Link Analysis Ranking

by

Panayiotis Tsaparas

A thesis submitted in conformity with the requirements
for the degree of Doctor of Philosophy
Graduate Department of Computer Science
University of Toronto

Link Analysis Ranking

Panayiotis Tsaparas

Doctor of Philosophy

Department of Computer Science

University of Toronto 2004

Abstract

The explosive growth and the widespread accessibility of the Web has led to surge of research activity in the area of information retrieval on the World Wide Web. Ranking has always been an important component of any information retrieval system. In the case of Web search its importance becomes critical. Due to the size of the Web, it is imperative to have ranking functions that capture the user needs. To this end the Web offers a rich context of information which is expressed through the hyperlinks. In this thesis we investigate, theoretically and experimentally, the application of *Link Analysis* to ranking on the Web.

Building upon the framework of hubs and authorities [57], we propose new families of Link Analysis Ranking algorithms. Some of the algorithms we define no longer enjoy the linearity property of the previous algorithms. As a result, it is harder to analyze them, or even prove that they actually converge. However, for a special case of the families we consider, we are able to prove that it will converge, and we provide a complete characterization of the combinatorial properties of the stationary authority weights it produces.

The plethora of Link Analysis Ranking algorithm, generates the necessity for a formal way to evaluate their properties and compare their behavior. We introduce a theoretical framework for the study of Link Analysis Ranking algorithms, and we define specific properties of the algorithms within this framework. Using these properties we are able to provide an axiomatic characterization of the InDegree algorithm that ranks pages according the number of in-coming links.

We conclude the thesis with an extensive experimental evaluation of Link Analysis Ranking. We test the algorithms over multiple queries, and we use user feedback to determine their quality. Our experiments reveal some of the limitations of Link Analysis Ranking. Specifically, it appears that for most algorithms, the nodes and the structures in the graph that they favor, do not correspond to the most relevant pages in the collection. These observations offer a new insight into the mechanics of the algorithms, and we believe that they will lead to improved algorithm design, and better input graphs for the algorithms.

Contents

List of Figures

Chapter 1

Introduction

1.1 Motivation

Searching on the World Wide Web is the second most frequent operation on the Web after e-mail [14]. Therefore, it is important to have tools that perform search efficiently and effectively. Recently, much research has been devoted to creating better search engines for the Web. Even though there is a rich literature in the area of information retrieval [5], the Web, due to its size, and the diversity of the users that perform search, poses new challenges and problems. In this thesis, we concentrate on the problem of *ranking*.

Ranking is an integral component of any information retrieval system. In the context of the World Wide Web the role of ranking becomes even more important. A query on the Web can have thousands, or even millions, of relevant results. If the ranking function does not output what the user is looking for within the top few positions of the ranking, the search engine is rendered useless. Web users do not have the patience to go through hundreds, or thousands of pages to find the one they are looking for. It has been documented that most Web users do not even look "below the fold", that is, below the first screen of results [14, 84, 52]. In this setting, the quality of the ranking function becomes critical.

Furthermore, the needs of the users when querying the Web are different. For example, a user that poses the query "microsoft" to a Web search engine is most likely looking for the home page of Microsoft Corporation, rather than the page of some random user that rants about the bugs in Microsoft products. In traditional information retrieval, this random page may be highly relevant to the query. However, Web users are not so much interested in finding relevant pages, as much as finding *authoritative* pages. The definition of an authoritative page appeals to the intuition of the Web user, and it can be roughly described as a page that is not only relevant to the query, but it is also a *trusted* source of correct information. In Web search the focus shifts from *relevance* to *authoritativeness.* The task of the ranking function becomes to identify and rank high the authoritative documents within a collection of Web pages.

To this end, the Web itself offers a rich context of information which is expressed through the hyperlinks. A Web page is part of a bigger picture that is defined through the pages that can reach, or can be reached by this page. These pages define the *context* in which the page appears. If we remove the page from this context, and treat it as a flat text file, we discard valuable information about the content and the quality of the page. This is why, currently, most commercial search engines make use, to some extent, of the hyperlink information.

There are two popular ways of using the hyperlink information. The first is explicit. The anchor text of the hyperlink, or the text that surrounds the hyperlink is often highly descriptive about the content of the Web page. In this thesis, we do not consider this use of hyperlinks. The second is implicit. If Web page q points to Web page p, then we may assume that page q endorses and recommends the content of page p. Therefore, we can think of the Web as a network of recommendations which contains information about the authoritativeness of the pages. Our task is to extract this latent information, and rank the pages according to their authoritativeness. We call the algorithms that rely on hyperlink information for deriving a ranking, *Link Analysis Ranking (LAR)* algorithms.

1.2 Link Analysis Ranking Algorithms

A Link Analysis Ranking algorithm starts with a collection of Web pages to be ranked. The algorithm then proceeds to extracting the hyperlinks between the pages, and constructing the underlying hyperlink graph. The hyperlink graph is constructed by creating a node for every Web page, and a *directed* edge for every hyperlink between two pages. The graph is given as input to the Link Analysis Ranking algorithm. The algorithm operates on the graph, and produces a *weight* for each Web page. This weight captures the authoritativeness of the page, and it is used to rank the pages. Our task is to devise Link Analysis Ranking algorithms that best discover the authoritative nodes in the graph.

Link Analysis Ranking can be traced back to two seminal papers by Brin and Page [13], and Kleinberg [58]. These two papers changed the way that people think about the Web, and spawned the research area of Link Analysis Ranking. They were followed by a substantial amount of research work [4, 1, 8, 10, 64, 78, 73]. The PAGERANK algorithm, introduced by Brin and Page, later became a commercial success story as an integral component of Google[1], the dominant search engine on the Web at this time.

Kleinberg [58] introduced the hubs and authorities paradigm. In this framework, every page is associated with a *hub* and an *authority* weight. Kleinberg defined the authority weight of a page to be the sum of the hub weights of the pages that point to this page, and the hub weight to be the sum of the authority weights of the pages that are pointed to by this page. He proposed HITS (Hyperlink Induced Topic Distillation), an iterative algorithm for computing the weights. The HITS algorithm has two implicit properties. The first is *symmetry*. Both hub and authority weights are computed in the same way; authority weights are the sum of the hub weights, while hub weights are the sum of authority weights. The second is *equality*. When computing the hub weights (resp. authority weights), the sum operator treats all authority (resp. hub) weights equally. However, there are cases where these two properties of the HITS algorithm lead to non-intuitive results. In this thesis, we deviate from these two properties, and we examine alternative ways of defining the authority and hub weights. Our definitions produce new families of LAR algorithms, with interesting properties.

Following the HITS and PAGERANK algorithms, a large number of modifications [8, 10, 64, 78], extensions [1, 73], as well as novel algorithmic approaches [10, 19, 48] were proposed. A Web practitioner now has a wide range of Link Analysis Ranking algorithms to choose from. Therefore, it is important to be able to compare different LAR algorithms, and evaluate their properties. Typically, this is done experimentally. However, given the subjective nature of the problem, and the inherent limitations of experimental evaluation, it becomes obvious that we need a formal way to study LAR algorithms. Ideally, we would like to identify a set of properties that characterize every LAR algorithm. Then, the practitioner can select the LAR algorithm with the desired properties. In this thesis, we initiate a theoretical analysis of LAR algorithms. We first introduce a framework for the evaluation and comparison of LAR algorithms. We then define several intuitive properties within this

[1] http://www.google.com

framework, and we study the behavior of various LAR algorithms. The framework sets the foundation for a theoretical analysis of LAR algorithms. The long term goal is to have a formal way of assessing the properties of LAR algorithms.

1.3 Contributions and guided tour of the thesis

The remainder of the thesis is structured as follows.

- In Chapter 2 we review the related literature on Link Analysis Ranking algorithms. We also present the necessary tools from linear algebra that will be used throughout the thesis.

- Chapter 3 introduces new Link Analysis Ranking algorithms. We modify the definition of the hub and authority weights provided by Kleinberg. As a result, we produce new families of algorithms, with new properties.

- Chapter 4 examines the application of non-linear dynamical systems to ranking. Some of the algorithms introduced in Chapter3 can no longer be reduced to an eigenvector computation since they apply non-linear operators. When departing from the comfortable world of linear algebra, even the proof of convergence of the algorithms is no longer a simple task. In Chapter 4 we analyze in detail the Max algorithm, a member of the family of the algorithms that we defined. We prove that the algorithm converges. Furthermore, we describe rigorously the combinatorial properties of this algorithm. The study reveals a well-defined mechanism for distributing the authority weights, and gives a clear characterization of the Max algorithm.

- In Chapter 5 we define a theoretical framework for comparing and analyzing LAR algorithms. We define different notions of distance, and similarity between LAR algorithms. We also define properties such as monotonicity, stability, locality, and label independence. Our framework allows for the axiomatic characterization of the InDegree algorithm, the algorithm that ranks pages according to the number of incoming links.

- In Chapter 6 we present experiments on multiple queries, using the proposed algorithms. We also present a comparative evaluation of different LAR algorithms. The experiments provide significant insight into the behavior of the different algorithms, and the types of nodes that they tend to favor in their rankings. They also reveal some of the limitations of Link Analysis Ranking. We also examine the application of LAR algorithms to the problem of finding related pages to a query Web page.

- Chapter 7 concludes the thesis with a summary of the results, and a discussion on possible future directions.

The material in Chapters 3 and 5 was first introduced in the collaborative work with Allan Borodin, Gareth Roberts, and Jeffrey Rosenthal [10], and it was later expanded in the journal version of this paper [11].

Chapter 2

Background and Previous Work

In this chapter we present the necessary background for the rest of the thesis. We also review the literature in the area of link analysis ranking upon which this work builds.

2.1 Preliminaries

A link analysis ranking algorithm starts with a set of Web pages. Depending on how this set of pages is obtained we distinguish between *query independent* algorithms, and *query dependent* algorithms. In the former case, the algorithm ranks the whole Web. The PageRank algorithm [13] was proposed as a query independent algorithm that produces a *PageRank* value for all Web pages. In the latter case, the algorithm ranks a subset of Web pages that is associated with the query at hand. Kleinberg [58] describes how to obtain such a query dependent subset. Using a text-based Web search engine a *Root Set* is retrieved consisting of a short list of Web pages relevant to a given query. Then, the Root Set is augmented by pages which point to pages in the Root Set, and also pages which are pointed to by pages in the Root Set, to obtain a larger *Base Set* of Web pages. This is the query dependent subset of Web pages on which the algorithm operates.

Given the set of Web pages, the next step is to construct the underlying hyperlink graph. A node is created for every Web page, and a *directed* edge is placed between two nodes if there is a hyperlink between the corresponding Web pages. The graph is *simple*. Even if there are multiple links between two pages, only a single edge is placed. No self-loops are allowed. The edges could be weighted using, for example, content analysis of the Web pages, similar to the spirit of the work of Bharat and Henzinger [8]. In our work we will assume that no weights are associated with the edges of the graph. Usually, links within the same Web site are removed since they do not convey an endorsement; they serve the purpose of navigation. Isolated nodes are removed from the graph.

Let P denote the resulting set of nodes, and let n be the size of the set P. Let $G = (P, E)$ denote the underlying graph. The input to the link analysis algorithm is the adjacency matrix W of the graph G, where $W[i, j] = 1$ if there is a link from node i to node j, and zero otherwise. The output of the algorithm is an n-dimensional vector $\boldsymbol{a}$, where a_i, the i-th coordinate of the vector $\boldsymbol{a}$, is the authority weight of node i in the graph. When convenient we may use $a(i)$ instead of a_i to denote the authority weight of node i. These weights are used to rank the pages.

We also introduce the following notation. For some node i, we denote by $B(i) = \{j : W[j, i] = 1\}$ the set of nodes that point to node i (Backwards links), and by $F(i) = \{j : W[i, j] = 1\}$ the set of nodes that are pointed to by node i (Forward links). Furthermore, we define an *authority node* in the graph G to be a node with non-zero in-degree, and a *hub node* in the graph G to be a node with non-zero out degree. We use A to denote the set of authority nodes, and H to denote the set of hub nodes. We have that $P = A \cup H$. We define the undirected *authority* graph $G_a = (A, E_a)$ on the set of authorities A, where we place an edge between two authorities i and j, if $B(i) \cap B(j) \neq \emptyset$. This corresponds to the (unweighted) graph defined by the matrix $W^T W$.

2.2 Previous Algorithms

In this section we describe some of the previous link analysis ranking algorithms that we will consider in this work.

2.2.1 The InDegree algorithm

A simple heuristic that can be viewed as the predecessor of link analysis ranking is to rank the pages according to their *popularity* (often also referred to as *visibility* [69]). The popularity of a page is measured by the number of pages that link to this page. We refer to this algorithm as the InDegree algorithm, since it ranks pages according to their in-degree in the graph G. That is, for every node i,

$$a_i = \frac{|B(i)|}{|E|} \ .$$

This simple heuristic was applied by several search engines in the early days of Web search [69]. Kleinberg [58] makes a convincing argument that the InDegree algorithm is not sophisticated enough to capture the authoritativeness of a node, even when restricted to a query dependent subset of the Web.

2.2.2 The PageRank Algorithm

The intuition underlying the InDegree algorithm is that a good authority is a page that is pointed to by many nodes in the graph G. Brin and Page [13] extended this idea further by observing that not all links carry the same weight. Links from pages of high quality should confer more authority. It is not only important how many pages point to a page, but also what is the quality of these pages. Therefore, they propose a one-level weight propagation scheme, where a good authority is one that is pointed to by many good authorities. They employ this idea in the PageRank algorithm. The PageRank algorithm performs a random walk on the graph G that simulates the behavior of a "random surfer". The surfer starts from some node chosen according to some distribution $\mathcal{D}$ (usually assumed to be the uniform distribution). At each step, the surfer proceeds as follows: with probability $1 - \epsilon$ an outgoing links is picked uniformly at random, and the surfer moves to a new page, and with probability ϵ the surfer jumps to a random page chosen according to distribution $\mathcal{D}$. The "jump probability" ϵ is passed as a parameter to the algorithm. The authority weight a_i of node i (called the PageRank of node i) is the fraction of time that the surfer spends at node i, that is, it is proportional to the number of visits to node i during the random walk.

Formally, assume for a moment that every node i has at least one out-going link. Then the PageRank of node i is given by the formula

$$a_i = \epsilon \mathcal{D}(i) + (1 - \epsilon) \sum_{j \in B(i)} \frac{a_j}{|F(j)|} \ . \tag{2.1}$$

Let W_r denote the matrix W after we normalize all the rows so that they sum to one. Also let J be an $n \times n$ "jump" matrix, where for all i, j,

$$J[i,j] = \epsilon\mathcal{D}(j). \tag{2.2}$$

Now let $M_{PR} = J + (1-\epsilon)W_r$. This is the matrix of the Markov Chain that corresponds to the random walk performed by the PAGERANK algorithm. The addition of the jump matrix guarantees that the Markov Chain is irreducible and aperiodic, then there is an equilibrium steady state distribution for the states of the Markov Chain. The stationary distribution $\boldsymbol{a}$ is the left eigenvector of the matrix M_{PR}, that is,

$$\boldsymbol{a} = \boldsymbol{a}M_{PR} \; .$$

Normally, the graph G contains many nodes with no out-going links. Brin and Page [13] propose to remove these nodes from the graph, and run the PAGERANK algorithm on the resulting graph. Then, they use Equation 2.1 to assign a PageRank value to the removed nodes. If we do not wish to resort to this heuristic method, the following two implementations of PageRank have been considered in the literature. The algorithm may force a random jump whenever reaching a dead-end. In this case the jump matrix is defined as follows.

$$J[i,j] = \begin{cases} \epsilon\mathcal{D}(j) & \text{if } F(j) \neq \emptyset \\ \mathcal{D}(j) & \text{if } F(j) = \emptyset \end{cases}$$

Again the authority vector is the left eigenvector of the matrix $M_{PR} = J + (1-\epsilon)W_r$.

Alternatively, self loops are introduced to dead-end nodes. That is, $W_r[i,i] = 1$ if $F(i) = \emptyset$. Then, $M_{PR} = J + (1-\epsilon)W_r$, where J is defined as in Equation 2.2.

2.2.3 The HITS Algorithm

Independent of Brin and Page, Kleinberg [58] proposed a more refined notion for the importance of Web pages. He proposed a two-level weight propagation scheme where endorsement is conferred on authorities through hubs, rather than directly between authorities. In his framework, every page can be thought of as having two identities. The *hub* identity captures the quality of the page as a pointer to useful resources, and the *authority* identity captures the quality of the page as a resource itself. A good authority is a source of useful information, while a good hub is a page that contains a useful collection of links. If we make two copies of each page, we can visualize graph G as a bipartite graph, where hubs point to authorities. There is a mutual reinforcing relationship between the two. A good hub is a page that points to good authorities, while a good authority is a page pointed to by good hubs. In order to quantify the quality of a page as a hub and an authority, Kleinberg associated every page with a hub and an authority weight. Following the mutual reinforcing relationship between hubs and authorities, Kleinberg defined the hub weight to be the sum of the authority weights of the nodes that are pointed to by the hub, and the authority weight to be the sum of the hub weights that point to this authority. Let $\boldsymbol{h}$ denote the n-dimensional vector of the hub weights, where h_i, the i-th coordinate of vector $\boldsymbol{h}$, is the hub weight of node i. We have that

$$a_i = \sum_{j \in B(i)} h_j \qquad \text{and} \qquad h_j = \sum_{i \in F(j)} w_i \ . \tag{2.3}$$

In matrix-vector terms

$$\boldsymbol{a} = W^T \boldsymbol{h} \qquad \text{and} \qquad \boldsymbol{h} = W \boldsymbol{a} \ .$$

Building upon the mutual reinforcing relationship between hubs and authorities, Kleinberg proposed the following iterative algorithm for computing the hub and authority weights. Initially all authority and hub weights are set to 1. At each iteration the operations $\mathcal{O}$ ("out") and $\mathcal{I}$ ("in") are performed. The $\mathcal{O}$ operation updates the authority weights, and the $\mathcal{I}$ operation updates the hub weights, both using the Equation 2.3. A normalization step is then applied, so that the vectors $\boldsymbol{a}$ and $\boldsymbol{h}$ become unit vectors in some norm. The algorithm iterates until the vectors converge. Let $\boldsymbol{a}^t$ denote the authority vector after the t-th iteration. Given a constant ϵ, we say the the vector $\boldsymbol{a}^t$ has converged, if $\|\boldsymbol{a}^t - \boldsymbol{a}^{t-1}\| \leq \epsilon$, where $\|\cdot\|$ is the normalization norm. This idea was later implemented as the HITS (Hyperlink Induced Topic Distillation) algorithm [40]. The algorithm is summarized in Figure 2.1.

