From Whence Does Your Authority Come? Utilizing Community Relevance in Ranking

Lan Nie and **Brian D. Davison** and **Baoning Wu**
Department of Computer Science & Engineering
Lehigh University
Bethlehem, PA 18015 USA
{lan2,davison,baw4}@cse.lehigh.edu

Abstract

A web page may be relevant to multiple topics; even when nominally on a single topic, the page may attract attention (and thus links) from multiple communities. Instead of indiscriminately summing the authority provided by all pages, we decompose a web page into separate subnodes with respect to each community pointing to it. Utilizing the relevance of such communities allows us to better model the semantic structure of the Web, leading to better estimates of authority for a given query. We apply a total of eighty queries over two real-world datasets to demonstrate that the use of community decomposition can consistently and significantly improve upon PageRank's top-ten results.

Introduction

Web search engines have adopted several different sources of evidence to rank web pages matching a user query, such as textual content and the link structure of the web. The latter is particularly beneficial in helping to address the abundance of relevant documents on the Web by determining authority of a page based on the links that point to it. PageRank (Page *et al.* 1998) and HITS (Kleinberg 1999) are two fundamental link analysis approaches. Put simply, in PageRank the importance of a page depends on the number and quality of pages that link to it. Under the HITS hub and authority model, a page is important if it is linked from hubs that also link to other important pages.

Both of these models treat all hyperlinks equally and assess a page's quality by summing the incoming authority flows indiscriminately. However, hyperlinks are not identical; they may be created in different contexts and represent different opinions. For example, a news website normally contains articles on multiple topics and may have links from different sources. These links convey endorsement in different topics, and mixing them indiscriminately, as traditional link analysis usually does, will hinder an understanding of web page reputations. As a result, a page will be assigned a generic score to tell whether it is good, fair or bad; but we will have no idea whether a popular page is good for "Sports" or "Arts", given it is textually related to both.

We argue that it is more helpful to determine the authority of a resource with respect to some topic or community than in general. To achieve this, we propose a novel ranking model—CommunityRank. By identifying the various communities that link to a resource, CommunityRank differentiates incoming flows so that we can keep track of the contribution from different communities. Such separation avoids the problem of a heavily linked page getting highly ranked for an irrelevant query, thus yielding more accurate search results.

In this paper, we show that by introducing community decomposition and considering community relevance, we can improve the accuracy of PageRank algorithms by as much as 10% relative to the original performance. The experiments are conducted on two real-world web datasets over a total of 80 queries. Our contributions include:

- A description of the use of community decomposition to better model the source of authority and the resulting scores.
- An experimental comparison of this approach to a number of well-known ranking algorithms demonstrating the superiority of our approach.

In the remainder of this paper we will introduce related work, followed by our CommunityRank model. The experimental framework and results will then be presented, after which we conclude with a discussion and future work.

Related work

Topicality in link analysis. Researchers have proposed a number of different approaches of combining topical information with link analysis.

Haveliwala's Topic-Sensitive PageRank (TSPR) (Haveliwala 2002) is the first published algorithm to incorporate topical information into link analysis. Pal and Narayan (2005) utilize similar biasing when following a link: surfers will prefer pages on the same topic as the current page. Richardson and Domingos' Intelligent Surfer (2002) adopts the idea of selecting links (and jump targets) using a probability distribution based on the relevance of the target to the surfer's query. Nie *et al.* (2006) also propose bringing topicality into web rankings. Their approach distributes a page's authority across topics through the use of a topical random surfer. Compared to the above, we also utilize topics to generate communities in this paper, but our approach is not limited to using topics when generating communities.

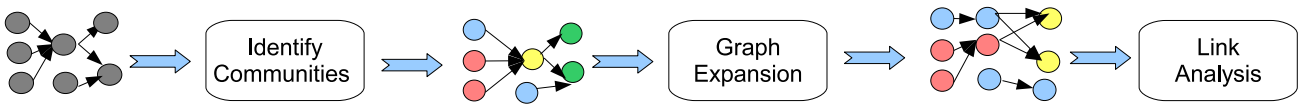

Figure 1: The CommunityRank process.

