56
0.2	75.91
0.1	76.13

Table 6: Percentage of sites affected when using different jump probabilities.

Additionally, by introducing the concept of propagating distrust among Web pages or sites, we show that the performance of demoting top ranked spam sites can be further improved.

Acknowledgments

This work was supported in part by the National Science Foundation under award IIS-0328825. We are grateful to Urban Müller for helpful discussions and for providing access to the search.ch dataset.

10. REFERENCES

[1] Pr0 - google's pagerank 0, 2002. http://pr.efactory.de/e-pr0.shtml.

[2] A. Acharya, M. Cutts, J. Dean, P. Haahr, M. Henzinger, U. Hoelzle, S. Lawrence, K. Pfleger, O. Sercinoglu, and S. Tong. Information retrieval based on historical data, Mar. 31 2005. US Patent Application number 20050071741.

[3] Z. Bar-Yossef, A. Z. Broder, R. Kumar, and A. Tomkins. Sic transit gloria telae: Towards an understading of the web's decay. In *Proceedings of the Thirteenth International World Wide Web Conference*, New York, May 2004.

[4] A. A. Benczur, K. Csalogany, T. Sarlos, and M. Uher. SpamRank - fully automatic link spam detection. In *Proceedings of the First International Workshop on Adversarial Information Retrieval on the Web (AIRWeb)*, 2005.

[5] G. Collins. Latest search engine spam techniques, Aug. 2004. Online at http://www.sitepoint.com/article/search-engine-spam-techniques.

[6] I. Drost and T. Scheffer. Thwarting the nigritude ultramarine: Learning to identify link spam. In *Proceedings of European Conference on Machine Learning*, pages 96–107, Oct. 2005.

[7] D. Fetterly, M. Manasse, and M. Najork. Spam, damn spam, and statistics: Using statistical analysis to locate spam web pages. In *Proceedings of WebDB*, pages 1–6, June 2004.

[8] D. Fetterly, M. Manasse, and M. Najork. Detecting phrase-level duplication on the world wide web. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research & Development in Information Retrieval*, pages 170–177, Salvador, Brazil, August 2005.

[9] E. Gray, J. Seigneur, Y. Chen, and C. Jensen. Trust propagation in small worlds. In *Proceedings of the First International Conference on Trust Management*, 2003.

[10] R. Guha. Open rating systems. Technical report, Stanford University, 2003.

[11] R. Guha, R. Kumar, P. Raghavan, and A. Tomkins. Propagation of trust and distrust. In *Proceedings of the 13th International World Wide Web Conference*, pages 403–412, New York City, May 2004.

[12] Z. Gyöngyi and H. Garcia-Molina. Web spam taxonomy. In *First International Workshop on Adversarial Information Retrieval on the Web (AIRWeb)*, Chiba, Japan, 2005.

[13] Z. Gyöngyi, H. Garcia-Molina, and J. Pedersen. Combating web spam with TrustRank. In *Proceedings of the 30th International Conference on Very Large Data Bases (VLDB)*, pages 271–279, Toronto, Canada, Sept. 2004.

[14] T. Haveliwala. Topic-sensitive PageRank. In *Proceedings of the Eleventh International World Wide Web Conference*, pages 517–526, Honolulu, Hawaii, May 2002.

[15] M. R. Henzinger, R. Motwani, and C. Silverstein. Challenges in web search engines. *SIGIR Forum*, 36(2):11–22, Fall 2002.

[16] J. Hirai, S. Raghavan, H. Garcia-Molina, and A. Paepcke. WebBase: a repository of Web pages. *Computer Networks*, 33(1–6):277–293, 2000.

[17] G. Jeh and J. Widom. Scaling personalized web search. In *Proceedings of the Twelfth International World Wide Web Conference*, pages 271–279, Budapest, Hungary, May 2003.

[18] S. Kamvar, T. Haveliwala, C. Manning, and G. Golub. Extrapolation methods for accelerating PageRank computations. In *Proceedings of the Twelfth International World Wide Web Conference*, 2003.

[19] P. Massa and C. Hayes. Page-rerank: using trusted links to re-rank authority. In *Proceedings of Web Intelligence Conference*, France, Sept. 2005.

[20] G. Mishne, D. Carmel, and R. Lempel. Blocking blog spam with language model disagreement. In *Proceedings of the First International Workshop on Adversarial Information Retrieval on the Web (AIRWeb)*, 2005.

[21] A. Ntoulas, M. Najork, M. Manasse, and D. Fetterly. Detecting spam web pages through content analysis. In *Proceedings of the 15th International Conference on the World Wide Web*, Edinburgh, Scotland, May 2006.

[22] Open Directory Project, 2005. http://dmoz.org/.

[23] L. Page, S. Brin, R. Motwani, and T. Winograd. The PageRank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project, 1998.

[24] A. Perkins. White paper: The classification of search engine spam, Sept. 2001. Online at http://www.silverdisc.co.uk/articles/spam-classification/.

[25] Räber Information Management GmbH. The Swiss search engine, 2006. http://www.search.ch/.

[26] B. Wu and B. D. Davison. Identifying link farm spam pages. In *Proceedings of the 14th International World Wide Web Conference*, pages 820–829, Chiba, Japan, May 2005.

[27] B. Wu, V. Goel, and B. D. Davison. Topical TrustRank: Using topicality to combat web spam. In *Proceedings of the 15th International World Wide Web Conference*, Edinburgh, Scotland, May 2006.

[28] C.-N. Ziegler and G. Lausen. Spreading activation models for trust propagation. In *Proceedings of the IEEE International Conference on e-Technology, e-Commerce, and e-Service*, Taipei, Taiwan, March 2004. IEEE Computer Society Press.