Kleinberg proves that the algorithm computes the principal left and right *singular* vectors of the adjacency matrix W. That is, the vectors $\boldsymbol{a}$ and $\boldsymbol{h}$ converge to the principal right eigenvectors of the matrices $M_H = W^T W$ and $M_H^T = W W^T$, respectively. The convergence of HITS to the singular vectors of matrix W is subject to the condition that the initial authority and hub vectors are not orthogonal to the principal eigenvectors of matrices M_H and M_H^T respectively. Since these eigenvectors have non-negative values, it suffices to initialize all weights to positive values, greater than zero. We discuss the properties of the

```
HITS

Initialize all weights to 1
Repeat until the weights converge:
    For every hub $i \in H$
        $h_i = \sum_{j \in F(i)} a_j$
    For every authority $i \in A$
        $a_i = \sum_{j \in B(i)} h_j$
    Normalize
```

Figure 2.1: The HITSAlgorithm

vectors in detail in Section 2.3 where we talk about *Singular Value Decomposition (SVD).* The convergence of the HITS algorithm does not depend on the normalization. Indeed, for different normalization norms, the authority weights are the same up to a constant scaling factor. Let $\|\cdot\|_p$ and $\|\cdot\|_q$ denote two different norms, and let $a_p(i)$ and $a_q(i)$ denote the weight of node i, when norm p and q respectively are used for the normalization step. Then we have that

$$a_p(i) = \frac{\|\boldsymbol{a}\|_q}{\|\boldsymbol{a}\|_p} a_q(i)$$

Note that the relative order of the nodes in the ranking does not depend on the normalization.

2.2.4 The SALSA Algorithm

An alternative algorithm, SALSA, was proposed by Lempel and Moran [64], that combines ideas from both HITS and PAGERANK. As in the case of HITS, visualize the graph G as a bipartite graph, where hubs point to authorities. The SALSA algorithm performs a random walk on the bipartite hubs and authorities graph, alternating between the hub and authority sides. The random walk starts from some authority node selected uniformly at random. The random walk then proceeds by alternating between backward and forward steps. When at a node on the authority side of the bipartite graph, the algorithm selects one of the incoming links uniformly at random and moves to a hub node on the hub side. When at node on the hub side the algorithm selects one of the outgoing links uniformly at random and moves to an authority. The authority weights are defined to be the stationary distribution of this random walk. Formally, the Markov Chain of the random walk has transition probabilities

$$P_a(i,j) = \sum_{k\,:\,k \in B(i) \cap B(j)} \frac{1}{|B(i)|} \frac{1}{|F(k)|}.$$

Recall that $G_a = (A, E_a)$ denotes the authority graph, where there is an (undirected) edge between two authorities if they share a hub. This Markov Chain corresponds to a random walk on the authority graph G_a, where we move from authority i to authority j with probability $P_a(i,j)$. Let W_r denote the matrix derived from matrix W by normalizing the entries such that, for each row, the sum of the entries is 1, and let W_c denote the matrix derived from matrix W by normalizing the entries such that, for each column, the sum of the entries is 1. Then the stationary distribution of the SALSA algorithm is the principal left eigenvector of the matrix $M_S = W_c^T W_r$. The algorithm is shown in Figure 2.2.

If the underlying authority graph G_a consists of more than one component, then the SALSA algorithm selects a starting point uniformly at random, and performs a random walk within the connected component that contains that node. Let j be a component that contains node i, let A_j denote the set of authorities in the component j, and E_j the set of links in component j. Then the weight of authority i in component j is

$$a_i = \frac{|A_j|}{|A|} \frac{|B(i)|}{|E_j|} \ .$$

If the graph G_a consists of a single component (we refer to such graphs as *authority connected* graphs), that is, the underlying Markov Chain is *irreducible*, then the algorithm reduces to the INDEGREE algorithm. Furthermore, even when the graph G_a is not connected, if the starting point of the random walk is selected with probability proportional to the "popularity" (in-degree) of the node in the graph G, then the algorithm again reduces to the INDEGREE algorithm. This algorithm was referred to as pSALSA (popularity SALSA) by Borodin et al. [10].

The SALSA algorithm can be thought of as a variation of the HITS algorithm (Figure 2.2). In the $\mathcal{I}$ operation of the HITS algorithm the hubs *broadcast* their weights to the authorities, and the authorities sum up the weight of the hubs that point to them. The SALSA algorithm modifies the $\mathcal{I}$ operation as follows. Instead of broadcasting, each hub *divides* its weight

```
Salsa

Initialize all weights uniformly
Repeat until the weights converge:
    For every hub $i \in H$
        $h_i = \sum_{j \in F(i)} \frac{1}{|B(i)|} a_j$
    For every authority $i \in A$
        $a_i = \sum_{j : j \in B(i)} \frac{1}{|F(j)|} h_j$
    Normalize
```

Figure 2.2: The Salsa Algorithm

equally among the authorities to which it points. Therefore,

$$a_i = \sum_{j : j \in B(i)} \frac{1}{|F(j)|} h_j \ .$$

Similarly, the Salsa algorithm modifies the $\mathcal{O}$ operation so that each authority divides its weight equally among the hubs that point to it. Therefore,

$$h_j = \sum_{i : j \in F(i)} \frac{1}{|B(i)|} a_j \ .$$

However, the Salsa algorithm does not really have the same "mutually reinforcing structure" that Kleinberg's algorithm does. Indeed, $a_i = \frac{|A_j|}{|A|} \frac{|B(i)|}{|E_j|}$, the relative authority of site i *within a connected component* is determined from local links, not from the structure of the component.

Lempel and Moran [64] define a similar Markov Chain for the hubs that has transition probabilities

$$P_h(i, j) = \sum_{k \,:\, k \in F(i) \cap F(j)} \frac{1}{|F(i)|} \frac{1}{|B(k)|} \ .$$

The stationary distribution $\boldsymbol{h}$ is the left eigenvector of the matrix $W_r^T W_c$.

2.3 Singular Value Decomposition

Singular Value Decomposition is a powerful technique for data analysis. In this section we give a brief overview of the method. For the following we assume a familiarity of the reader with basic concepts and terminology of linear algebra, and we refer the reader to the excellent textbook of Strang [87] for an introduction to the field.

The Singular Value Decomposition of a matrix M is defined as follows [87].

Theorem 2.1 *Given an m by n matrix M, we can express it as*

$$M = U\Sigma V^T \tag{2.4}$$

where U is a column orthonormal m by r matrix, r is the rank of the matrix M, $\Sigma = diag(\sigma_1, \sigma_2, \ldots, \sigma_r)$ is an r by r diagonal matrix, where $\sigma_1, \sigma_2, \ldots, \sigma_r$ are positive and non-zero, and V is a column orthogonal n by r matrix.

Recall that the rank of a matrix is the number of rows (or columns) that are linearly independent. Also, a matrix U is column orthonormal if and only if its column vectors are all orthogonal (i.e., their dot product is equal to zero) and all column vectors have unit length in the Euclidean norm. Equivalently, $U^TU = I$, where I is the identity matrix. The values $\sigma_1, \ldots, \sigma_r$ are called *singular values* of the matrix, while the column vectors of matrices U and V are called *left singular* and *right singular* vectors respectively. If we insist that the singular values are sorted in a non-increasing order, then, if the singular values are distinct, this decomposition is unique. Given this decomposition of M, we will refer to the k-th pair of singular vectors, as the k-th *principal* singular vectors. We will often refer to the first principal singular vectors, as the principal singular vectors.

It is not hard to see (by performing simple matrix multiplications) that the left singular vectors in matrix U are the eigenvectors of the matrix MM^T, while the right singular vectors in matrix V are the eigenvectors of the matrix M^TM. Furthermore, the r non-zero eigenvalues of these two matrices, are equal to the squares of the singular values of the matrix M.

The Singular Value Decomposition has some interesting properties. From the definition of SVD, we can write M as

$$M = \sum_{i=1}^{r} \sigma_i u_i v_i^T.$$

Let k be a number such that $1 \leq k \leq r$. We can approximate matrix M by

$$M_k = \sum_{i=1}^{k} \sigma_i u_i v_i^T$$

that is, as the sum of k rank one matrices defined by the first k principal singular vectors. Matrix M_k has rank k, and it can be shown that it is the best possible rank k approximation of matrix M with respect to the Frobenius and the L_2 matrix norms. Let $\|M\|_F$ denote the *Frobenius* norm of matrix M, where

$$\|M\|_F = \sum_{1\leq i\leq m}\sum_{1\leq j\leq n} M[i,j]^2 \ .$$

Also, let $\|M\|_2$ denote the L_2 norm of matrix M, where

$$\|M\|_2 = \max_{|x|=1} |Mx| \ ,$$

where $|x|$ denotes the Euclidean norm of vector x. The proof of the following theorem can be found in the textbook of Golub and Van Loan [41].

Theorem 2.2 *Let the SVD of A be given by Theorem 2.1. If $k < r$ and*

$$A_k = \sum_{i=1}^{k} \sigma_i u_i v_i^T ,$$

then

$$\min_{rank(B)=k} ||A - B||_F = ||A - A_k||_F = \sum_{i=k+1}^{r} \sigma_i^2 \tag{2.5}$$

and

$$\min_{rank(B)=k} ||A - B||_2 = ||A - A_k||_2 = \sigma_{k+1} \tag{2.6}$$

The columns of the U and V matrices are orthogonal unit vectors. Therefore any k of them define a basis for a k-dimensional space. Intuitively, the singular vectors define the basis of a *feature* space. The matrix M can be thought of as a matrix that associates two different types of entities: objects (rows) with attributes (columns). The objects are expressed as vectors in the attribute space, while the attributes are defined as vectors in the object space. The singular vectors of matrix V define features in the attribute space, while the singular vectors of matrix U define features in the object space. From Equation 2.4 we have that

$$MV = U\Sigma \ .$$

The product MV in the left hand side defines a projection of the objects (row vectors) from the attribute space on the feature space defined by the column vectors of matrix V. The positions of the projections in the feature space are given by the rows of the matrix $U\Sigma$ in the right hand side. Therefore, the matrix V defines the directions on which the object vectors are projected to in the feature space, and $U\Sigma$ defines the mapping of the object vectors to the feature space. Similarly, the column vectors of U define the directions on which the attribute vectors are projected to the feature space, and the rows of the matrix $V\Sigma$ define the positions of the projections in the feature space.

Achlioptas and McSherry [2] present an intuitive explanation as to why the k principal singular vectors correspond to the k strongest *linear trends* in the dataset. Consider a matrix M and let v_1 denote the principal right singular vector of M. It can be proved that the vector v_1 is the "certificate" for the L_2 norm of the matrix M. That is, the vector v_1 is the unit vector that causes $|Mx|$ to be maximized. For matrix M, let $M[i]$ denote the vector of the i-th row of matrix M. Then

$$|Mv_1| = \sqrt{\left(\sum_{i=1}^{m} M[i] \cdot v_1\right)^2} .$$

We can think of the dot product as a measure of the similarity of two vectors, where similarity captures how closely aligned these two vectors are. Then, vector v_1 captures the strongest linear trend in the attribute space, in the sense that it is the vector that is most closely aligned with the row vectors of the matrix M. The strength of the strongest linear trend is captured in the value of $|Mv_1| = \sigma_1$. Let $R_1 = M - M_1$ denote the matrix M after we remove the rank one approximation $M_1 = Mv_1v_1^T$. The second principal right singular vector corresponds to the certificate of the strongest linear trend of the matrix R_1. Continuing this process we can obtain all the right singular vectors of matrix M.

Returning to link analysis, the Hits algorithm computes the principal singular vectors of the authority to hub matrix W^T. The hub vector $\boldsymbol{h}$, and the authority vector $\boldsymbol{a}$ are the right and left principal singular vectors respectively of the matrix W^T. In this matrix, the rows correspond to the authorities, which can be thought of as vectors in a hub space. The hub vector $\boldsymbol{h}$ captures the strongest linear trend within this space. The authority vectors are projected on this vector. The projection lengths are the authority weights, and they capture how closely aligned is each authority with the strongest linear trend. The authority vector $\boldsymbol{a}$ is also the strongest linear trend in the hub space.

Drineas et al. [29] reveal another property of the HITS algorithm. They propose to think of the vectors $\boldsymbol{a}$ and $\boldsymbol{h}$ as *clusters* of authorities and hubs. The value of a_i is the *intensity* with which the node i belongs to the cluster $\boldsymbol{a}$. Similarly, the value h_i denotes the intensity with which node i belongs to cluster $\boldsymbol{h}$. The authors propose to use $|\boldsymbol{h}^T W|^2$ and $|W\boldsymbol{a}|^2$ as a measure of the strength of the clusters. The i-th coordinate of vector $\boldsymbol{h}^T W$ is $(\boldsymbol{h}W)_i = \sum_{j=1}^{n} h_j W[j,i]$ and it captures the *frequency* with which node i appears in the neighborhood of the cluster $\boldsymbol{h}$. A large value for $(\boldsymbol{h}W)_i$ implies that node i is pointed to by many hubs with high intensity value. Thus, node i is strongly affiliated with the cluster $\boldsymbol{h}$. Furthermore, node i reinforces the relationship between the hubs. Since $|\boldsymbol{h}^T W|^2 = \sum_{i=1}^{n}(\boldsymbol{h}W)_i^2$ we want to maximize this reinforcing relation. Note that $\boldsymbol{h}^T W = \sigma_1 \boldsymbol{a}$. The vectors $\boldsymbol{a}$ and $\boldsymbol{h}$ are constructed such that $|\boldsymbol{h}^T W|^2$ and $|W\boldsymbol{a}|^2$ are maximized, since they are the certificates for the L_2 norm of the W and W^T matrices respectively. Thus, the vectors $\boldsymbol{h}$ and $\boldsymbol{a}$ have the property that they maximize the reinforcing relation between hubs and authorities. A symmetric argument can be made if we consider the cluster $\boldsymbol{a}$.

The reinforcing relation is captured in the fact that the HITS algorithm assigns the highest authority and hub weights to a set of nodes that are tightly interconnected; the most *Tightly Knit Community* in the graph, as it is often referred to in the literature [58, 64]. That is, HITS promotes sets of nodes S and T such that there are many edges of the form (u, v), where $u \in S$ and $v \in T$. Drineas et al. [29] give further support to this claim. Consider the residual matrix $R_1 = W - W_1$ that is obtained by removing matrix $W_1 = \sigma_1 \boldsymbol{a}^T \boldsymbol{h}$ from matrix W. They prove that for any subsets S, T of nodes

$$\left| \sum_{i \in S, j \in T} R_1[i,j] \right| \leq \sigma_2 \sqrt{|S||T|}\ .$$

Since $\sigma_1 \geq \sigma_2$, it follows that matrix W_1 contains the most dense connected component in the matrix W.

2.3.1 Some Applications of Singular Value Decomposition in Computer Science

One of the most celebrated applications of Singular Value Decomposition has been in the area of information retrieval [23, 6]. Deerwerster et al. [23] introduced the Latent Semantic

Analysis where they proposed to apply Singular Value Decomposition on the document to term matrix. The effect of SVD is not only to reduce the dimensionality of the term space, but also to group together documents on the same topic, thus dealing with problems such as *synonymy* and *polysemy.* Synonymy refers to the case that two different words have the same meaning (e.g., car and automobile), while polysemy refers to the case that a word has multiple meanings (e.g., windows, referring to the actual house windows, and the Microsoft operating system). The problem of synonymy in Information Retrieval is to be able to retrieve documents that contain the synonyms of the query word. The problem of polysemy is to be able to separate between documents that refer to different meanings of the query word. Latent Semantic Analysis appears to be able to deal with these two problems. Preliminary work on the theoretical support of this phenomenon was later provided by Papadimitriou et al. [75], and it was later extended by the work of Azar et al. [4], and Achlioptas et al. [1].

Singular Value Decomposition has been applied in many different settings. Usually, it is applied as a dimensionality reduction tool, for obtaining a concise and compact representation of a large dataset. For example, we reference some applications in lossy compression [60], query processing [77], image recognition [91], and clustering [71, 29, 54].

2.4 Other Related Work

Link analysis has been previously considered in different areas. In the area of social networks, given a network of endorsements between the members of a community, we are interested in determining the *standing* of an individual within the community. This is a notion very close to that of the authoritativeness. The algorithms proposed by Katz [55], and Hubbell [51] for computing the standing of an individual can be seen as the forefathers of the link analysis ranking algorithms for the Web.

The area closest to the link analysis of Web documents is *bibliometrics* [31]. The object of study of bibliometrics is a collection of academic documents. Link analysis is applied to the citation structure of the documents for calculating the *impact* of a scientific journal. The algorithms in this area include the celebrated *impact factor* algorithm by Garfield [42], as well as extensions and variations of this algorithm [76, 38, 27, 28].

The analysis of link structures in hypertext documents dates back to the work of

Frisse [37], who proposed heuristics for enhancing the *relevance* of a document using hyperlink information. Botafogo, Rivlin and Shneiderman [12] define *index* and *reference* nodes, which can be viewed as the predecessors of hubs and authorities. They also proposed to use the *centrality* of a node as a ranking criterion. Carrière and Kazman [15] proposed simple heuristics that use the in and out degree of a node for ranking. Marchiori [69] proposed the HyperSearch algorithm for determining the relevance of a hypertext document, using the relevance of the pages that can be reached from this document.