Other work has considered breaking a page into subpages. Chakrabarti *et al.* (1999) proposed segmenting a page into small pagelets containing contiguous subsets of links consistently on a particular topic. Cai *et al.* (2004) use a vision based algorithm to partition a page into blocks. In contrast, in this work we replicate a page based on various incoming conceptual communities rather than breaking it apart based on the page's structural layout.

Link analysis for community discovery. The process of discovering communities on the web has also been studied. Kumar *et al.* (1999) utilized bipartite subgraphs and cores to find web communities. Flake *et al.* (2000) proposed using a maximum flow and minimum cut method to identify these communities. Andersen and Lang (2006) utilized a local random walk model to detect more pages from a seed set of pages within the same community. The above researchers utilize link analysis as a tool to detect communities on the web. In contrast, our proposed approach employs the community information to help link analysis ranking algorithms.

Roberts and Rosenthal (2003) propose a simple algorithm to find page clusters based on their outlink sets, and the authority value of a page is proportional to the number of clusters which tend to link to it rather than the number of pages which link to it. This is somewhat similar in spirit to our approach, but we consider the pages' content when generating communities, and do not value each community equally.

The CommunityRank Model

A web page may be relevant to multiple topics; even when on a single topic, the page may attract attention from multiple communities. Instead of indiscriminately summing the authority provided by each community, it may be worthwhile to keep them apart. To achieve this, we decompose a web page into several community-specific subnodes so that authority from a particular community only flows into the corresponding subnode. The re-organized hyperlink structure gives a more accurate and robust representation of the relationship of the Web, thus preventing a resource that is highly popular for one topic (e.g., community A) from dominating the results of another topic (community C) in which it is less authoritative. This process is depicted in Figure 1, and described below.

Identify communities. In our approach, a community is defined as a group of parents to a page that share similar concerns. We argue that a recommendation is conveyed via a hyperlink within some context. By representing that context, the task of community identification is mapped into clustering the contexts of recommendations to a common page. There are various options to represent a hyperlink's context. In this paper, we will consider two possibilities:

- **fulltext:** the full contents of the document in which the hyperlink appears, or
- **anchortext:** the anchortext of the hyperlink.

Other options (not explored here) could include contents surrounding the link's occurrences in that page, topic distribution, web graph structure, page layout and so on.

The next step is to group these contexts into clusters based on their similarity. Clustering can be an expensive process, and in our model needs to be applied to the set of parents for each document in the collection. In our current implementation, we adopt a simplification: we predefine twelve broad categories, chosen from the top level of the dmoz Open Directory Project (ODP) (ODP 2006), and use a textual classifier to determine the category of each context. In this way, the contexts of hyperlinks to a given node are categorized into several communities based on their classification labels.

Building the graph model. With the community disambiguation process in place, the next step is to split pages into community-specific subnodes and set up the network among them. To understand how to construct this graph model, we take an initial look at a small web made up of seven pages in Figure 2. In the original web (shown in the left part), node A is linked from two relatively distinct communities, and that community X represents 33% of the links and Y, 66%. As we suggested, node A is split into two independent subnodes, e.g. A_X and A_Y, as the right part of Figure 2 shows. A similar process is applied to every node in the web.

The next question is how to map the link structure present among the original nodes to the community-specific subnodes. For the purpose of separating authority flows on different topics, incoming links are distributed among subnodes such that each subnode only gets links from a single community. In contrast, the outgoing links are duplicated across all subnodes so that the total authority flow passed on to the original node's children remains approximately the same as before splitting into subnodes. As shown in the right part of Figure 2, links from community X and community Y are separately directed to A_X and A_Y, respectively, while the outgoing link $A \rightarrow E$ is replicated in each subnode: $A_X \rightarrow E$ and $A_Y \rightarrow E$.

Link analysis. Popular link analysis algorithms can be put in two categories: "PageRank" based ranking schemes and "hub and authority" based ranking scheme. In this section, we introduce how to apply each of them to the graph model presented above.

The PageRank algorithm presents a so-called "random surfer" model, where a web surfer on a given page may either with probability d follow an outgoing link of the current page chosen uniformly at random, or otherwise jump to a randomly selected page in the Web.