However, the origin of link analysis ranking can be pinpointed to the ground-breaking work of Kleinberg [57, 58], and Brin and Page [13]. These two algorithms spawned the area of link analysis ranking and were followed by a substantial amount of research work. Bharat and Henzinger [8] considered improvements on the Hits algorithm by using textual information to weight the importance of nodes. They also down-weight the importance of links that arrive from, or end at the same host by averaging. The ARC algorithm [16] enhances the Hits algorithm by weighting the edges of the graph, taking into account the anchor text of the hyperlink, and the surrounding text. Further improvements are suggested by Li, Shang and Zhang [67] that make use of relevance scores. Bharat and Mihaila [9] suggest an elaborate way of finding experts on topics, and constructing a bipartite graph of experts and targets. They assign an expert cost to every expert. Then they propose to set the authority weight of the targets to be the sum of the expert weights of the experts that point to them. Rafiei and Mendelzon [78, 70] consider the application of link analysis for determining the *reputation* of a page. They propose the application of the PageRank algorithm, and they also consider a combination of the PageRank and Hits algorithms. This is the same as the Salsa algorithm, except that, similar to PageRank, at every step the algorithm makes a jump to a random page with probability ϵ. Almost the same algorithm is also proposed by Ng, Zheng and Jordan [72, 73], termed *Randomized HITS.*

Page, Brin, Motwani and Winograd [74] propose personalized versions of the PageRank algorithm, by selecting the jump distribution $\mathcal{D}$ so that it favors pages selected by the user. Recently, Haveliwala [44] proposed a topic sensitive version of the PageRank algorithm. For a given set of topics he computes a topic specific PageRank value by making the jump distribution biased towards that topic. Then depending on how related the query is to that topic, he sets the final weight of the document to be a weighted combination of the PageRank values for each topic. A similar idea is explored by Jeh and Widom [53] where a sophisticated algorithm for combining PageRank vectors is proposed that requires less computational time and storage space, and it allows for a larger set of *base topics* to be considered. Richardson and Domingos [79] consider a topic sensitive version of PageRank where for every query word a different jump distribution $\mathcal{D}$ is used, and a different PageRank value is computed, and they argue about the scalability of their approach in terms of storage and computational costs.

Extensions of the Hits algorithm that use multiple eigenvectors were proposed by Ng, Zheng and Jordan [73], and Achlioptas et al. [4]. Ng, Zheng and Jordan propose to use multiple singular vectors of the adjacency matrix for defining a subspace in the hub space on which to project the authority vectors. Achlioptas et al., propose a model for the generation of links, text and user queries. Their model assumes the existence of latent communities of Web pages. Based on this model they propose an algorithm that uses Singular Value Decomposition to discover authoritative pages for a given query.

A different line research exploits the application of probabilistic and statistical techniques for computing rankings. The PHits algorithm by Cohn and Chang [19] assumes a probabilistic model in which a link is caused by latent "factors" or "topics". They use the Expectation Maximization (EM) Algorithm of Dempster et al. [24] to compute the authority weights of the pages. Their work is based on the *Probabilistic Latent Semantic Analysis* framework introduced by Hofmann [48], who proposed a probabilistic alternative to Singular Value Decomposition. Hofmann [49] proposes an algorithm similar to PHits which also takes into account the text of the documents. In the work with Borodin, Roberts and Rosenthal [10] we considered a probabilistic algorithm that assumes that each page has latent authority, hub, and "link tendency" parameters, and that the generation of a link between two pages is governed by these parameters. Given a well specified model, they condition on the observed data (the actual links in the graph) to compute the posterior distribution of the parameters. They apply a Metropolis Monte Carlo algorithm for computing the conditional means of the authority parameters which are output as the authority weights. Roberts and Rosenthal [80] propose an algorithm that groups similar pages together to create "super-nodes". Then, they apply probabilistic techniques on this graph to compute the ranking. They observed that their algorithms have the property of escaping the Tightly Knit Community (TKC) effect.

Tomlin [90] generalizes previous work on targeted advertising [89], and proposes a generalization of the PageRank measure. He proposes to generalize the idea of the random surfer, and to assign flows on the edges of the Web graph. Then, using the maximum entropy principle he estimates the values for flows on the edges of the graph. The author proposes, as a ranking value for every page, the total flow that arrives to a page (*TrafficRank*). As a by-product of the algorithm a Lagrange multiplier is computed for each page, which captures the "temperature" of the page. The author proposes to use this "temperature" (*HOTness*) as the authority value.

The problem of retrieving and ranking documents from a given corpus has been studied extensively in the area of Information Retrieval [5]. An influential conference in this area is the Text REtrieval Conference (TREC). The aim of TREC is to evaluate document search algorithms, using benchmark text collections. The document collections accompanied with a set of queries. For each query, relevance feedback is provided for each document in the collection. The algorithms are evaluated by computing the *precision* over the top k results. Recently, the TREC conference has created a separate Web Track for the study of document search algorithms over collections of Web documents. Early studies [82, 83, 46, 47] with link analysis ranking algorithms (including PageRank and Hits) indicated that link analysis does not improve the quality of search, compared to traditional text searching algorithms. However, the study of Singhal and Kaszkiel [85] demonstrated that commercial search engines that use some form of link analysis outperform the state of the art TREC algorithms. Their study focused on *Home Page queries*, where the objective of the user is to discover the home page of an organization, or a person. They speculated that the observations in the earlier studies were due to the limitations of the corpus on which the algorithms were tested, and the nature of the queries. Recent experiments with link analysis algorithms on TREC Web data [20] demonstrate that, although link analysis does not improve much upon traditional information retrieval algorithms for *topical* queries (queries for finding relevant pages to a topic), it exhibits considerable improvement when applied to Home Page queries. The experiments were limited to specific LAR algorithms and techniques. Although these studies are indicative of the limitations of Link Analysis Ranking, the TREC experiments do not offer a conclusive argument for the value of Link Analysis in Ranking.

Apart from ranking, link analysis has been applied in diverse contexts. Lempel and Soffer [66] propose the application of link analysis for retrieving images. Dean and Henzinger [22] propose the application of link analysis for finding related pages to a query page.

Guo, Shao, Botev and Shanmugasundaram [43] consider the application of the PageRank algorithm on XML documents, where rankings are computed at the granularity of XML elements rather than documents. Chakrabarti, Dom and Indyk [17] propose the use of link information for classification of Web pages. Lee [62] uses links for aggregating the results of multiple search engines.

There is a considerable amount of work on clustering Web pages using link information. Kumar, Raghavan, Rajagopalan and Tomkins [61] defined a community as a set of pages that contain a complete bipartite graph, called the *core* of the community. They proposed efficient algorithms for identifying such bipartite cliques. Flake and co-authors [35, 36] use maximum flow algorithms to define and discover communities in Web graphs. Fagin et al. [33] consider the application of link analysis ranking on intranets. Recently, Bhalotia et al. [7] proposed the application of link analysis to the problem of ranking the query results for keyword searches in relational data. There is a substantial amount of work in this area, but a detailed review is beyond the scope of this thesis.

Chapter 3

Link Analysis Ranking Algorithms

3.1 Implicit Properties of the HITS Algorithm

The idea underlying the HITS algorithm can be captured in the following recursive definition of quality: "A good authority is one that is pointed to by many good hubs, and a good hub is one that points to many good authorities". Therefore, the quality of some page p as an authority (captured by the authority weight of page p) depends on the quality of the pages that point to p as hubs (captured in the hub weight of the pages), and vice versa. Kleinberg proposes to associate the hub and authority weights through the addition operation. The authority weight of a page p is defined to be the sum of the hub weights of the pages that point to p, and the hub weight of the page p is defined to be the sum of the authority weights of the pages that are pointed to by p. This definition has the following two implicit properties. It is *symmetric*, in the sense that both hub and authority weights are defined in the same way. If we reverse the orientation of the edges in the graph G, then authority and hub weights are swapped. The HITS algorithm is also *egalitarian*, in the sense that when computing the authority weight of some page p, the hub weights of the pages that point to page p are all treated equally (similarly when computing the hubs weights).

However, these two properties may some times lead to non-intuitive results. Consider for example the graph in Figure 3.1. In this graph there are two components. The black component consists of a single authority pointed to by a large number of hubs. The white component consists of a single hub that points to a large number of authorities. If the number of white authorities is larger than the number of black hubs then the HITS algorithm will allocate all authority weight to the white authorities, while giving zero weight to the

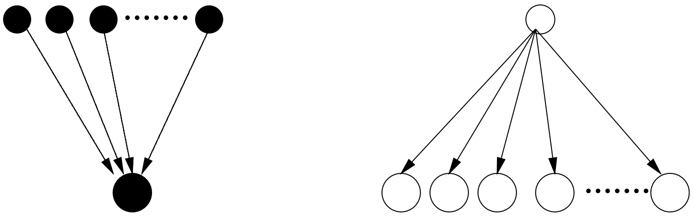


Figure 3.1: A bad example for HITS algorithm

black authority. The reason for this is that the white hub is deemed to be the best hub, thus causing the white authorities to receive more weight. However, intuition suggests that the black authority is better than the white authorities and should be ranked higher.

In this example, the two implicit properties of the HITS algorithm combine to produce this non-intuitive result. Equality means that all authority weights of the nodes that are pointed to by a hub contribute equally to the hub weight of the node. As a result quantity becomes quality. The hub weight of the white hub increases inordinately because it points to many weak authorities. This leads us to question the definition of the hub weight, and consequently other implicit property of HITS. Symmetry assumes that hubs and authorities are qualitatively the same. However, there is a difference between the two. For example, intuition suggests that a node with high in-degree is likely to be a good authority. On the other hand, a node with high out-degree is not necessarily a good hub. If this was the case, then it would be easy to increase the hub quality of a page, simply by adding links to random pages. It seems that we should treat hubs and authorities in different manners.

In this chapter we challenge both implicit properties of HITS. We present different ways for breaking the symmetry and equality principles and we study the ranking algorithms that emerge.

3.2 The Hub-Averaging (HUBAVG) Algorithm

In the example of Figure 3.1, the symmetric and egalitarian nature of the HITS algorithm has the effect that the quality of the white hub is determined by the quantity of authorities it points to. Thus, the white hub is rewarded simply because it points to a large number of authorities, even though they are of low quality. We propose a modification of the HITS algorithm to help remedy the above-mentioned problem. The Hub-Averaging algorithm (HUBAVG) (first presented in the collaborative work with A. Borodin, G. Roberts, and J.

```
HubAvg

Initialize authority weights to 1
Repeat until the weights converge:
    For every hub $i \in H$
        $h_i = \frac{1}{|F(i)|} \sum_{j \in F(i)} a_j$
    For every authority $i \in A$
        $a_i = \sum_{j \in B(i)} h_j$
    Normalize
```

Figure 3.2: The HubAvg Algorithm

Rosenthal [10]) updates the authority weights like the Hits algorithm, but it sets the hub weight of some node i to the average authority weight of the authorities pointed to by hub i. Thus for some node i we have

$$a_i = \sum_{j \in B(i)} h_j \qquad \text{and} \qquad h_i = \frac{1}{|F(i)|} \sum_{j \in F(i)} a_j \ . \tag{3.1}$$

The intuition of the HubAvg algorithm is that a good hub should point *only* (or at least mainly) to good authorities, rather than to both good and bad authorities. Note that in the example of Figure 3.1, HubAvg assigns the same weight to both black and white hubs, and it identifies the black authority as the better authority. The HubAvg algorithm is summarized in Figure 3.2.

The HubAvg algorithm can be viewed as a "hybrid" of the Hits and Salsa algorithms. The operation of averaging the weights of the authorities pointed to by a hub is equivalent to dividing the weight of a hub among the authorities it points to. Thus, we can replace Equation 3.1 by the following.

$$a_i = \sum_{j \in B(i)} \frac{1}{|F(j)|} h_j \qquad \text{and} \qquad h_i = \sum_{j \in F(i)} a_i \ .$$

Therefore, the HubAvg algorithm performs the $\mathcal{O}$ operation like the Hits algorithm (broadcasting the authority weights to the hubs), and the $\mathcal{I}$ operation like the Salsa algorithm (dividing the hub weights to the authorities). This lack of symmetry between the update of hubs and authorities is motivated by the qualitative difference between hubs and authorities previously discussed. The authority weights for the HubAvg algorithm converge to

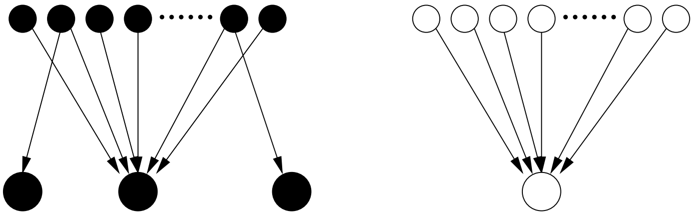


Figure 3.3: A bad example for the HubAvg algorithm

the principal right eigenvector of the matrix $M_{HA} = W^T W_r$.

It is interesting to observe what happens if we make the algorithm symmetric. There are two ways to re-establish symmetry. We can let the authorities divide their weight among the hubs that point to them. In this case authority and hub weights are defined as follows.

$$a_i = \sum_{j \in B(i)} \frac{1}{|F(j)|} h_j \qquad \text{and} \qquad h_i = \sum_{j \in F(i)} \frac{1}{|B(j)|} a_i \ .$$

The authority weights will then converge to the principal *left* eigenvector of the matrix $W_c^T W_r$ and the algorithm becomes the Salsa algorithm. Alternatively, we can make the authority weight of a node be the average of the hub weights that point to that node. In this case the authority and hub weights are updated as follows.

$$a_i = \frac{1}{|B(i)|} \sum_{j \in B(i)} h_j \qquad \text{and} \qquad h_i = \frac{1}{|F(i)|} \sum_{j \in F(i)} a_j$$

The authority weights will then converge to the principal *right* eigenvector of the matrix $W_c^T W_r$. Since this is a stochastic matrix, the principal right eigenvector is the uniform vector. Thus, the algorithm degenerates to the algorithm that assigns the same weight to each node in the graph.

3.3 The Authority Threshold (AT(k)) Family of Algorithms

The HubAvg algorithm has its own shortcomings. Consider for example the graph in Figure 3.3. In this graph there are again two components, one black and one white. They are completely identical, except for the fact that some of the hubs of the black component point to a few extra authorities. If we run the HubAvg algorithm on this graph, then the white authority will receive higher authority weight than the black authority. This is due to the fact that the black hubs are *penalized* for pointing to these “weaker” authorities. The HubAvg algorithm rewards hubs that point *only* (or mainly) to good authorities. Hubs that have links to a few poor authorities are penalized. However, this is not always fair. In the example of Figure 3.2, the black authority seems to be at least as authoritative as the white authority. Although we would not like the black hubs to be rewarded for pointing to these weak authorities, we do not necessarily want them to be penalized either. Such situations may arise in practice, where a node is at the same time a strong hub on one topic, and a weak hub on another topic. Such hubs are penalized by the HubAvg algorithm.

What we want is to reduce the effect of the weak authorities on the computation of the hub weight, while at the same time we retain the positive effect of the strong authorities. A simple solution is to apply a threshold operator, that retains only the highest authority weights. We propose the *Authority-Threshold*, AT(k), algorithm (first presented in the collaborative work with A. Borodin, G. Roberts, and J. Rosenthal [10]), which sets the hub weight of node i to be the sum of the k largest authority weights[1] of the authorities pointed to by node i. This corresponds to saying that a node is a good hub if it points to *at least* k good authorities. The value of k is passed as a parameter to the algorithm.

Formally, let $F_k(i)$ denote the subset of $F(i)$ that contains k nodes with the highest authority weights. That is, for any node $p \in F(i)$, such that $p \notin F_k(i)$, $a_p \leq a_q$, for all $q \in F_k(i)$. If $|F(i)| \leq k$, then $F_k(i) = F(i)$. The AT(k) algorithm computes the authority and hub weights as follows.

$$a_i = \sum_{j \in B(i)} h_j \qquad \text{and} \qquad h_i = \sum_{j \in F_k(i)} a_j$$

The outline of the AT(k) algorithm is shown in Figure 3.4.

It is interesting to examine what happens at the extreme values of k. For $k = 1$, the threshold operator becomes the max operator. We will discuss this case in detail in Section 4.2. If d_{out} is the maximum out-degree of any node in the graph G, then for $k \geq d_{out}$, the AT(d_{out}) algorithm is the Hits algorithm.

[1] Other types of threshold are possible. For example the threshold may depend on the largest difference between two weights.

```
AT($k$)

Initialize authority weights to 1
Repeat until the weights converge:
    For every hub $i \in H$
        $h_i = \sum_{j \in F_k(i)} a_j$
    For every authority $i \in A$
        $a_i = \sum_{j \in B(i)} h_j$
    Normalize
```

Figure 3.4: The AT(k) Algorithm

3.4 The Norm(p) Family of Algorithms

The Authority Threshold algorithm operates on the principle of *preferential treatment* of the authority weights. That is, higher authority weights should be more important in the computation of the hub weight. This principle is enforced by applying a threshold operator. A smoother approach is to *scale* the weights, so that lower authority weights contribute less to the hub weight. An obvious question is how to select the scaling factors. A natural solution is to use the weights themselves to determine the scaling factors.

This idea is implemented in the Norm(p) family of algorithms. In this case we set the hub weight of node i to be the p-norm of the vector of the authority weights of the nodes pointed to by node i. Recall that the p-norm of vector $\mathbf{x} = (x_1, \ldots, x_n)$ is defined as follows:

$$\|\mathbf{x}\|_p = \left(\sum_{i=1}^{n} x_i^p \right)^{1/p} .$$

The authority and hub weights are computed as follows:

$$a_i = \sum_{j \in B(i)} h_j \qquad \text{and} \qquad h_i = \left(\sum_{j \in F(i)} a_j^p \right)^{1/p} .$$

The value of p is passed as a parameter to the algorithm. We assume that $p \in [1, \infty]$ As p increases the value of the p-norm is dominated by the highest weights. For example, for $p = 2$, we essentially scale every weight with itself. The outline of the Norm(p) algorithm is shown in Figure 3.5. An almost identical algorithm was proposed by Gibson, Kleinberg and Raghavan [39] for clustering categorical data.

```
NORM(p)

Initialize authority weights to 1
Repeat until the weights converge:
    For every hub $i \in H$
        $h_i = \left(\sum_{j \in F(i)} a_j^p\right)^{1/p}$
    For every authority $i \in A$
        $a_i = \sum_{j \in B(i)} h_j$
    Normalize
```

Figure 3.5: The NORM(p) Algorithm

Again, it is interesting to examine the behavior of the algorithm in the extreme cases of the value p. For $p = 1$ the NORM(1) algorithm is the HITS algorithm. For $p = \infty$ the p-norm reduces to the max operator.

3.4.1 Making it Symmetric: The DOUBLENORM(p) algorithm

The NORM(p) algorithm can be made symmetric by setting the authority weight of a node to be the p-norm of the vector of the hub weights of the hubs that point to that node. Then the authority and hub weights are computed as follows:

$$a_i = \left(\sum_{j \in B(i)} h_j^p\right)^{1/p} \qquad \text{and} \qquad h_i = \left(\sum_{j \in F(i)} a_j^p\right)^{1/p} .$$

Let DOUBLENORM(p) denote this algorithm. The behavior of this symmetric version is particularly intriguing. First consider the limiting case $p = \infty$. When all initial weights are set to 1, then it is easy to see that all nodes will receive weight 1. If we initialize the authority weights to some other configuration then the algorithm assigns the authority weights as follows. Recall (Section 2.1) that the authority graph G_a is defined on the set of authorities A, and that there exists an (undirected) edge between two authorities if they have a hub in common. For every component in the graph G_a, all nodes in the component receive the weight of the node with the maximum initial weight in the component.

For $1 \leq p < \infty$, we will prove that the DOUBLENORM(p) algorithm converges and it produces the same ranking as the HITS algorithm. For the following, fix p, and assume for a moment that the DOUBLENORM(p) algorithm converges. Now, let $a_q(i)$ denote the authority weight of node i assigned by the DOUBLENORM(p) algorithm when the normalization is performed in the L_q norm. Also, let $w_q(i)$ denote the authority weight of node i assigned by the HITS algorithm, when the normalization is performed in the L_q norm. We prove the following.

Theorem 3.1 *The* DOUBLENORM*(p) algorithm converges, and* $a_q(i) = \big(w_{q/p}(i)\big)^{1/p}$.

Proof: We will prove the theorem using induction on the number of iterations. Let $a_q^t(i)$ denote the authority weight of node i, after t iterations of the DOUBLENORM(p) algorithm. Similarly, for the HITS algorithm let $w_q^t(i)$ denote the corresponding quantity. Let $a^0(i)$ denote the initial weight of node i for the DOUBLENORM(p) algorithm. We initialize the weight of the i-th node in the HITS algorithm to $w^0(i) = (a^0(i))^p$. We note that the convergence of the HITS algorithm does not depend on the initialization. We will prove that, for all $t \geq 0$, $a_q^t(i) = \left(w_{q/p}^t(i)\right)^{1/p}$.

For $t = 0$, it is obviously true (the initial weight does not depend on the normalization). Assume that it is true at the end of iteration t. Let h_j and g_j denote the hub weight of node j assigned by the DOUBLENORM(p) and the HITS algorithm respectively at iteration $t+1$. We have that

$$\begin{aligned}(h_j)^p &= \sum_{i\in F(j)} \left(a_q^t(i)\right)^p \\ &= \sum_{i\in F(j)} w_{q/p}^t(i) \\ &= g_j\end{aligned}$$

Let $\overline{a}^{t+1}(i)$ and $\overline{w}^{t+1}(i)$ denote the authority weights of node i before applying the normalization step, for the DOUBLENORM(p) and HITS algorithms, respectively. We have that

$$\overline{w}^{t+1}(i) = \sum_{j\in B(i)} g_j \ ,$$

and

$$\overline{a}^{t+1}(i) = \left(\sum_{j\in B(i)} (h_j)^p\right)^{1/p}$$

$$
\begin{aligned}
&= \left(\sum_{j\in B(i)} g_j\right)^{1/p} \\
&= \left(\overline{w}^{t+1}(i)\right)^{1/p}
\end{aligned}
$$

Therefore, we have that

$$
\begin{aligned}
a_q^{t+1}(i) &= \frac{\overline{a}^{t+1}(i)}{\left(\sum_{i=1}^n \left(\overline{a}^{t+1}(i)\right)^q\right)^{1/q}} \\
&= \frac{\left(\overline{w}^{t+1}(i)\right)^{1/p}}{\left(\sum_{i=1}^n \left(\overline{w}^{t+1}(i)\right)^{q/p}\right)^{1/q}} \\
&= \left(\frac{\overline{w}^{t+1}(i)}{\left(\sum_{i=1}^n \left(\overline{w}^{t+1}(i)\right)^{q/p}\right)^{p/q}}\right)^{1/p} \\
&= \left(w_{q/p}^{t+1}(i)\right)^{1/p}
\end{aligned}
$$

As $t \to \infty$, $w_{q/p}^t(i) \to w_{q/p}(i)$. Therefore, the DoubleNorm(p) algorithm converges, and $a_q(i) = \left(w_{q/p}(i)\right)^{1/p}$. Note that the convergence of DoubleNorm(p) follows from the convergence of the Hits algorithm, and thus it does not depend on the initial authority weights. □

From Theorem 3.1 we observe that, although the authority weights produced by the DoubleNorm(p) and Hits algorithms may be significantly different, the actual rankings (orderings of pages) are identical. Let a_i and w_i denote the authority weight for node i produced by algorithms DoubleNorm(p) and Hits respectively. Then, for any two nodes i, j, $a_i < a_j$ if and only if $w_i < w_j$. This is independent of the normalization, and it holds for all $1 \le p < \infty$. Thus, we have the following surprising phenomenon. For any $1 \le p < \infty$, the DoubleNorm(p) algorithm produces the same ranking as Hits, but for $p = \infty$, the DoubleNorm(p) algorithm becomes the uniform algorithm that assigns the same weight to all nodes. However, this can be explained by the following observation. For any two nodes i and j, we have that

$$
\frac{a_q(i)}{a_q(j)} = \frac{\left(w_{q/p}(i)\right)^{1/p}}{\left(w_{q/p}(j)\right)^{1/p}} = \left(\frac{w_{q/p}(i)}{w_{q/p}(j)}\right)^{1/p} .
$$

The ratio $\frac{w_{q/p}(i)}{w_{q/p}(j)}$ is independent of p and q. Thus, as p grows, the ratio $\frac{a_q(i)}{a_q(j)}$ converges to 1. We discuss similarity between algorithms more in Chapter 5.

3.5 The Breadth-First-Search (BFS) Algorithm

In this section we introduce a link analysis ranking algorithm that combines ideas from both InDegree and Hits algorithms. The InDegree algorithm computes the authority weight of a page taking into account only the popularity of this page within its immediate neighborhood, and disregarding the rest of the graph. On the other hand, the Hits algorithm considers the whole graph, taking into account the structure of the graph around the node, rather than just the popularity of that node in the graph.

Let B denote a path that consists of a single edge that we follow backwards, and let F denote a path that consists of a single forward edge. We combine these to obtain longer paths. For example, a $(BF)^n$ path is a path that alternates between backward and forward links n times. If we assume that the normalization in Hits is performed in the L_1 norm, then after n iterations of the Hits algorithm the authority weight of authority i is $|(BF)^n(i)|/|(BF)^n|$, where $|(BF)^n(i)|$ is the number of $(BF)^n$ paths that leave node i, and $(BF)^n$ denotes the set of all $(BF)^n$ paths in the graph. Another way to think of this is that the contribution of a node $j \neq i$ to the weight of i is equal to the number of $(BF)^n$ paths that go from i to j. Therefore, if nodes j and i belong to a bipartite component, their weights increase exponentially fast. This may not always be desirable, especially if the bipartite component is not representative of the query.

We now describe the Breadth-First-Search (BFS) algorithm (first proposed in the collaborative work with A. Borodin, G. Roberts, and J. Rosenthal [10]), as a generalization of the InDegree algorithm, and a restriction of the Hits algorithm. The BFS algorithm extends the idea of popularity that appears in the InDegree algorithm from a one-link neighborhood to an n-link neighborhood. The construction of the n-link neighborhood is inspired by the Hits algorithm. However, instead of considering the number of $(BF)^n$ *paths* that leave i, it considers the number of $(BF)^n$ *neighbors* of node i. Abusing the notation, let $(BF)^n(i)$ denote the set of nodes that can be reached from i by following a $(BF)^n$ path. The contribution of node j to the weight of node i depends on the distance of the node j from i. We adopt an exponentially decreasing weighting scheme. Therefore, the weight of node i is determined as follows:

$$a_i = |B(i)| + \frac{1}{2}|BF(i)| + \frac{1}{2^2}|BFB(i)| + \ldots + \frac{1}{2^{2n-1}}|(BF)^n(i)| + \frac{1}{2^{2n}}|(BF)B^n(i)|.$$

The algorithm starts from node i, and visits its neighbors in BFS order, alternating between backward and forward steps. Every time we move one link further from the starting node i, we update the weight factors accordingly. The algorithm stops either when n links have been traversed, or the nodes that can be reached from node i are exhausted.

The idea of applying an exponentially decreasing weighting scheme to *paths* that originate from a node has been previously considered by Katz [55]. In the algorithm of Katz, for some fixed parameter $\alpha < 1$, the weight of node i is equal to

$$a_i = \sum_{k=1}^{\infty} \sum_{j=1}^{n} \alpha^k W^k[j,i]$$

where W^k is the k-th power of the adjacency matrix W. The entry $W^k[j,i]$ is the number of paths in the graph G of length k from node j to node i. As we move further away from node i, the contribution of the paths decreases exponentially. There are two important differences between BFS and the algorithm of Katz. First, the way the paths are constructed is different, since the BFS algorithm alternates between backward and forward steps. More important, the BFS algorithm and the algorithm of Katz is that the BFS algorithm considers the *neighbors* at distance k. Every node j contributes to the weight of node i just once, and the contribution of node j is $1/2^k$ (or α^k if we select a different scaling factor), where k is the shortest path from j to i. In the algorithm of Katz, the same node j may contribute multiple times, and its contribution is the number of paths that connect j with i. The algorithm of Katz resembles the PageRank algorithm, where the weight of node i can be expressed as the sum of paths leading to node i.

Chapter 4

Applications of Non-Linear Dynamical Systems to Link Analysis Ranking

In this chapter we examine the application of dynamical systems to Link Analysis Ranking. We are especially interested in dynamical systems that use non-linear operators. We study in detail one such system, prove that it converges, and give a characterization of the combinatorial properties of the weights that the algorithm produces.

4.1 Dynamical Systems

A discrete dynamical system is defined [25] as a process that starts with an n-dimensional real vector, and repeatedly applies a function $g : \mathbb{R}^n \to \mathbb{R}^n$. We define a *configuration* of the system as any intermediate value of the vector. The initial assignment of values is called the *initial configuration* of the dynamical system. Of particular interest are the *fixed configurations* (or fixed points). These are vectors $\boldsymbol{x}$, such that $g(\boldsymbol{x}) = \boldsymbol{x}$. We will also refer to these vectors as the *stationary* configurations of the dynamical system. An interesting question in dynamical systems is the limiting value of the dynamical system. That is, if $g^t(\boldsymbol{x})$ denotes the t-th iteration of the function g, then we are interested in understanding the limiting behavior of $g^t(\boldsymbol{x})$, as $t \to \infty$, for different initial values of $\boldsymbol{x}$. For an introduction to dynamical systems, we refer the reader to the texts by Denavey [25] , Sandefur [81], and

Holmgren [50].

In the case of link analysis algorithms, the real vector is the authority weight vector, and the function g propagates the authority weight in the graph G. Let $\boldsymbol{a}^t$ denote the authority weight vector after t iterations of the algorithm. The outline of a dynamical system for link analysis ranking is shown in Figure 4.1. We note that, alternatively, we could define the output of an LAR algorithm as a fixed point of the function g. The dynamical system provides one possible way for computing such fixed points.

```
DynamicalSystem_g ($\boldsymbol{a}^0$)

Initialize authority weights to $\boldsymbol{a}^0$
Repeat until the weights converge:
        $\boldsymbol{a}^t = g(\boldsymbol{a}^{t-1})$
```

Figure 4.1: The outline of a dynamical system

Depending on the function g we distinguish between two types of dynamical systems: *linear* and *non-linear*. In linear dynamical systems, the function g is of the form $g(\boldsymbol{x}) = M\boldsymbol{x}$, where M is an $n \times n$ matrix. The majority of the link analysis algorithms that have appeared so far in the literature [58, 13, 64, 78, 10, 1, 73] can be described as linear dynamical systems. Table 4.1 shows the linear function that corresponds to the PageRank, Salsa, Hits, and HubAvg. The functions for Hits and HubAvg are referred to in the literature as *linear automorphisms* [25]. We can transform a linear automorphism into a linear system by dividing the entries of the corresponding matrix M by its principal eigenvalue [81].

Algorithm	Function
PageRank	$g_{PR}(\boldsymbol{a}) = M_{PR}^T \boldsymbol{a}$
Salsa	$g_S(\boldsymbol{a}) = M_S^T \boldsymbol{a}$
Hits	$g_H(\boldsymbol{a}) = \frac{M_H \boldsymbol{a}}{\lVert M_H \boldsymbol{a} \rVert}$
HubAvg	$g_{HA}(\boldsymbol{a}) = \frac{M_{HA} \boldsymbol{a}}{\lVert M_{HA} \boldsymbol{a} \rVert}$

Table 4.1: LAR algorithms as Linear dynamical systems

Consider now a linear dynamical system with matrix M. We assume that the matrix M has positive entries, non-negative non-defective[1] real eigenvalues, and principal eigenvalue 1. For such linear dynamical systems, linear algebra offers the tools to analyze the limiting behavior of the system. If the principal eigenvalue is unique, then if we initialize the system to some initial configuration $\boldsymbol{x}^0$ that is not orthogonal to the principal eigenvector of the matrix the dynamical system converges to a fixed configuration, which is the principal eigenvector of the matrix M that defines the system. If the principal eigenvalue is not unique, then the system converges to a linear combination of the principal eigenvectors. When $\boldsymbol{x}^0$ is orthogonal to the principal eigenvector of the matrix, the system converges to the highest-indexed eigenvector to which it is not initially orthogonal.

For the linear systems we consider, the PageRank, Salsa, Hits, and HubAvg algorithms, the conditions for the matrices of the systems are satisfied and the convergence of the systems is guaranteed. For the PageRank and Salsa algorithms, the convergence follows from the fact that the M_{PR} and M_S matrices are stochastic matrices. For the Hits and HubAvg the convergence follows from the fact that the M_H and M_{HA} matrices are symmetric matrices.

Things become more complicated when the function g is non-linear. Outside of the well understood world of linear algebra, we know very little about the behavior of dynamical systems. In Link Analysis Ranking, non-linear dynamical systems arise when we apply non-linear operators for the computation of the weights. Examples of non-linear dynamical systems that are applied to ranking include the AT(k) and Norm(p) families of algorithms, since the threshold and p-norm operators that are applied for the computation of the hub weights are non-linear. For these algorithms we do not even know if they converge, which is a basic requirement for a well defined Link Analysis Ranking algorithm.

In the following we study in detail the Max algorithm, a special case of both the AT(k) and Norm(p) families, where we set the hub weight to be the maximum authority weight of the nodes pointed to by the hub. This algorithm was previously considered by Gibson, Kleinberg and Raghavan [39], but it was not rigorously analyzed. We prove that the algorithm converges, and we characterize the combinatorial properties of the stationary configuration.

[1] A defective eigenvalue is one that has geometric multiplicity (number of associated eigenvectors) less than its algebraic multiplicity (number of times the eigenvalue is repeated).

```
MAX($\boldsymbol{a}^0$)

Initialize authority weights to $\boldsymbol{a}^0$
Repeat until the weights converge:
    For every hub $i \in H$
        $h_i = \max_{j \in F(i)} a_j$
    For every authority $i \in A$
        $a_i = \sum_{j \in B(i)} h_j$
    Normalize in the $L_\infty$ norm
```

Figure 4.2: The MAX Algorithm

4.2 The MAX algorithm

The MAX algorithm is a special case of both the AT(k) algorithm for the threshold value $k = 1$, and the NORM(p) algorithm for the value $p = \infty$. The underlying intuition is that a hub node is as good as the best authority that it points to. That is, a good hub is one that points to at least one good authority.

Formally, we define the MAX algorithm as follows. The algorithm sets the hub weight of node i to be the maximum authority weight over all authority weights of the nodes pointed to by node i. The authority weights are computed as in the HITS algorithm. Therefore, the authority and hub weights as computed as follows.

$$a_i = \sum_{j \in B(i)} h_j \qquad \text{and} \qquad h_i = \max_{j \in F(i)} a_j \ .$$

The outline of the algorithm is shown in Figure 4.2. We set the normalization norm to be the *max* (or infinity) norm. This makes the analysis easier, but it does not affect the convergence, and the combinatorial properties of the stationary configuration.

4.3 Preliminaries

We will now introduce some of the terminology that we will use in the remainder of this chapter. We first define a notion of *time*. We define time t to be the moment immediately after the t-th iteration of the algorithm. We denote by $\overline{a}_i^t$ the un-normalized weight of node i at time t. The weights are normalized in the L_∞ norm. This means that the normalization factor at step t is the maximum un-normalized authority weight, and the maximum normalized weight is 1. We use a_i^t to denote the normalized weight of node i at time t. When not specified, the weight of node i at time t refers to the normalized weight of node i at time t. We denote $a_i = \lim_{t\to\infty} a_i^t$, the limit of the weight of node i, as $t \to \infty$, assuming that the limit exists. When convenient we will use $\overline{a}^t(i)$, $a^t(i)$, and $a(i)$ for the quantities $\overline{a}_i^t$, a_i^t, and a_i respectively.

We also define the mapping $f^t : H \to A$, where the hub j is mapped to authority i, if at time t the authority i is the authority with the maximum weight among all the authorities pointed to by hub j. If there are many authorities in $F(j)$ that have the largest weight, we arbitrarily select one of the authorities (e.g., according to some predefined ordering). We use $f(j) = \lim_{t\to\infty} f^t(j)$ to denote the limit of the mapping function as $t \to \infty$, assuming again that the limit exists. For an authority i, the un-normalized weight of node i at time t is $\overline{a}_i^t = \sum_{j\in B(i)} a^{t-1}\left(f^{t-1}(j)\right)$.