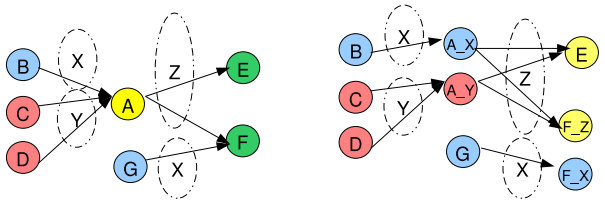

Figure 2: Splitting the web graph.

An intuition behind PageRank is that a high-valued page p is also likely to be a good source containing links to other good pages. The hub and authority model (Kleinberg 1999; Rafiei & Mendelzon 2000) presents a different viewpoint: a good page p does not mean that itself is a good source of further links; but it does mean that pages that point to p may be good sources, since they already led to page p.

When the hub and authority model is extended to include a random surfer (Rafiei & Mendelzon 2000; Ng, Zheng, & Jordan 2001), the surfer's behavior is a bit more involved. Here we follow a non-term-specific variation of Rafiei & Mendelzon, and so when the current page is p, the surfer either jumps to a randomly selected page, or randomly picks any page q that has a link into page p and makes a forward transition out of page q. In this model, the surfer can follow links in both forward and backward direction. The probability that page p is "forward" visited by the random surfer is defined as its authority $A(p)$ and the probability to be "backward" visited is hub score $H(p)$, which can be formulated as

$$\begin{aligned} H(q) &= d \sum_{p:q\to p} \frac{A(p)}{I(p)} + (1-d)\frac{1}{N} \\ A(p) &= d \sum_{q:q\to p} \frac{H(q)}{O(q)} + (1-d)\frac{1}{N} \end{aligned} \quad (1)$$

where $I(p)$ and $O(p)$ are page p's indegree and outdegree, N is the graph's size. As in the original HITS scheme, hubs and authorities interact with each other. The main difference is that a node's authority (hubness) will be distributed among its parents (children) instead of entirely copied to each parent (child). In addition, we include a random jump transition. By incorporating these features, this HITS-like propagation, denoted as Global HITS (*GHITS*), can be applied to the full web graph, not just a query-specific subgraph.

We can directly apply PageRank or Global HITS onto our community-based graph and denote such combinations as Community-PageRank (*CPR*) and Community-HITS (*CHITS*) seperately.

Query-time ranking. After authority calculations are complete, every subnode will have a traditional authority score based on its associated community. The next task is performed at query time: to be able to rank results for a particular query q, we need to calculate a query-specific importance score for each web page. This can be achieved by summing the scores of subnodes that belong to a page weighted by their affinity to this query. We consider two representations for each community: either by the category label shared by community members, or by its content centroid. Correspondingly, the relevance between a query and a community can be calculated in two ways:

- **category-level relevance** using a textual classifier to generate a probability distribution for a query j across the predefined categories, in which the i^{th} component represents q's relative relevance to category i.
- **term-level relevance** textual similarity between a community's centroid and a query, with both in the form of term-vectors.

We consider both forms in our experimental work.

Experimental Setup

The main goal of the proposed CommunityRank is to improve the quality of web research. Hence, we compare the retrieval performance of well-known ranking algorithms versus the proposed CommunityRank approaches.

Datasets. To avoid a corpus bias, we used two different data collections in our experiments. One is the TREC[1] GOV collection, a 1.25M Web pages crawl of the .gov domain from 2002. The second data set is a 2005 crawl from the Stanford WebBase (Cho *et al.* 2006), containing 57.7M pages and approximately 900M links.

To test various ranking algorithms on the GOV corpus, we chose the topic distillation task in the web track of TREC 2003, which contains 50 queries. For experiments on WebBase, we selected 30 queries (shown in Table 1) from those frequently used by previous researchers, ODP category names, and popular queries from Lycos and Google.