Recall that $G = (P, E)$ denotes the underlying graph, and $G_a = (A, E_a)$ the authority graph, where there exists an undirected edge between two authorities if they share a hub. Assume now that the graph G_a consists of k connected components $C_1, C_2, \ldots, C_k$. Let $\boldsymbol{a}^0$ be the weight vector of the initial configuration. The weight assigned by configuration $\boldsymbol{a}^0$ to component C_i is the sum of weights of all authorities in C_i. We define a *fair* initial configuration as a configuration that assigns non-zero weight to all components in the graph G_a. We will assume that the initial configuration is always fair. If the component C_i is assigned zero weight by the vector $\boldsymbol{a}^0$, then the weights of the nodes in C_i will immediately converge to zero. Thus, we can disregard the nodes in the component C_i and assume that the algorithm operates on a smaller graph $\tilde{G}$, initialized to a fair configuration $\tilde{\boldsymbol{a}}^0$.

Finally, for some node $i \in A$, let d_i denote the in-degree of authority i in the graph G, and let $d = \max\{d_i : i \in A\}$, denote the maximum in-degree of any authority in the graph G. Let $S \subseteq A$ denote the set of nodes with in-degree d. We call these nodes, the *seeds* of the algorithm. Seed nodes play an important role in the MAX algorithm. We define U to be the set of non-seed nodes. Thus, $A = S \cup U$.

4.4 Convergence of the MAX algorithm

In this section we prove that the algorithm converges for any initial configuration. First, we prove that the weights of the seed nodes always converge.

Lemma 4.1 *The weight of every seed node $s \in S$ is a non-decreasing function of time.*

Proof: Consider any seed node $s \in S$, and let $t \geq 0$ be some time in the execution of the algorithm. At iteration $t+1$, for every hub node j, we have that $h_j = \max\{a_i^t : i \in F(j)\}$. Thus, for every hub $j \in B(s)$, $h_j \geq a_s^t$. Therefore, at time $t+1$, the un-normalized weight of node s is $\overline{a}_s^{t+1} = \sum_{j \in B(s)} h_j \geq d a_s^t$. Let x be the authority node with maximum un-normalized weight at time $t+1$. Since the weight of any authority at time t is at most 1, we have that for every hub $j \in B(x)$, $h_j \leq 1$. It follows that $\overline{a}_x^{t+1} = \sum_{j \in B(x)} h_j \leq d_x \leq d$. Therefore, after normalization

$$a_s^{t+1} \geq \frac{d}{\overline{a}_x^{t+1}} a_s^t \geq a_s^t$$

which concludes the proof. □

Corollary 4.1 *The weight of every seed node $s \in S$ converges for any initial configuration.*

Proof: For every seed node $s \in S$, the weight of s is a non-decreasing function that is upper-bounded, therefore it will converge. □

Note that "non-decreasing" means that the weights either increase, or remain constant. We now prove that there always exists a seed node with stationary weight 1.

Lemma 4.2 *For every initial configuration there exists a point in time t_0, such that for some seed node $s \in S$, $a_s^t = 1$, for all $t \geq t_0$.*

Proof: Assume that for every $s \in S$, $a_s^t < 1$, for all $t \geq 0$. We will then prove that for every $s \in S$, $a_s^t \geq a_s^0 d^t/(d-1)^t$ by induction on t. For $t = 0$ it is trivially true. Now, assume that at time t, $a_s^t \geq a_s^0 d^t/(d-1)^t$. At time $t+1$, we have (as in the proof of Lemma 4.1) that the un-normalized weight of s is $\overline{a}_s^{t+1} \geq d a_s^t$. Let x be the node with maximum un-normalized weight at time $t+1$. Node x cannot be a seed node, since then we would have that $a_x^{t+1} = 1$, reaching a contradiction with our initial assumption. Since x is not a seed node, $\overline{a}_x^t \leq d_x \leq d-1$. Normalizing the weight of s by $\overline{a}_x^t$ we have that

$$a_s^{t+1} \geq \frac{d}{(d-1)} a_s^t \geq \frac{d^{t+1}}{(d-1)^{t+1}} a_s^0 \ .$$

Therefore, the weight of every seed node is an increasing function of time. As $t \to \infty$, $a_s^t \to \infty$. Since, the weights are bounded, we reach a contradiction. Therefore, there must exist some point in time t_0 such that, for some node $s \in S$, $a_s^{t_0} = 1$. From Lemma 4.1 we know that the weight of the seed nodes is a non-decreasing function, thus, $a_s^t = 1$, for all $t \geq t_0$. Therefore, for this seed node, the weight increases until it becomes 1, and then it remains constant for the remaining iterations. □

For the following, given an *accuracy constant* δ, we say that the weight of some node i has converged at time t_i if $|a_i^{t+1} - a_i^t| \leq \delta$, for all $t \geq t_i$.[2] Corollary 4.1 and Lemma 4.2 guarantee that the seed nodes will converge, and at least one of the seeds will converge to weight 1. Let t_0 denote the first time that all seed nodes have converged, and let s be a seed node with weight 1. For $t \geq t_0$, the un-normalized weight of s is d. Furthermore, it is easy to see that for every other authority i, $\overline{a}_i^t \leq \overline{a}_s^t$. Therefore, for all $t \geq t_0$, the normalization factor $\|\overline{\boldsymbol{a}}^t\|_\infty$ is equal to d, the maximum in-degree of graph G, independent of the vector $\boldsymbol{a}^t$.

We are now ready to consider the convergence of the Max algorithm. The proof proceeds roughly as follows. We first prove that as $t \to \infty$ the configuration $\boldsymbol{a}^t$ of the Max algorithm is independent of the weights of the non-seed nodes at time t_0, and depends solely on the stationary weights of the seeds. Then, we set the weights of the non-seed nodes to zero at time t_0 and we prove that in this case the system converges. The fact that the configuration is independent of the non-seed weights implies that the system converges for any configuration of the non-seed nodes, which in turn implies convergence of the Max algorithm. However, "setting" the weights of the non-seed nodes to zero, is not simple to do without disrupting the Max algorithm. To this end, we need to introduce an auxiliary system Aux.

The system Aux is defined with two parameters. The first is the initial configuration $\boldsymbol{a}^0$ of the Max algorithm. The second is a weight vector $\boldsymbol{u}$ for the non-seed nodes in U. Given some configuration vector $\boldsymbol{v}$, we use $\boldsymbol{v}_S$ to denote the projection of $\mathbf{v}$ on the seed nodes S, and $\boldsymbol{v}_U$ to denote the projection of $\boldsymbol{v}$ on the non-seed nodes U. For the following we use $\boldsymbol{a}^t$ to denote the weight vector of the Max algorithm at time t, and $\boldsymbol{x}^t$ to denote the weight vector of the system Aux at time t. The structure of Aux is given in Figure 4.3.

The system Aux sets the weights of the seed nodes to the stationary weights of the Max algorithm when run on the initial configuration $\boldsymbol{a}^0$, and it updates the weights of

[2] Any other method for testing convergence is applicable. Our analysis does not depend on the definition of convergence.

```
Aux($\boldsymbol{a}^0$, $\boldsymbol{u}$)

Run the Max algorithm on $\boldsymbol{a}^0$
Let $t_0$ be the time that the seed nodes converge
$\boldsymbol{x}_S^0 = \boldsymbol{a}_S^{t_0}$   $\boldsymbol{x}_U^0 = \boldsymbol{u}$
Repeat until the weights converge:
    For every hub $i \in H$
        $h_i = \max_{j \in F(i)} x_i^t$
    For every authority $i \in U$
        $x_i^{t+1} = \sum_{j \in B(i)} h_j$
    For every authority $s \in S$
        $x_s^{t+1} = a_s^{t_0+t+1}$
    Normalize in the $L_\infty$ norm
```

Figure 4.3: The Aux dynamical system

the non-seed nodes in the regular fashion. Note that if $\boldsymbol{u} = \boldsymbol{a}_U^{t_0}$, then for every node i, $x_i^t = a_i^{t+t_0}$, for all $t \geq 0$. That is, Aux($\boldsymbol{a}^0$,$\boldsymbol{a}_U^{t_0}$) and Max($\boldsymbol{a}^0$) are equivalent; the system Aux converges if and only if the system Max converges. The Aux system serves the purpose of "disconnecting" the seed nodes from the non-seed nodes. This will become clear in the following.

We will now prove that in the limit the configuration of Aux is independent of the initial configuration $\boldsymbol{u}$ of the non-seed nodes. To assist the proof, we introduce the following conventions. We assume that at the initialization of the Aux system, each node i receives an amount of *mass* μ_i^0 of *color* i. The weight of this mass is x_i^0, where a unit of mass corresponds to a unit of weight. That is, there is a one to one correspondence between mass and weight, except for the fact that mass has color. As mass is moved around in the graph, by measuring the amount of mass of color i at time t, we can quantify the contribution of the initial weight of authority i to the configuration $\boldsymbol{x}^t$at time t.

Consider the Aux system at time $t-1$. Recall that the function f^{t-1} maps every hub j to the authority i which at time $t-1$ has the maximum weight among all authorities in $F(j)$. We take the following view of the t-th iteration. Every authority i sends its mass to all hubs that map to i at time $t-1$ (assuming that mass can be replicated). Consider a hub j, for which $f^{t-1}(j) = i$. The hub j receives the mass of the authority i, and sends it to all the authorities in $F(j)$, *except* the seed nodes in S. Every seed node $s \in S$ receives mass of color s, with weight $a_s^{t+t_0}$. Non-seed authority i receives mass from every hub in $B(i)$. The weight of i is the total weight of all the mass it receives. If node i receives μ units of mass of color k, we say that node i *contains* μ units of mass of color k. We use μ_k^t to denote the total mass of color k in the system at time t. The amount of mass of color k contained in node i at time t is the contribution of the initial weight x_k^0 of node k to the weight x_i^t of node i, at time t.

We are now ready to prove the following lemma.

Lemma 4.3 *For every non-seed node $k \in U$ in the* Aux *system, as $t \to \infty$, $\mu_k^t \to 0$.*

Proof: First, we note that by definition of the Aux system, no seed node ever receives mass of color k, for all $k \in U$. We will prove that for all $t \geq 0$, every authority $i \in U$ contains at most $\mu_k^0(d-1)^t/d^t$ units of mass of color k. For $t = 0$ the claim is trivially true. Assume that it is true at time t. At the iteration $t+1$, the hub j receives the mass of the authority p, such that $f^t(j) = p$. By the inductive hypothesis, every authority contains at most $\mu_k^0(d-1)^t/d^t$ units of mass of color k; therefore, after this first step of the iteration every hub j contains at most $\mu_k^0(d-1)^t/d^t$ units of mass of color k.

Consider now some authority $i \in U$. Authority i receives the mass of $d_i \leq d-1$ hubs. Since every hub contains at most $\mu_k^0(d-1)^t/d^t$ units of mass of color k it follows that at the end of iteration $t+1$ authority i contains at most $\mu_k^0(d-1)^{t+1}/d^t$ units of mass of color k. At the normalization step, the mass at every authority is scaled by a factor $1/d$. Thus, at the end of iteration $t+1$, authority i contains at most $\mu_k^0(d-1)^{t+1}/d^{t+1}$ units of mass of color k.

Therefore, the total mass of color k in the graph at time t is at most $\mu_k^t = |A|\mu_k^0(d-1)^t/d^t$, where $|A|$ is the number of authorities. Thus, as $t \to \infty$, $\mu_k^t \to 0$. □

Corollary 4.2 follows immediately from Lemma 4.3.

Corollary 4.2 *The configuration $\lim_{t\to\infty} \boldsymbol{x}^t$ of the* Aux *system is independent of the initialization vector $\boldsymbol{u}$.*

Let $\mathbf{0}$ denote the vector of all zeros. We now prove the following lemma.

Lemma 4.4 *The system* Aux$(\boldsymbol{a}^0, \mathbf{0})$ *converges for any configuration $\boldsymbol{a}^0$.*

Proof: We will prove that the weights of all authorities in the system are non-decreasing functions of time. Since the weights are upper bounded it follows that they will converge.

For every seed node $s \in S$, $x_s^t = a_s^{t+t_0}$, that is, the weight of the seed nodes in the AUX system at time t is the same with the weight of the seed nodes in the MAX system at time $t + t_0$. From Lemma 4.1 we know that for the MAX algorithm, the weights of all seed nodes are non-decreasing functions of time. Therefore, x_s^t is a non-decreasing function of time, for all $s \in S$.

We will now prove that for every authority $i \in U$, $x_i^t \geq x_i^{t-1}$ for all $t \geq 1$, using induction on time. For $t = 1$, $x_i^1 \geq 0$, so the claim is trivially true. Assume that it is true at time t. Consider now the difference $\overline{x}_i^{t+1} - \overline{x}_i^t$. We break up the hubs in $B(i)$ into two sets. The set V contains the hubs $j \in B(i)$ such that $f^t(j) = f^{t-1}(j)$; that is, the hubs whose mapping does not change at time t. The set W contains the hubs $j \in B(i)$ such that $f^t(j) \neq f^{t-1}(j)$, that is, the hubs whose mapping changes at time t.

We have that $\overline{x}_i^{t+1} - \overline{x}_i^t = S_1 + S_2$, where

$$S_1 = \sum_{j \in V} \left(x^t \left(f^t(j) \right) - x^{t-1} \left(f^{t-1}(j) \right) \right) \quad \text{and} \quad S_2 = \sum_{j \in W} \left(x^t \left(f^t(j) \right) - x^{t-1} \left(f^{t-1}(j) \right) \right) .$$

For every $j \in V$, there exists $p \in A$ such that $f^t(j) = f^{t-1}(j) = p$. By the inductive hypothesis we have that $x_p^t - x_p^{t-1} \geq 0$. Therefore, $S_1 \geq 0$. For every $j \in W$, there exist $p, q \in A$ such that $f^t(j) = p$, and $f^{t-1}(j) = q$. Since at time t the mapping of the hub j switches from q to p, it follows that $x_p^t > x_q^t$, and $x_p^{t-1} \leq x_q^{t-1}$ (or $x_p^t \geq x_q^t$, and $x_p^{t-1} < x_q^{t-1}$ depending on the way that we break the ties). By the induction hypothesis we have that $x_q^t \geq x_q^{t-1}$. Therefore, $x_p^t - x_q^{t-1} \geq x_p^t - x_q^t \geq 0$. Thus, $S_2 \geq 0$, and $\overline{x}_i^{t+1} - \overline{x}_i^t \geq 0$. Since $x_i^{t+1} - x_i^t = (\overline{x}_i^{t+1} - \overline{x}_i^t)/d$, it follows that $x_i^{t+1} \geq x_i^t$. □

Theorem 4.1 *The* MAX *algorithm converges for any initial configuration. The stationary configuration of* MAX *is determined by the stationary weights of the seed nodes.*

Proof: For any initial configuration $\boldsymbol{a}^0$ the system AUX$(\boldsymbol{a}^0, \mathbf{0})$ converges. From Corollary 4.2 the limiting behavior of AUX is independent of the initial configuration of the non-seed nodes. Therefore, for any vector $\boldsymbol{u}$, the AUX$(\boldsymbol{a}^0, \boldsymbol{u})$ system will converge, and it will converge to the same vector as AUX$(\boldsymbol{a}^0, \mathbf{0})$. When $\boldsymbol{u} = \boldsymbol{a}_U^{t_0}$, the system AUX$(\boldsymbol{a}^0, \boldsymbol{a}_U^{t_0})$ is equivalent to the MAX$(\boldsymbol{a}^0)$ algorithm. Therefore, the MAX algorithm converges for any initial configuration. From Corollary 4.2 it follows that the stationary configuration depends only on the weights of the seed nodes at time t_0. □

We are particularly interested in the *uniform* initial configuration, when all nodes are initialized to the same weight. Since the configuration is a unit vector in the L_∞ norm all nodes are initialized to weight 1. In this case from Lemma 4.1, we know that the weight of the seed nodes will immediately converge to 1. Corollary 4.3 follows directly from Theorem 4.1.

Corollary 4.3 *For every initial configuration that assigns weight 1 to all seed nodes, the* Max *algorithm converges to the same weight vector as when initialized to the uniform configuration.*

In the case of the uniform initial configuration we have a very clear characterization of the *rate of convergence* of the algorithm. In this case, the seed nodes converge immediately to weight 1. Given an accuracy constant δ, the Max algorithm converges when the mass of the non-seed nodes becomes less than δ. Let d' denote the second-highest in-degree in the graph. As we saw in Lemma 4.3, for every non-seed node k, after t iterations, the mass of color k is equal to $\mu_k^t \leq |A|(d'/d)^t$. We have that $|A|(d'/d)^t \leq \delta$, if

$$t \geq \frac{\log(|A|/\delta)}{\log(d/d')}$$

Thus, the rate of convergence depends upon the size of the graph, and the ratio between the highest, and second-highest in-degree in the graph.