Evaluation. Since there is no standard evaluation benchmark for the WebBase dataset, the relevance between query and search results has to be inspected manually. In our evaluation system, the top ten search results generated by various ranking algorithms were mixed together. For each randomly selected query and URL pair, subjects (a total of 7 participants) were asked to rate the relevance as quite relevant, relevant, not sure, not relevant, and totally irrelevant, to which we assigned the scores of 4, 3, 2, 1, 0, respectively. We used two metrics to evaluate the performance. The first is Precision at 10 (P@10), which reports the fraction of "good" documents ranked in the top 10 results. In our setting, a result is marked as "good" if its average human judgment

[1] http://trec.nist.gov/

harry potter	college football	diabetes
music lyrics	george bush	automobile warranty
online dictionary	britney spear	herpes treatments
olsen twins	diamond bracelet	madonna
weight watchers	windshield wiper	brad pitt
playstation	jennifer lopez	the passion of christ
new york fireworks	lord of the rings	poker
halloween costumes	iraq war	tsunami
games	poems	musculoskeletal disorders
tattoos	jersey girl	st patricks day cards

Table 1: Set of thirty queries used for relevance evaluation in WebBase.

Method	Link Context	Community Relevance	Propagation
CPR_FC	Fulltext	Category	CPR
CPR_AC	Anchortext	Category	CPR
CPR_FT	Fulltext	Term	CPR
CPR_AT	Anchortext	Term	CPR
CHITS_FC	Fulltext	Category	CHITS
CHITS_AC	Anchortext	Category	CHITS
CHITS_FT	Fulltext	Term	CHITS
CHITS_AT	Anchortext	Term	CHITS

Table 2: Different configurations of the CommunityRank model.

score is above 2.5. To further explore the quality of retrieval, we also evaluated the ranking algorithms over the Normalized Discounted Cumulative Gain (NDCG) (Jarvelin & Kekalainen 2000) metric. NDCG credits systems with high precision at top ranks by weighting relevant documents according to their rankings in the returned search results; this characteristic is crucial in web search. We denote the NDCG score for the top 10 ranked results as NDCG@10.

For GOV data, TREC provides relevance judgments for performance evaluation. There are 10.32 relevant documents per query on average for the topic distillation task of TREC 2003. In addition to P@10 and NDCG@10, we add Mean Average Precision (MAP) and Rprec as evaluation metrics since they are widely used in TREC.

Ranking methods compared. We compare five ranking algorithms to our proposed approach: BM2500 (Robertson 1997), PageRank (PR), Global HITS (GHITS), Topic-Sensitive PageRank (TSPR) and Intelligent Surfer (IS). BM2500, PR and GHITS are used as baselines; we additionally chose TSPR and IS because, similar to our model, they measure a page's reputation with respect to different aspects (topic or term) instead of mixing them together.

As discussed previously, the CommunityRank model may have several options; the resulting different combinations are shown in Table 2. In the experimental section below, we will study and compare their performances.

We rank all documents using a combination of the query-specific IR score and the authority score generated by link analysis approaches. The IR score is calculated using the OKAPI BM2500 (Robertson 1997) weighting function, and the parameters are set the same as Cai *et al.* (2004). The combination can be score-based, where a page's final score is a weighted summation of its authority score and IR score; it also can be order-based, where weighted ranking positions based on importance score and relevance score are summed together. In our implementation, we choose the order-based option. All ranking results presented in this paper are already combined with IR scores.

Textual classification. We use a well-known naive Bayes classifier, "Rainbow" (McCallum 1996), to decide the category for each hyperlink's context for the purpose of community recognition. The classifier is trained on 19,000 pages from each of twelve categories of the ODP hierarchy.

Method	NDCG@10	P@10	MAP	Rprec
BM2500	0.199	0.120	0.149	0.140
PR	0.218	0.138	0.153	0.153
GHITS	0.204	0.136	0.143	0.154
CPR_FC	0.240	0.148	0.168	0.184
CPR_AC	0.231	0.140	0.165	0.167
CPR_FT	0.210	0.134	0.148	0.165
CPR_AT	0.219	0.130	0.159	0.162
CHITS_FC	0.241	0.144	0.173	0.184
CHITS_AC	0.218	0.142	0.155	0.169
CHITS_FT	0.215	0.132	0.160	0.160
CHITS_AT	0.210	0.126	0.159	0.163

Table 3: Performance on GOV.

Experimental Results

Community decomposition and graph expansion. Through community decomposition, each node in the web graph is split into subnodes with respect to the different communities linking to it. As a result, the 1.25 million pages in GOV are decomposed into 1.43 million community-specific subnodes and the 57.7 million nodes on WebBase are divided into 69.3 million subnodes (here we used "fulltext" to represent contexts in these calculations). The "community indegree", or number of communities pointing to a page, is 1.3-1.4 on average. The indegree distribution is shown in Figure 3. The number of communities linking to a document is correlated with the number of pages pointing to it (with coefficients of 0.29 and 0.27 for GOV and WebBase, respectively).

Many pages on the web cover topics from different communities. For example, in the GOV dataset, the page http://dc.gov/ is the government homepage of District of Columbia. We found parent pages from various communities, such as "Recreation", "Sports", "Business", "Health" and "Computers", pointing to it. For another example, the page http://www.tourism.wa.gov/ is the official site of Washington's state tourism. Its parents were categorized into either "Recreation" or "Business" communities.

Results on GOV. Three approaches, BM2500, PageRank and GHITS, are chosen as baselines. Their performance on four evaluation metrics are shown in the first three lines of Table 3. Recall that the CommunityRank model may have different settings. Table 3 investigates which policy is opti-

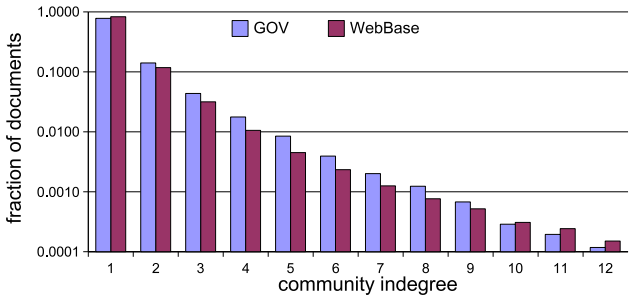

Figure 3: Distribution of community indegree.

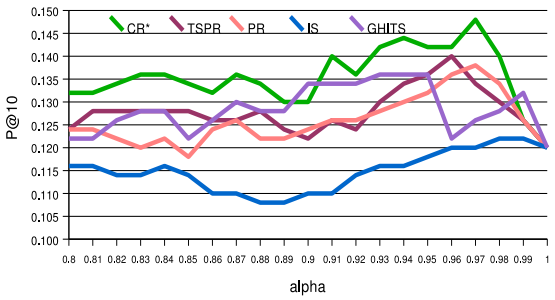

Figure 4: Combination of IR and importance scores on GOV.

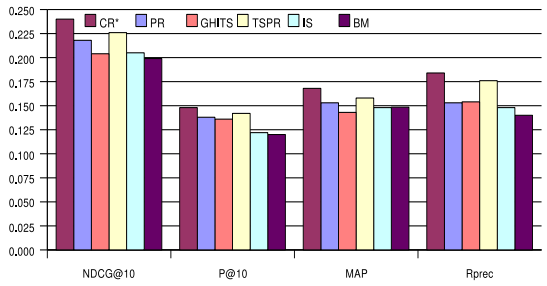

Figure 5: Comparison of overall performance for GOV.

mal. As can be seen, *CPR_FC* get the best performance on P@10. Methods shown in bold outperform the three baselines. The other four approaches, which use "Term-level" rather than "Category-level" to measure query relevance, fail on one or more metrics when comparing to PageRank or GHITS. We also observe that "Fulltext" representation is slightly better than "Anchortext" representation while "CHITS" propagation is similar to "CPR" propagation.

In the following experiments, we compare the winner of our model *CPR_FC*, denoted as *CR** here, with the other four rankers: PageRank, Global HITS, Topic-Sensitive PageRank and Intelligent Surfer.

We first conduct a parameter study to investigate how different weights for importance and relevance scores will affect ranking systems' performance. Figure 4 shows the precision@10 as α is varied for the four ranking approaches, where α is the weight of BM2500 score in the combination. As can be seen, CR* curve is almost always equal to or above other curves in the graph, showing that our approach generally outperforms other approaches. All curves converged to the baseline when α is 1, which corresponds to the performance of BM2500. In GOV dataset, for each approach, we tune the combining parameter for the best P@10 and output its results with this optimal combination as final results. In contrast, for experiments on WebBase, we fix the weight of IR score as 0.8 to save the cost of manual evaluation across different values of α.