4.5 The stationary configuration

In this section we give a better understanding of the way the Max algorithm assigns the weights to the authorities. We first introduce the auxiliary graph G_A. Assume that the algorithm has converged, and let a_i denote the stationary weight of node i. Define $H(i) = \{j \in H : f(j) = i\}$ to be the set of hubs that are mapped to authority i. Recall that the authority graph G_a defined in Section 2.1 is an undirected graph, where we place an edge between two authorities if they share a hub. We now derive the *directed weighted* graph $G_A = (A, E_A)$ on the authority nodes A, from the authority graph G_a as follows. Let i and j be two nodes in A, such that there exists an edge (i,j) in the graph G_a, and $a_i \neq a_j$. Let $B(i,j) = B(i) \cap B(j)$ denote the set of hubs that point to both authorities i and j. Without loss of generality assume that $a_i > a_j$. If $H(i) \cap B(i,j) \neq \emptyset$, that is, there exists

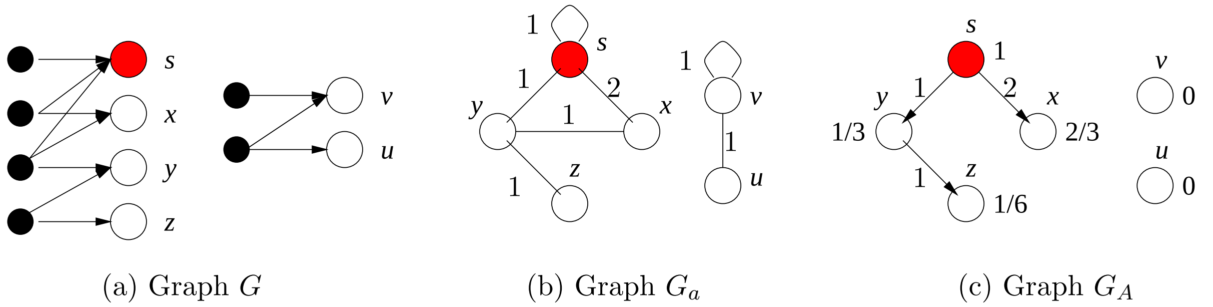


Figure 4.4: Graphs G, G_a, and G_A.

at least one hub in $B(i,j)$ that is mapped to the authority i, then we place a directed edge from i to j. The weight $c(i,j)$ of the edge (i,j) is equal to the size of the set $H(i) \cap B(i,j)$, that is, it is equal to the number of hubs in $B(i,j)$ that are mapped to i. The intuition of the directed edge (i,j) is that there are $c(i,j)$ hubs that propagate the weight of node i to node j. The graph G_A captures the flow of authority weight between authorities.

Now, let $N(i)$ denote the set of nodes in G_A that point to node i. Also, let $c_i = \sum_{j \in N(i)} c(j,i)$, denote the total weight of the edges that point to i in the graph G_A. This is the number of hubs in the graph G that point to i, but are mapped to some node with weight greater than i. The remaining $d_i - c_i$ hubs (if any) are mapped to node i, or to some node with weight equal to the weight of i. We set $b_i = d_i - c_i$. The number b_i is also equal to the size of the set $H(i)$, the set of hubs that are mapped to node i, when all ties are broken in favor of node i.

An example of the graphs G, G_a, and G_A is shown in Figure 4.4. Every edge $\{i,j\}$ in the graph G_a is tagged with the number of hubs $B(i) \cap B(j)$ that point to both i and j nodes. The numbers next to the nodes of graph G_A are the stationary weights, and the weights on the edges are the $c(i,j)$ values.

The following proposition gives a recursive formula for weight a_i, given the weights of the nodes in $N(i)$.

Proposition 4.1 *The weight of node i satisfies the equation*

$$a_i = \sum_{j \in N(i)} c(j,i) a_j / d + b_i a_i / d \ .$$

Proof: Recall that for every node i, $a_i = \sum_{j \in B(i)} a\left(f(j)\right)/d$. From the hubs in $B(i)$, b_i of them are mapped to node i, or to some node with weight equal to a_i. These hubs recycle the weight of node i, and they contribute weight $b_i a_i/d$ to the weight of node i. The remaining hubs bring in the weight of some other authority. For every $j \in N(i)$, there are $c(j,i)$ hubs in $B(i)$ that are mapped to node j. These hubs propagate the weight a_j of node j to node i. Thus, they collectively contribute weight $c(j,i)a_j/d$ to the weight of node i. Therefore, we have that

$$a_i = \sum_{j \in N(i)} c(j,i)a_j/d + b_i a_i/d \ .$$

□

By definition, the graph G_A is a DAG. Therefore, there must exist some nodes, such that no node in G_A points to them. We define a *source node* in the graph G_A to be a node x, such that $N(x) = \emptyset$ (i.e., there is no node in G_A that points to x), and $a_x > 0$. Lemma 4.2 guarantees that at least one such node exists. In the example of Figure 4.4(c), there is only one source node, the seed node s. Nodes v and u have no incoming edges, but they are not source nodes, since they have zero weight. We now prove that the set of source nodes is identical to the set of the seed nodes.

Lemma 4.5 *A node is a source node of the graph G_A if and only if it is a seed node in the graph G.*

Proof: Let x be a source node of the graph G_A. Since $N(x) = \emptyset$, it follows that all d_x hubs that point to x are mapped to x, or to some node with weight equal to a_x. Therefore $b_x = d_x$. We have that $a_x = a_x d_x/d$. Since $a_x > 0$, it follows that $d_x = d$.

Let s be a seed node. Assume that s is not a source node in the graph G_A. Then, either $a_s = 0$, or $N(s) \neq \emptyset$. We have assumed that the initial configuration is a fair configuration, that is, the initial configuration assigns to every component of the graph G_a non-zero weight. If C_s is the component in the graph G_a that contains node s, then at least one node in C_s was initialized to non-zero weight. Therefore, there exists some point in time t_s such that the $a_s^{t_s} > 0$. From Lemma 4.1 we know that the weight of every seed node is a non-decreasing function of time, therefore, $a_s^t \geq a_s^{t_s}$ for all $t \geq t_s$. Therefore, $a_s > 0$.

Assume that $N(s) \neq \emptyset$. We have that

$$a_s = \sum_{i \in N(s)} c(i,s)a_i/d + b_s a_s/d \ .$$

For every $i \in N(s)$ we have that $a_i > a_s$. Therefore, it follows that

$$a_s = \sum_{i \in N(s)} c(i,s)a_i/d + b_s a_s/d \ > \ c_s a_s/d + b_s a_s/d = a_s d/d = a_s \ ,$$

thus reaching a contradiction. □

We now turn our attention to the non-seed nodes of the graph. For the following, we say that node i is *connected* to a seed node in the graph G_a if there exists a path in the graph G_a from a seed node to node i. We say that node i is *reachable* from a seed node in the graph G_A if there exists a directed path in the graph G_A from a seed node to the node i. We will often say that a node is reachable to indicate that it is reachable from a seed node in the graph G_A. In the example of Figure 4.4, nodes x, y, z are connected to, and reachable from the seed node s, while nodes u and v are neither connected to, nor reachable from seed node s.

Lemma 4.6 *A node i is reachable from a seed node in the graph G_A if and only if $a_i > 0$.*

Proof: We will prove that every reachable node has positive weight using induction on the length of the shortest path from a seed node to node i in the graph G_A. Let $radius_A(s,i)$ be the length of the shortest path from seed node s to node i in the graph G_A. Let $radius_A(i) = \min_{s \in S} radius_A(s,i)$ be the shortest path from any seed node to node i in the graph G_A. For every node i with $radius_A(i) = 0$, that is, the seed nodes themselves, the lemma is trivially true. Assume that it is true for every node i with $radius_A(i) \leq \ell$. Every node j with $radius_A(j) = \ell + 1$ must be connected to a node i with $radius_A(i) = \ell$. From Proposition 4.1 we have that $a_j \geq c(i,j)a_i/d > 0$.

Assume now that node $i \in U$ is not reachable from a seed node. If $N(i) = \emptyset$, then it must be that $a_i = 0$. Otherwise, node i is a source node. From Lemma 4.5 this is not possible, since node i is not a seed node. Assume now that $N(i) \neq \emptyset$, that is, there exists some node in the graph G_A that points to node i. Then starting from node i we can follow edges backwards in the graph G_A to other non-reachable nodes. Since the graph G_A contains no cycles, we will eventually find a node j that is not reachable, and has no incoming edges. Since j is not a seed node, we have that $a_j = 0$, and $a_j > a_i \geq 0$, thus reaching a contradiction. Therefore, there cannot be any node pointing to node i, and $a_i = 0$. □

Lemma 4.7 *A node is reachable from a seed node in the graph G_A if and only if it is connected to a seed node in the graph G_a.*

Proof: Obviously, by definition of the graphs G_a and G_A, if a node is not connected to seed node, then it is be reachable from a seed node. We will now prove that every node i, that is connected to a seed node in the graph G_a, it is also reachable from a seed node in the graph G_A, using induction on the length of the shortest path from a seed node to node i in the graph G_a. Let $radius_a(s,i)$ be the length of the shortest path from seed node s to i in the graph G_a. Let $radius_a(i) = \min_{s \in S} radius_a(s,i)$ be the shortest path from any seed node to node i in the graph G_a. For every node i such that $radius_a(i) = 0$, that is, the seed nodes themselves, the lemma is trivially true. Assume that it is true for every node i with $radius_a(i) = \ell$. Now consider some node j with $radius_a(j) = \ell + 1$. Since node j is connected to a seed node in the graph G_a, there exists a node i with $radius_a(i) = \ell$ such that the edge (i,j) belongs to graph G_a. This implies that there exits at least one hub h that points to both i and j. Let $f(h) = k$ be the mapping of this hub. Node k is not necessarily node i or j, and it is not necessarily the case that $radius_a(k) \leq \ell$. However, we know that hub h points to both i and k, and that $a_k \geq a_i$. By the inductive hypothesis, node i is reachable, so $a_i > 0$. Thus, $a_k > 0$.

Consider now the nodes j and k. If $a_j \geq a_k$, then $a_j > 0$, therefore, node j is reachable. Otherwise, for the pair (k,j) we have that: [a] there exists an edge (k,j) in the graph G_a (since the nodes j and k share the hub h); [b] $B(k,j) \cap H(k) \neq \emptyset$ (since the hub h is mapped to k); [c] $a_k > a_j$. Therefore, there must exist a directed edge (k,j) in the graph G_A. Since $a_k > 0$, Lemma 4.6 guarantees that node k is reachable from a seed node in G_A. Thus, node j is also reachable. □

For some node i, and some seed node s, we define $dist(s,i)$ to be the distance of the longest path in G_A from s to i. We define the distance of node i, $dist(i) = \max_{s \in S} dist(s,i)$, to be the maximum distance from a seed node to i, over all seed nodes. We note that the distance is well defined, since the graph G_A is a DAG. We now summarize the results of this section in the following theorem.

Theorem 4.2 *Given a graph G, let $C_1, C_2, \ldots C_k$ be the connected components of the graph G_a. For every component C_i, $1 \leq i \leq k$, if component C_i does not contain a seed node, then $a_x = 0$, for all x in C_i. If component C_i contains a seed node, then every node x in*

C_i is reachable from a seed node in C_i, and $a_x > 0$. Given the weights of the seed nodes, we can recursively compute the weight of a reachable (in the graph G_A) node x at distance $\ell > 0$, using the equation

$$a_x = \frac{1}{d - b_x} \sum_{j \in N(x)} c(j, x) a_j \ ,$$

where for all $j \in N(i)$, $dist(j) < \ell$.

Proof: Let C_i denote the i-th component of the graph G_a. Obviously, if a node is not connected to a seed node in graph G_a, it cannot be reachable from a seed node in the graph G_A. Therefore, if component C_i does not contain a seed node, then, from Lemma 4.6, for every x in C_i, $a_x = 0$. Assume now that the component C_i contains a seed node. Lemma 4.7 guarantees that every node x in C_i becomes reachable from a seed node in the graph G_A. The weight of node $x \in C_i$ can be computed recursively using Proposition 4.1. We have that

$$a_x = \sum_{j \in N(x)} c(j, x) a_j / d + b_x a_x / d \ .$$

Therefore,

$$a_x = \frac{1}{d - b_x} \sum_{j \in N(x)} c(j, x) a_j \ .$$

If node x is at distance ℓ, then all nodes $j \in N(x)$ have $dist(j) < \ell$. Therefore, starting from the seed nodes, we can iteratively compute the weights of all nodes at increasing distances. □

Theorem 4.2 is in agreement with our findings in the Section 4.4, where we observed that the stationary configuration depends solely on the stationary weights of the seed nodes. Note that Theorem 4.2 does not provide a constructive way of assigning weights to the nodes, since the graph G_A depends on the stationary configuration. However, it provides a useful insight in the mechanics of the algorithm, and in the way the weight is propagated from the seed nodes to the remaining authorities. All weight emanates from the seed nodes, and it floods the rest of the nodes, propagated in the graph G_A. As the distance from the seed nodes increases, the weight decreases exponentially by a scaling factor d. However, well connected nodes, and nodes with high in-degree in the graph G, reinforce their own weight. For node i, there are b_i hubs that recycle the weight of node i. Thus, high in-degree can increase the weight of a node, even if it is far from a seed node.

The uniform initial configuration case: In the case of the uniform initial configuration we know that the seed nodes will converge immediately to weight 1. Therefore, the Max algorithm will rank the seed nodes first. The rest of the nodes receive less weight than the seed nodes. Their weight is determined from Theorem 4.2, and depends upon their relation with the seed nodes of the graph, the connectivity with the rest of the nodes and their own in-degree.

The arbitrary initial configuration case: If we knew the stationary weights of the seed nodes then we would be able to compute the weights of the rest of the nodes recursively, using the formula in Theorem 4.2. However, the weights of the seeds depend on the initial configuration. Lemma 4.2 guarantees that at least one seed will receive weight 1. We obtain the following corollary of Lemma 4.2 for the special case of graphs that contain a single seed node (a case we encounter often in our experiments).

Corollary 4.4 *For a graph G that contains a single seed node, the algorithm* Max *converges for any initial fair configuration to the same stationary configuration as when initialized to the uniform configuration.*

One would hope that all seeds converge to weight 1, for all initial configurations, in which case we would fall back to the uniform case. However, this is not the case. One can construct simple examples of graphs that consist of multiple disconnected components, where, depending on the weight assigned to each component, the algorithm converges to different configurations. Consider for example the graph in Figure 4.5. The graph G consists of two components. and has just 3 authorities. The first component consists of just authority v which is pointed to by 3 hubs. The second component contains two authorities. Authority u is pointed to by 3 hubs, and one of these hubs points also to authority w. There are two seeds in the graph, authorities v and u. Assume that we initialize the algorithm with weights $a_v^0 = 1$, $a_u^0 = x$, and $a_w^0 = 1$, where $0 \leq x \leq 1$. Then, the algorithm will converge to the weights $a_v = 1$, $a_u = (1+2x)/3$, and $a_w = (1+2x)/9$. For different values of x we obtain a different stationary weight configuration. This is something to be expected. From Corollary 4.2 it is clear that the only mass that “survives” is the mass of the seed nodes. Therefore, by varying the weight of the seed nodes we vary the stationary weight vector.

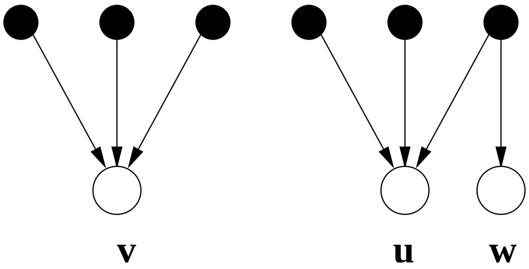


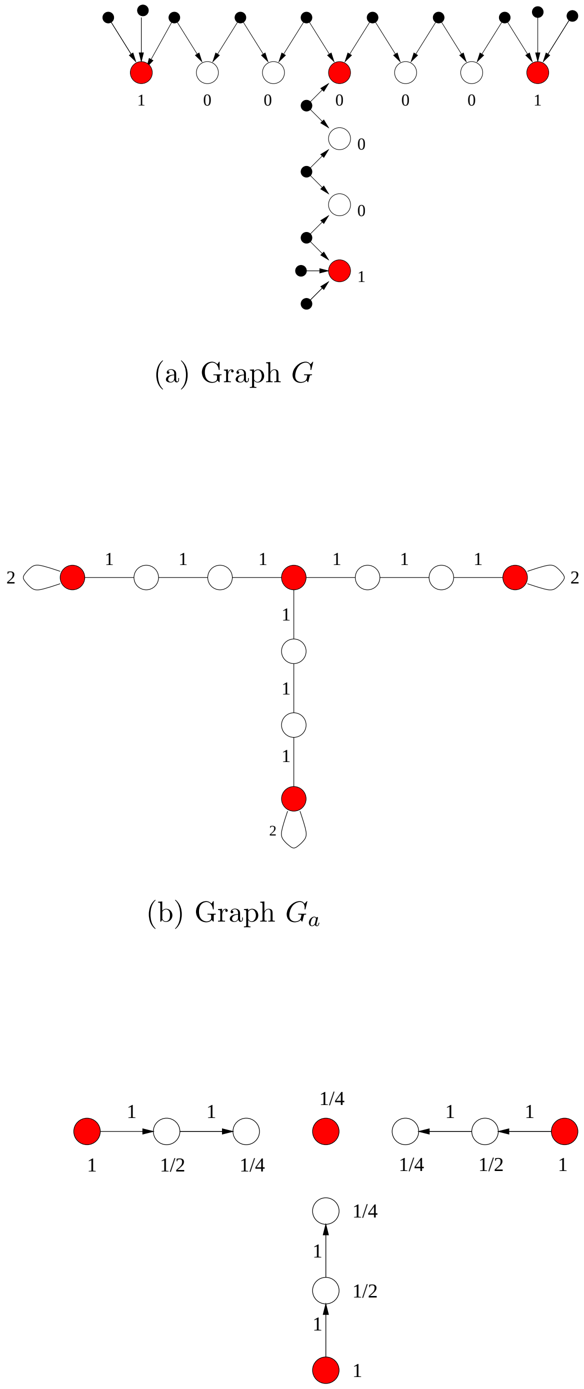
Figure 4.5: An example with a non-uniform initial configuration

A natural question is whether we can prove a similar result if we consider an authority connected graph, that is, a graph G, such that the authority graph G_a is connected. We now present a counter example, where we show that for an authority connected graph G, there exists an initial configuration such that one of the seed nodes converges to a weight less than 1. Furthermore, there exist non-seed nodes that have weight greater than the weight of that seed node.

Proposition 4.2 *The* MAX *algorithm does not always converge to the same weight vector for all initial configurations, even when restricted on authority connected graphs.*

Proof: Consider the graph G in Figure 4.6(a). The large red and white nodes are the authorities, while the small black nodes are the hubs. The shaded (red) nodes are the seed nodes of the graph G. There are four seeds in the graph, each with in-degree 3. Figure 4.6(b) shows the corresponding graph G_a. The initial configuration assigns weight 1 to all seed nodes, except for the central seed, which receives zero weight. The non-seed nodes are also initialized to zero weight. The initial weights for each node are shown next to vertices of the graph in Figure 4.6(a).