Figure 5 shows the overall performance comparison. CR* outperforms other approaches on all metrics. An observation is that IS does not work well on TREC data, as it performs even more poorly than PageRank. To determine whether these improvements are statistically significant, we calculated several single-tailed t-tests to compare CR* with all other approaches. As Table 4 shows, CR* significantly exceeds the other approaches at a 95% confidence level on both metrics, except for TSPR.

Results on WebBase. Performance on the BM2500, PageRank and GHITS baselines using NDCG@10 and P@10 can be found in the top three rows of Table 5. This table also lists the performances of different community rankers on WebBase. All the results are better than the baseline performances. We note that *CPR_FT* achieves the best performance by outperforming PageRank by 5% on NDCG@10 and 5.3% on P@10; we denote it as *CR** for further comparisons. In contrast to the results presented for GOV, "Term-Relevance" outperforms "Category-Relevance" on WebBase.

Method	NDCG@10	P@10
BM2500	0.431	0.553
PR	0.445	0.560
GHITS	0.463	0.563
CPR_FC	0.493	0.607
CPR_AC	0.496	0.593
CPR_FT	0.500	0.613
CPR_AT	0.500	0.610
CHITS_FC	0.492	0.583
CHITS_AC	0.480	0.573
CHITS_FT	0.501	0.610
CHITS_AT	0.494	0.603

Table 5: Performance on WebBase.

Figure 6 shows the overall performance comparison. Intelligent surfer and CR* lead the competition with P@10 of 62.0% and 61.3%, NDCG@10 of 0.497 and 0.500 respectively. Interestingly, different from CR*'s consistent superiority on GOV and WebBase, Intelligent Surfer shows drastically different performance on the two datasets, from the worst to the best.

Again we performed t-tests to compare CR* to the other approaches. As Table 6 shows, CR* significantly outperforms BM2500, PR, GHITS, TSPR (on NDCG@10) at 90% or better confidence level, CR* and intelligent surfer are statistically indistinguishable.

Discussion. From the experiments shown above, we learn

Metric	PR	GHITS	IS	TSPR	BM2500
P@10	0.025	0.034	0.013	0.082	0.007
NDCG@10	0.024	0.002	0.024	0.143	0.007

Table 4: P-values for the t-test on GOV.

Metric	PR	GHITS	IS	TSPR	BM2500
P@10	0.013	0.008	0.395	0.291	0.006
NDCG@10	0.002	0.07	0.489	0.077	0.0005

Table 6: P-values for the t-test on WebBase.

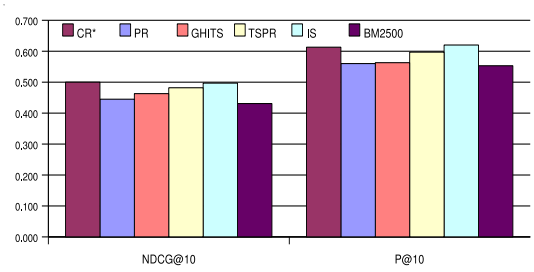

Figure 6: Comparison of overall performance for WebBase.

that the "Anchortext" representation does not work as well as the "Fulltext" representation for WebBase. One possible reason is that anchortext representation is short and generally similar across links to a given page, and thus less informative. "CHITS" propagation is similar to "CPR" propagation in performance. In contrast, term level relevance measurement outperforms category level measurement on WebBase, but fails on GOV. Intuitively, there are different policies dealing with *narrow* and *broad* queries. On one hand, we find the need to generalize narrow queries, like those in GOV having only 10 relevant documents, from the term-level to category-level to include more potential candidates; on the other hand, with broad queries like those we used in WebBase that have plenty of relevant results, we focus on the term-level relevancy to refine the search.