When the algorithm converges, we obtain graph G_A shown in Figure 4.6(c). The number next to each node in the graph is the stationary weight of the node. The weights on the edges are equal to the $c(i,j)$ values. Obviously, the algorithm does not converge to the same stationary configuration as when initialized to the uniform configuration, since the central seed node receives weight less than 1. Also, in this example there exist non-seed nodes that receive weight greater than the weight of the central seed node. □



(a) Graph G

(b) Graph G_a

(c) Graph G_A

Figure 4.6: An example with a non-uniform initial configuration for authority connected graphs

Chapter 5

A Theoretical Framework for the Analysis of LAR Algorithms

5.1 Motivation

The seminal work of Kleinberg [57] and Brin and Page [13] was followed by an avalanche of Link Analysis Ranking algorithms [10, 8, 64, 78, 4, 1, 73]. Faced with this wide range of choices for LAR algorithms, researchers usually resort to experiments to evaluate them and determine which one is more appropriate for the problem at hand. However, experiments are only indicative of the behavior of the LAR algorithm. In many cases, experimental studies are inconclusive. Furthermore, there are often cases where algorithms exhibit similar properties and ranking behavior. For example, in their experimental study, Borodin et al. [10] observed a strong "similarity" between two seemingly unrelated algorithms.

It seems that experimental evaluation of the performance of an LAR algorithm is not sufficient to fully understand its ranking behavior. We need a precise way to evaluate the properties of the LAR algorithms. We would like to be able to formally answer questions of the following type. "How similar are two LAR algorithms?". "On what kind of graphs do two LAR algorithms return similar rankings?". "How does the ranking behavior of an LAR algorithm depend on the specific class of graphs?". "How does the ranking of an LAR algorithm change as the underlying graph is modified?". "Is there a set of properties that characterize an LAR algorithm?".

In this chapter we describe a formal study of LAR algorithms. We introduce a theoretical framework that allows us to define properties of the LAR algorithms, and compare their ranking behavior. We conclude with an axiomatic characterization of the InDegree algorithm.

5.2 Link Analysis Ranking algorithms

We first need to formally define a Link Analysis Ranking algorithm. Let $\mathcal{G}_n$ denote the set of all possible graphs of size n. The size of the graph is the number of nodes in the graph. Let $\overline{\mathcal{G}}_n \subseteq \mathcal{G}_n$ denote a collection of graphs in $\mathcal{G}_n$. We define a link analysis algorithm $\mathcal{A}$ as a function $\mathcal{A} : \overline{\mathcal{G}}_n \to \mathbb{R}^n$, that maps a graph $G \in \overline{\mathcal{G}}_n$ to an n-dimensional real vector. The vector $\mathcal{A}(G)$ is the authority weight vector (or weight vector) produced by the algorithm $\mathcal{A}$ on graph G. The value of the entry $\mathcal{A}(G)[i]$ of vector $\mathcal{A}(G)$ denotes the authority weight assigned by the algorithm $\mathcal{A}$ to the node i. We will use $\boldsymbol{a}$ (or often $\boldsymbol{w}$) to denote the authority weight vector of algorithm $\mathcal{A}$. In this chapter we will sometimes use $a(i)$ instead of a_i to denote the authority weight of node i. All algorithms that we consider are defined over $\mathcal{G}_n$, the class of all possible graphs. We will also consider another class of graphs, $\mathcal{G}_n^{AC}$, the class of *authority connected* graphs. Recall that a graph G is authority connected, if the corresponding authority graph G_a consists of a single component.

We will assume that the weight vector $\mathcal{A}(G)$ is normalized under some chosen norm. The choice of normalization affects the output of the algorithm, so we distinguish between algorithms that use different norms. For any norm L, we define an L-algorithm $\mathcal{A}$ to be an algorithm, where the weight vector of $\mathcal{A}$ is normalized under L. That is, the algorithm maps the graphs in $\mathcal{G}_n$ onto the unit L-sphere. For the following, when not stated explicitly, we will assume that the weight vectors of the algorithms are normalized under the L_p norm for some $1 \leq p \leq \infty$.

5.3 Monotonicity

The first property of LAR algorithms that we define is *monotonicity*. Monotonicity requires that, if all hubs that point to node j also point to node k, then node k should receive authority weight at least as high as that of node j. Formally, we define monotonicity as follows.

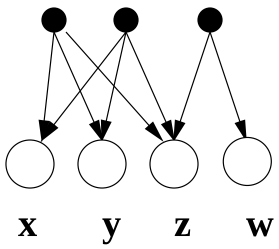


Figure 5.1: The non-monotonicity of AuthorityAvg

Definition 5.1 *An LAR algorithm $\mathcal{A}$ is* monotone *on the class of graphs $\overline{\mathcal{G}}_n$ if it has the following property. For every graph $G \in \overline{\mathcal{G}}_n$, and for every pair of nodes j and k in the graph G, if $B(j) \subseteq B(k)$, then $\mathcal{A}(G)[j] \leq \mathcal{A}(G)[k]$.*

Monotonicity appears to be a "reasonable" property but one can define "reasonable" algorithms that are not monotone. For example, consider the AuthorityAvg algorithm, the authority analogue of the HubAvg algorithm, where the authority weight of a node is defined to be the average of the hub weights of the nodes that point to this node. Consider now the graph in Figure 5.1. In this graph we have that $B(x) \subset B(z)$ and $B(y) \subset B(z)$, but AuthorityAvg assigns higher weight to nodes x and y than to z. An idea similar to that of the AuthorityAvg algorithm is suggested by Bharat and Henzinger [8]. When computing the authority weight of node i they average the weights of hubs that belong to the same domain. Another example of a non-monotone algorithm is the HubTheshold algorithm defined by Borodin et al. [10].

Theorem 5.1 *The algorithms* InDegree*,* Hits*,* PageRank*,* Salsa*,* HubAvg*,* AT*(k),* Norm*(p), and* BFS *are all monotone.*

Proof: Let j and k be two nodes in a graph G, such that $B(j) \subseteq B(k)$. For the InDegree algorithm monotonicity is obvious, since the authority weights are proportional to the in-degrees of the nodes, and the in-degree of j is less than, or equal to the in-degree of k. The same holds for the Salsa algorithm within each authority connected component, which is sufficient to prove the monotonicity of the algorithm.

For the PageRank algorithm, if a_j and a_k are the weights of nodes j and k, then we have that

$$a_j = \sum_{i=1}^{n} M_{PR}[i,j]a_i \qquad \text{and} \qquad a_k = \sum_{i=1}^{n} M_{PR}[i,k]a_i$$

where M_{PR} is the matrix for the PAGERANK algorithm, defined in Section 2.2.2. Then, for all i, $M_{PR}[i,j] \leq M_{PR}[i,k]$. Therefore, $a_j \leq a_k$.

For the HITS, HUBAVG, AT(k), and NORM(p) algorithms, it suffices to observe that, at every iteration t,

$$\overline{a}_j^t = \sum_{i \in B(j)} h_i \leq \sum_{i \in B(k)} h_i = \overline{a}_k^t$$

where $\overline{a}_j^t$ and $\overline{a}_k^t$ are the weights of nodes j and k at iteration t before applying the normalization step. Normalization may result in both weights converging to zero, as $t \to \infty$, but it cannot be the case that in the limit $a_j > a_k$.

For the BFS algorithm, it suffices to observe that, since $B(j) \subseteq B(k)$, every node that is reachable from node j is also reachable from node k. Thus $a_j \leq a_k$. □

We also define the following stronger notion of monotonicity.

Definition 5.2 *An LAR algorithm $\mathcal{A}$ is* strictly monotone *on the class of graphs $\overline{\mathcal{G}}_n$ if it has the following property. For every graph $G \in \overline{\mathcal{G}}_n$, and for every pair of nodes j and k in the graph G, $B(j) \subset B(k)$ if and only if $\mathcal{A}(G)[j] < \mathcal{A}(G)[k]$.*

We can now prove the following theorem.

Theorem 5.2 *The algorithms* INDEGREE, PAGERANK, SALSA, *and* BFS *are strictly monotone on the class $\mathcal{G}_n$, while the algorithms* HITS, HUBAVG, *and* MAX *are not strictly monotone on the class $\mathcal{G}_n$. The algorithms* INDEGREE, HITS, PAGERANK, SALSA, HUBAVG, AT*(k),* NORM*(p), and* BFS *are all strictly monotone on the class of authority connected graphs $\mathcal{G}_n^{AC}$.*

Proof: The strict monotonicity on the class of the authority connected graphs $\mathcal{G}_n^{AC}$ follows directly from the proof of Theorem 5.1 for the monotonicity of the algorithms. Similarly, for the strict monotonicity of INDEGREE, PAGERANK, SALSA, and BFS on the class $\mathcal{G}_n$.

For the HITS and HUBAVG algorithms, consider a graph $G \in \mathcal{G}_n$ consisting of two disconnected components. If the components are chosen appropriately the HITS and HUBAVG algorithms will allocate all the weight to one of the components, and zero weight to the nodes of the other component. Furthermore, if one of the two components does not contain a seed node, then the MAX algorithm will allocate zero weight to the nodes of that component. If chosen appropriately, the nodes in the component that receive zero weight violate the strict monotonicity property. □

A different notion of monotonicity is considered by Chien et al. [18]. In their paper, they study how the weight of a node changes as new links are added to the node. In this setting an algorithm is monotone if the weight of a node increases as its in-degree increases.

5.4 Distance measures between LAR algorithms

We are interested in comparing different LAR algorithms, as well as studying the ranking behavior of a specific LAR algorithm as we modify the underlying graph. To this end we need to define a distance measure between the rankings produced by the algorithms. Recall that an LAR algorithm $\mathcal{A}$ is a function that maps a graph G from a class of graphs $\overline{\mathcal{G}}_n$ to an n-dimensional vector $\mathcal{A}(G)$. Let $\mathcal{A}_1$ and $\mathcal{A}_2$ be two LAR algorithms defined on the class $\overline{\mathcal{G}}_n$. We define the *distance* between the algorithms $\mathcal{A}_1$ and A_2 on graph $G \in \overline{\mathcal{G}}_n$ as $d\left(\mathcal{A}_1(G), \mathcal{A}_2(G)\right)$, where $d : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ is some function that maps two real n-dimensional weight vectors $\boldsymbol{a}_1$, $\boldsymbol{a}_2$ to a real number $d(\boldsymbol{a}_1, \boldsymbol{a}_2)$.

5.4.1 Geometric distance measures

The first distance functions we consider capture the closeness of the actual weights assigned to every node. The authority weight vectors can be viewed as points in an n-dimensional space, thus we can use common geometric measures of distance. We consider the Manhattan distance, that is, the L_1 distance of the two vectors. Let $\mathcal{A}_1$ and $\mathcal{A}_2$ be two LAR algorithms defined on the class $\overline{\mathcal{G}}_n$, and let $\boldsymbol{a}_1$, $\boldsymbol{a}_2$ be the weight vectors of the algorithms on some graph $G \in \overline{\mathcal{G}}_n$. We define the d_1 distance measure between $\mathcal{A}_1$ and $\mathcal{A}_2$ on G as follows

$$d_1(\boldsymbol{a}_1, \boldsymbol{a}_2) = \min_{\gamma_1, \gamma_2 \geq 1} \sum_{i=1}^{n} \left|\gamma_1 a_1(i) - \gamma_2 a_2(i)\right| \ .$$

The constants γ_1 and γ_2 are meant to allow for an arbitrary scaling of the two vectors, thus eliminating large distances that are caused solely due to normalization factors. For example, let $\overline{\boldsymbol{w}} = (1, 1, ..., 1, 2)$, and $\overline{\boldsymbol{v}} = (1, 1, ..., 1)$ be the output of two algorithms $\mathcal{A}_1$ and $\mathcal{A}_2$, before any normalization is applied. These two vectors appear to be close. Suppose that we normalize the output of the algorithm in the L_∞ norm, and let $\boldsymbol{w}_\infty$ and $\boldsymbol{v}_\infty$ denote the normalized vectors. Then $\sum_{i=1}^{n} |w_\infty(i) - v_\infty(i)| = (n-1)/2 = \Theta(n)$. As we will show in Section A.2 (Appendix A), the maximum L_1 distance between any two L_∞-unit vectors is $\Theta(n)$, therefore, these two vectors appear to be far apart. Suppose now that we normalize in the L_1 norm, and let $\boldsymbol{w}_1$ and $\boldsymbol{v}_1$ denote the normalized vectors. Then $\sum_{i=1}^{n} |w_1(i) - v_1(i)| = \frac{2(n-1)}{n(n+1)} = \Theta(1/n)$. The maximum L_1 distance between any two L_1-unit vectors is $\Theta(1)$, therefore, the two vectors now appear to be close. We use the constants γ_1, γ_2 to avoid such discrepancies.

Instead of the L_1 distance, we may use other geometric distance measures, such as the Euclidean distance L_2. In general we define the d_q distance, as the L_q distance of the weight vectors. Formally,

$$d_q(\boldsymbol{a}_1, \boldsymbol{a}_2) = \min_{\gamma_1,\gamma_2 \geq 1} \sum_{i=1}^{n} \|\gamma_1 a_1(i) - \gamma_2 a_2(i)\|_q$$

For the remainder of the chapter we only consider the d_1 distance measure.

5.4.2 Rank distance measures

The next distance functions we consider capture the similarity between the *ordinal* rankings produced by the two algorithms. The motivation behind this definition is that the ordinal ranking is the usual end-product seen by the user. Let $\boldsymbol{a}$ be the n-dimensional authority weight vector of an algorithm $\mathcal{A}$ over some graph $G = (P, E)$ in $\mathcal{G}_n$. The vector $\boldsymbol{a}$ induces a *ranking* of the nodes in P, such that a node i is ranked above node j if $a_i > a_j$. If all weights are distinct, the authority weights induce a *total ranking* of the elements in P. If the weights are not all distinct, then we have a *partial ranking* of the elements in P. We will also refer to total rankings as *permutations*.

We are interested in measuring the similarity between the rankings induced by the two algorithms. We will first review some of the metrics for comparing permutations, and we will see how these can be extended to the case of partial rankings. For the remainder of this section, we borrow heavily from the work of Fagin et al. [34] on comparing top-k lists.

Distance measures between permutations

The problem of comparing permutations has been studied extensively [56, 26, 30]. In this setting, we compare two *complete rankings* of a set of elements. Let P be a set of elements that we wish to order (in our case the nodes in the graph). A permutation σ is defined as a bijection from the set P to the set $[n] = \{1, 2, \ldots, n\}$, where n is the size of P. The value $\sigma(i)$ is interpreted as the position (rank) of the element $i \in P$ in the ranking. We say that element i is ranked *ahead* of element j if $\sigma(i) < \sigma(j)$. We also use $\mathcal{P}$ to denote the set of all distinct unordered pairs of nodes in P, and S_P to denote the set of all possible permutations of the elements of P.

The *Kendall's tau* distance measure between permutations is defined as follows. Given two permutations σ_1 and σ_2, we define the indicator function $\mathcal{I}_{\sigma_1\sigma_2}(i,j)$, such that $\mathcal{I}_{\sigma_1\sigma_2}(i,j) = 0$, if i and j are ranked in the same order in both σ_1 and σ_2, and $\mathcal{I}_{\sigma_1\sigma_2}(i,j) = 1$ otherwise. Kendall's tau is defined as follows.

$$K(\sigma_1, \sigma_2) = \sum_{\{i,j\}\in\mathcal{P}} \mathcal{I}_{\sigma_1\sigma_2}(i,j)$$

Kendall's tau is equal to the number of bubble sort swaps that are necessary to convert one permutation to the other. The maximum value of Kendall's tau is $n(n-1)/2$, and it occurs when the permutation σ_1 is the reverse of the permutation σ_2. In the following we normalize the Kendall's tau distance with $n(n-1)/2$, so that it takes values in $[0,1]$.

Another metric for comparing permutations is *Spearman's Footrule* metric [26], which is the L_1 distance between the two permutations. Formally, for two permutations $\sigma_1, \sigma_2 \in S_P$, $F(\sigma_1, \sigma_2) = \sum_{i=1}^{n} |\sigma_1(i) - \sigma_2(i)|$.

Measures for comparing partial rankings

In an ideal world a ranking algorithm would produce a distinct authority weight for each node in the set P. Then we would be able to compare rankings by applying directly the distance measures on permutations. However, there are cases where the ranking algorithms may assign equal weights to two different nodes. Thus, an authority weight vector usually produces a partial ranking of the nodes. In this section we will show how we can extend the measures we discussed in this setting. We later discovered that our results have been proven independently by Fagin et al [32].

Let $\boldsymbol{a}$ be an authority vector produced by algorithm $\mathcal{A}$. We say that permutation σ is *consistent* with the vector $\boldsymbol{a}$ if the following holds. For every pair of nodes $\{i,j\}$ if $a_i < a_j$ then $\sigma(i) < \sigma(j)$. That is, if node i receives weight greater than the weight of node j, then i must be ranked above j. Let $\Sigma_{\boldsymbol{a}} \subseteq S_P$ denote the set of permutations that are consistent with weight vector $\boldsymbol{a}$.

There are several ways of generalizing the Kendall's tau distance for the case of partial rankings. Let $\boldsymbol{a}_1$ and $\boldsymbol{a}_2$ be the weight vectors and let Σ_1 and Σ_2 be the sets of permutations that are consistent with vectors $\boldsymbol{a}_1$ and $\boldsymbol{a}_2$ respectively. Similar to the work of Fagin et al. [34], we define the following *rank distance* measures.