Intelligent surfer exhibits quite poor performance on GOV dataset. Since intelligent surfer only wanders within a term-specific subgraph, given a small Web like GOV, the subgraph is less likely to be well-connected and applicable to link analysis. Based on our statistics, the average density of links per page of term-specific subgraphs in GOV (for terms of 50 queries) is 3.11 versus 16.5 in WebBase. In addition, intelligent surfer has significant time and space overhead since it needs to generate term-specific rankings for all terms in advance; on the contrary, the CommunityRank model only needs to be calculated once while achieving matching performance on WebBase and better results on GOV.

Conclusion

In this paper we propose a novel community ranking algorithm which decomposes the normal web graph into community-based subnode graph. Experimental results on two real datasets indicate that our approach consistently improves search engines' ranking performance. In the future, we expect to further study different choices for clustering, the effects of link weights, and to apply this model on query-specific datasets. We would also like to consider how to describe the reputation of a page within the communities in which it is found.

Acknowledgments. This work was supported in part by grants from Microsoft Live Labs ("Accelerating Search") and the National Science Foundation under CAREER award IIS-0545875. We also thank TREC and Stanford University for access to their web collections.

References

Andersen, R., and Lang, K. J. 2006. Communities from seed sets. In *Proceedings of the 15th International World Wide Web Conference*, 223–232.

Cai, D.; He, X.; Wen, J.-R.; and Ma, W.-Y. 2004. Block-level link analysis. In *Proceedings of the 27th ACM SIGIR Conference on Research and Development in Information Retrieval*.

Chakrabarti, S.; Dom, B. E.; Gibson, D.; Kleinberg, J. M.; Kumar, S. R.; Raghavan, P.; Rajagopalan, S.; and Tomkins, A. 1999. Mining the Web's link structure. *IEEE Computer* 60–67.

Cho, J.; Garcia-Molina, H.; Haveliwala, T.; Lam, W.; Paepcke, A.; Raghavan, S.; and Wesley, G. 2006. Stanford WebBase components and applications. *ACM Transactions on Internet Technology* 6(2):153–186.

Flake, G. W.; Lawrence, S.; and Giles, C. L. 2000. Efficient identification of web communities. In *Proceedings of the Sixth ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD-2000)*, 150–160.

Haveliwala, T. H. 2002. Topic-sensitive PageRank. In *Proceedings of the Eleventh International World Wide Web Conference*.

Jarvelin, K., and Kekalainen, J. 2000. IR evaluation methods for retrieving highly relevant documents. In *Proceedings of the 23rd ACM SIGIR Conference on Research and Development in Information Retrieval*.

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

Kumar, R.; Raghavan, P.; Rajagopalan, S.; and Tomkins, A. 1999. Trawling the Web for emerging cyber-communities. *Computer Networks* 31(11–16):1481–1493.

McCallum, A. K. 1996. Bow: A toolkit for statistical language modeling, text retrieval, classification and clustering. http://www.cs.cmu.edu/˜mccallum/bow.

Ng, A. Y.; Zheng, A. X.; and Jordan, M. I. 2001. Stable algorithms for link analysis. In *Proceedings of the 24th ACM SIGIR Conference on Research and Development in Information Retrieval*, 258–266.

Nie, L.; Davison, B. D.; and Qi, X. 2006. Topical link analysis for web search. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research & Development on Information Retrieval*.

2006. Open Directory Project (ODP). `http://www.dmoz.com/`.

Page, L.; Brin, S.; Motwani, R.; and Winograd, T. 1998. The PageRank citation ranking: Bringing order to the Web. Unpublished draft.

Pal, S. K., and Narayan, B. L. 2005. A web surfer model incorporating topic continuity. *IEEE Transactions on Knowledge and Data Engineering* 17:726–729.

Rafiei, D., and Mendelzon, A. O. 2000. What is this page known for? Computing web page reputations. In *Proceedings of the Ninth International World Wide Web Conference*.

Richardson, M., and Domingos, P. 2002. The Intelligent Surfer: Probabilistic Combination of Link and Content Information in PageRank. In *Advances in Neural Information Processing Systems 14*. MIT Press.

Roberts, G. O., and Rosenthal, J. S. 2003. Downweighting tightly knit communities in world wide web rankings. *Advances and Applications in Statistics* 3(3):199–216.

Robertson, S. E. 1997. Overview of the OKAPI projects. *Journal of Documentation* 53:3–7.