The *minimizing Kendall rank distance*, $K_{min}(\boldsymbol{a}_1, \boldsymbol{a}_2)$.

$$K_{min}(\boldsymbol{a}_1, \boldsymbol{a}_2) = \frac{1}{n(n-1)/2} \min_{\sigma_1 \in \Sigma_1, \sigma_2 \in \Sigma_2} K(\sigma_1, \sigma_2)$$

The *Hausdorff Kendall rank distance*, $K_{haus}(\boldsymbol{a}_1, \boldsymbol{a}_2)$.

$$K_{haus}(\boldsymbol{a}_1, \boldsymbol{a}_2) = \frac{1}{n(n-1)/2} \max \left\{ \max_{\sigma_1 \in \Sigma_1} \min_{\sigma_2 \in \Sigma_2} K(\sigma_1, \sigma_2), \max_{\sigma_2 \in \Sigma_2} \min_{\sigma_1 \in \Sigma_1} K(\sigma_1, \sigma_2) \right\}$$

This is an application of the well known Hausdorff distance metric between sets for the sets Σ_1, Σ_2, where the distance between the individual pairs of elements from Σ_1, Σ_2, is taken to be Kendall's tau.

The *Kendall rank distance with penalty p*, $K^{(p)}(\boldsymbol{a}_1, \boldsymbol{a}_2)$.

$$K^{(p)}(\boldsymbol{a}_1, \boldsymbol{a}_2) = \frac{1}{n(n-1)/2} \sum_{\{i,j\} \in \mathcal{P}} \mathcal{I}^{(p)}_{\boldsymbol{a}_1 \boldsymbol{a}_2}(i,j)$$

where $\mathcal{I}^{(p)}_{\boldsymbol{a}_1 \boldsymbol{a}_2}(i,j)$ is a penalty function defined over the set $\mathcal{P}$. For the definition of the function $\mathcal{I}^{(p)}_{\boldsymbol{a}_1 \boldsymbol{a}_2}(i,j)$, we break up $\mathcal{P}$ into the following subsets.

- The set $\mathcal{E} = \mathcal{E}(\boldsymbol{a}_1, \boldsymbol{a}_2) \subseteq \mathcal{P}$ contains all pairs $\{i,j\} \in \mathcal{P}$ such that $a_1(i) = a_1(j)$ and $a_2(i) = a_2(j)$. For all $\{i,j\} \in \mathcal{E}$, $\mathcal{I}^{(p)}_{\boldsymbol{a}_1 \boldsymbol{a}_2}(i,j) = 0$.

- The set $\mathcal{U} = \mathcal{U}(\boldsymbol{a}_1, \boldsymbol{a}_2) \subseteq \mathcal{P}$ contains all pairs $\{i,j\} \in \mathcal{P}$ such that $a_1(i) < a_1(j)$ and $a_2(i) < a_2(j)$, or $a_1(i) > a_1(j)$ and $a_2(i) > a_2(j)$. For all $\{i,j\} \in \mathcal{U}$, $\mathcal{I}^{(p)}_{\boldsymbol{a}_1 \boldsymbol{a}_2}(i,j) = 0$.

- The set $\mathcal{X} = \mathcal{X}(\boldsymbol{a}_1, \boldsymbol{a}_2) \subseteq \mathcal{P}$ contains all pairs $\{i,j\} \in \mathcal{P}$ such that $a_1(i) < a_1(j)$ and $a_2(i) > a_2(j)$, or $a_1(i) > a_2(j)$ and $a_1(i) < a_2(j)$. For all $\{i,j\} \in \mathcal{X}$, $\mathcal{I}^{(p)}_{\boldsymbol{a}_1 \boldsymbol{a}_2}(i,j) = 1$.

- The set $\mathcal{Y} = \mathcal{Y}(\boldsymbol{a}_1, \boldsymbol{a}_2) \subseteq \mathcal{P}$ contains all pairs $\{i,j\} \in \mathcal{P}$ such that satisfy $a_1(i) = a_1(j)$ and $a_2(i) \neq a_2(j)$. For all $\{i,j\} \in \mathcal{Y}$, $\mathcal{I}^{(p)}_{\boldsymbol{a}_1 \boldsymbol{a}_2}(i,j) = p$.

- The set $\mathcal{Z} = \mathcal{Z}(\boldsymbol{a}_1, \boldsymbol{a}_2) \subseteq \mathcal{P}$ contains all pairs $\{i,j\} \in \mathcal{P}$ that satisfy $a_2(i) = a_2(j)$ and $a_1(i) \neq a_1(j)$. For all $\{i,j\} \in \mathcal{Z}$, $\mathcal{I}^{(p)}_{\boldsymbol{a}_1 \boldsymbol{a}_2}(i,j) = p$.

The parameter p takes values in $[0, 1]$. The value $p = 0$ gives a lenient approach, where we penalize the algorithm only for pairs that are weighted so that they *force* an inconsistent ranking. The value $p = 1$ gives a strict approach, where we penalize the algorithm for all pairs that are weighted so that they *allow* for an inconsistent ranking. Values of p in $(0, 1)$ give a combined approach. Clearly,

$$K^{(p)}(\boldsymbol{a}_1, \boldsymbol{a}_2) = (1 - p)K^{(0)}(\boldsymbol{a}_1, \boldsymbol{a}_2) + pK^{(1)}(\boldsymbol{a}_1, \boldsymbol{a}_2) \ . \tag{5.1}$$

Also, from the definition of $K^{(p)}$, it is obvious that $K^{(p)}(\boldsymbol{a}_1, \boldsymbol{a}_2) = |\mathcal{X}(\boldsymbol{a}_1, \boldsymbol{a}_2)| + p(|\mathcal{Y}(\boldsymbol{a}_1, \boldsymbol{a}_2)| + |\mathcal{Z}(\boldsymbol{a}_1, \boldsymbol{a}_2)|)$. For the remainder, when it is understood, we will omit the indexes $(\boldsymbol{a}_1, \boldsymbol{a}_2)$ when referring to the subsets of $\mathcal{P}$. For example we will use $\mathcal{X}$ to denote $\mathcal{X}(\boldsymbol{a}_1, \boldsymbol{a}_2)$.

We prove the following theorem for the different Kendall rank distances.

Theorem 5.3 *Given two authority vectors $\boldsymbol{a}_1$ and $\boldsymbol{a}_2$ we have the following.*

1. $K_{min}(\boldsymbol{a}_1, \boldsymbol{a}_2) = K^{(0)}(\boldsymbol{a}_1, \boldsymbol{a}_2) = |\mathcal{X}(\boldsymbol{a}_1, \boldsymbol{a}_2)|$.

2. $K_{haus}(\boldsymbol{a}_1, \boldsymbol{a}_2) = |\mathcal{X}(\boldsymbol{a}_1, \boldsymbol{a}_2)| + \max\{|\mathcal{Y}(\boldsymbol{a}_1, \boldsymbol{a}_2)|, |\mathcal{Z}(\boldsymbol{a}_1, \boldsymbol{a}_2)|\}$.

3. *For any* $0 < p < p' \leq 1$ *we have that* $K^{(p)}(\boldsymbol{a}_1, \boldsymbol{a}_2) \leq K^{(p')}(\boldsymbol{a}_1, \boldsymbol{a}_2) \leq \frac{p'}{p} K^{(p)}(\boldsymbol{a}_1, \boldsymbol{a}_2)$.

Proof: Let Σ_1 and Σ_2 denote the set of all permutations consistent with $\boldsymbol{a}_1$ and $\boldsymbol{a}_2$ respectively. First, obviously, for all $\sigma_1 \in \Sigma_1$, and $\sigma_2 \in \Sigma_2$, $K(\sigma_1, \sigma_2) \geq |\mathcal{X}|$, since for all pairs $\{i, j\} \in \mathcal{X}$, σ_1 and σ_2 rank i and j differently. Also, for all pairs $\{i, j\} \in |\mathcal{U}|$, $\mathcal{I}_{\sigma_1 \sigma_2}(i, j) = 0$. Thus, these pairs do not contribute to the rank distance.

For the proof of part (1) of the theorem, consider a permutation $\sigma_1^* \in \Sigma_1$ such that σ_1^* ranks the pairs in $\mathcal{Z}$ in an order that is consistent with the vector $\boldsymbol{a}_2$. Then consider a permutation $\sigma_2^* \in \Sigma_2$, such that σ_2^* ranks the pairs in $\mathcal{Y}$ in an order that is consistent with the vector $\boldsymbol{a}_1$, and ranks the elements in $\mathcal{E}$ in the same order as σ_1^*. Then $K(\sigma_1^*, \sigma_2^*) = |\mathcal{X}| = \min_{\sigma_1 \in \Sigma_1, \sigma_2 \in \Sigma_2} K(\sigma_1, \sigma_2)$.

For the proof of part (2), we first observe that the pairs in $\mathcal{X}$ contribute $|\mathcal{X}|$ to K_{haus}. Consider now $\max_{\sigma_1 \in \Sigma_1} \min_{\sigma_2 \in \Sigma_2} K(\sigma_1, \sigma_2)$. Suppose that σ_1 is fixed to a permutation σ_1^*. Then, the permutation σ_2^* that minimizes $K(\sigma_1^*, \sigma_2^*)$ is the one that ranks the pairs in $\mathcal{E}$ and $\mathcal{Y}$ in the same order as σ_1^*. $K(\sigma_1^*, \sigma_2^*)$ is obviously maximized when σ_1^* ranks the pairs in $\mathcal{Z}$ in the reverse order of that defined by the vector $\boldsymbol{a}_2$. Thus, $\max_{\sigma_1 \in \Sigma_1} \min_{\sigma_2 \in \Sigma_2} K(\sigma_1, \sigma_2) =$

$|\mathcal{X}|+|\mathcal{Z}|$. Symmetrically, $\max_{\sigma_2\in\Sigma_2}\min_{\sigma_1\in\Sigma_1} K(\sigma_1,\sigma_2) = |\mathcal{X}|+|\mathcal{Y}|$. Thus, $K_{haus}(\boldsymbol{a}_1,\boldsymbol{a}_2) = |X| + \max\{|\mathcal{X}|, |\mathcal{Y}|\}$.

For the proof of part (3), $K^{(p)}(\boldsymbol{a}_1,\boldsymbol{a}_2) \le K^{(p')}(\boldsymbol{a}_1,\boldsymbol{a}_2)$ follows directly from the fact that $p \le p'$ and the definition of $K^{(p)}$. Furthermore,

$$\frac{K^{(p')}(\boldsymbol{a}_1,\boldsymbol{a}_2)}{K^{(p)}(\boldsymbol{a}_1,\boldsymbol{a}_2)} \le \frac{K^{(p')}(\boldsymbol{a}_1,\boldsymbol{a}_2) - K^{(0)}(\boldsymbol{a}_1,\boldsymbol{a}_2)}{K^{(p)}(\boldsymbol{a}_1,\boldsymbol{a}_2) - K^{(0)}(\boldsymbol{a}_1,\boldsymbol{a}_2)} = \frac{p'}{p}$$

The inequality follows from the fact that for any $0 < a < y$, $x/y \le (x-a)/(y-a)$ if and only if $x \le y$. The equality follows from Equation 5.1. □

From Theorem 5.3 if follows that $K^{(0)}(\boldsymbol{a}_1,\boldsymbol{a}_2) \le K_{haus}(\boldsymbol{a}_1,\boldsymbol{a}_2) \le K^{(1)}(\boldsymbol{a}_1,\boldsymbol{a}_2)$. Note that, similar to K_{min}, we could also define $K_{max}(\boldsymbol{a}_1,\boldsymbol{a}_2) = \frac{1}{n(n-1)/2}\max_{\sigma_1\in\Sigma_1,\sigma_2\in\Sigma_2} K(\sigma_1,\sigma_2)$, and $K_{avg}(\boldsymbol{a}_1,\boldsymbol{a}_2) = \frac{1}{n(n-1)/2}E(K(\sigma_1,\sigma_2))$, where the expectation is taken over all $\sigma_1 \in \Sigma_1$ and $\sigma_2 \in \Sigma_2$. However, for both distance measures, the pairs in the set $\mathcal{E}$ contribute to the K_{max} and K_{avg} distances. It seems counter-intuitive to penalize the algorithms for pairs of nodes to which they assign equal weights, therefore, we do not consider these distance measures. For the remainder of the thesis, we focus on $K^{(0)}$ and $K^{(1)}$ rank distance measures.

Not all of the distance measures that we defined are metrics. We examine the properties of the measures in Appendix A, where we prove that some of the distance measures are metrics and that some are near metrics. The metric property is useful when examining the similarity and stability properties we discuss below.

5.4.3 Summary of distance measures used in the remainder of the thesis

Let $G \in \mathcal{G}_n$ be a graph and let $\boldsymbol{a}_1 = \mathcal{A}_1(G)$, and $\boldsymbol{a}_2 = \mathcal{A}_2(G)$ be the output of algorithms of $\mathcal{A}_1$ and $\mathcal{A}_2$ on G. For the remainder of the chapter, for comparing the weights of the algorithms, we will use the the d_1 distance measure.

$$d_1(\boldsymbol{a}_1,\boldsymbol{a}_2) = \min_{\gamma_1,\gamma_2\ge 1} \|\gamma_1\boldsymbol{a}_1 - \gamma_2\boldsymbol{a}_2\|_1$$

For comparing the rankings produced by the algorithms we will use $d_r^{(0)} = K^{(0)}$, which we will refer to as *weak rank distance*, and $d_r^{(1)} = K^{(1)}$, which we will refer to as *strict rank distance.* The two functions can be defined using the indicator functions $\mathcal{I}^{(0)}_{\boldsymbol{a}_1\boldsymbol{a}_2}(i,j)$ and

$\mathcal{I}^{(1)}_{\boldsymbol{a}_1\boldsymbol{a}_2}(i,j)$. The $\mathcal{I}^{(0)}_{\boldsymbol{a}_1\boldsymbol{a}_2}(i,j)$ function is defined as follows.

$$\mathcal{I}^{(0)}_{\boldsymbol{a}_1\boldsymbol{a}_2}(i,j) = \begin{cases} 1 & \text{if } \ (a_1(i) < a_1(j) \wedge a_2(i) > a_2(j)) \vee (a_1(i) > a_1(j) \wedge a_2(i) < a_2(j)) \\ 0 & \text{otherwise} \end{cases}$$

The $\mathcal{I}^{(1)}_{\boldsymbol{a}_1\boldsymbol{a}_2}(i,j)$ is defined as follows.

$$\mathcal{I}^{(1)}_{\boldsymbol{a}_1\boldsymbol{a}_2}(i,j) = \begin{cases} 0 & \text{if } a_1(i) < a_1(j) \Leftrightarrow a_2(i) > a_2(j) \\ 1 & \text{otherwise} \end{cases}$$

We define the weak rank distance, $d_r^{(0)}$, as follows.

$$d_r^{(0)}(\boldsymbol{a}_1, \boldsymbol{a}_2) = \frac{1}{n(n-1)/2} \sum_{\{i,j\}\in\mathcal{P}} \mathcal{I}^{(0)}_{\boldsymbol{a}_1\boldsymbol{a}_2}(i,j)$$

We define the strict rank distance, $d_r^{(1)}$, as follows.

$$d_r^{(1)}(\boldsymbol{a}_1, \boldsymbol{a}_2) = \frac{1}{n(n-1)/2} \sum_{\{i,j\}\in\mathcal{P}} \mathcal{I}^{(1)}_{\boldsymbol{a}_1\boldsymbol{a}_2}(i,j)$$

We will also use $d_r^{(p)}$ to denote the $K^{(p)}$ Kendall rank distance.

We note that there are other possible distance measures that can be defined between rankings. For example, we could view the weight vectors as probability distributions, and apply information theoretic measures for comparing them. Also, Fagin et al [32] generalize the Spearman's Footrule measure [26] for the case of partial rankings. These measures are beyond the scope of this thesis.

5.5 Similarity of LAR algorithms

We now turn to the problem of comparing two LAR algorithms. We first give the following generic definition of *similarity* of two LAR algorithms, for any distance function d, and any normalization norm $L = ||\cdot||$.

Definition 5.3 *Two L-algorithms $\mathcal{A}_1$ and $\mathcal{A}_2$ are similar on the class of graph $\overline{\mathcal{G}}_n$ under* *distance d, if as $n \to \infty$*

$$\max_{G \in \overline{\mathcal{G}}_n} d\left(\mathcal{A}_1(G), \mathcal{A}_2(G)\right) = o(M_n)$$

where $M_n = \sup_{\|\boldsymbol{w}_1\|=\|\boldsymbol{w}_2\|=1} d(\boldsymbol{w}_1, \boldsymbol{w}_2)$ is the maximum distance between any two n-dimensional vectors with unit norm $L = ||\cdot||$.

In the definition of similarity, instead of taking $\max_{G \in \overline{\mathcal{G}}_n}$ we may use some other operator. For example, if there exists some distribution over the graphs in $\overline{\mathcal{G}}_n$, we could replace max by the expectation of the distance between the algorithms. In this thesis, we only consider the max operator.

We now give the following definitions of similarity for the d_1, $d^{(0)}$ and $d_r^{(1)}$ distance measures. For the d_1 distance measure, in Section A.2 of Appendix A we show that the maximum d_1 distance between any two n-dimensional L_p unit vectors is $\Theta(n^{1-1/p})$.

Definition 5.4 *Let $1 \leq p \leq \infty$. Two L_p-algorithms $\mathcal{A}_1$ and $\mathcal{A}_2$ are* d_1-similar *(or,* similar*) on the class of graphs $\overline{\mathcal{G}}_n$, if as $n \to \infty$,*

$$\max_{G \in \overline{\mathcal{G}}_n} d_1\left(\mathcal{A}_1(G), \mathcal{A}_2(G)\right) \;=\; o\left(n^{1-1/p}\right)$$

Definition 5.5 *Two algorithms, $\mathcal{A}_1$ and $\mathcal{A}_2$, are* weakly rank similar *on the class of graphs $\overline{\mathcal{G}}_n$, if as $n \to \infty$,*

$$\max_{G \in \overline{\mathcal{G}}_n} d_r^{(0)}(\mathcal{A}_1(G), \mathcal{A}_2(G)) \;=\; o(1)$$

Definition 5.6 *Two algorithms, $\mathcal{A}_1$ and $\mathcal{A}_2$, are* strictly rank similar *on the class of graphs $\overline{\mathcal{G}}_n$, if as $n \to \infty$,*

$$\max_{G \in \overline{\mathcal{G}}_n} d_r^{(1)}(\mathcal{A}_1(G), \mathcal{A}_2(G)) \;=\; o(1)$$

Definition 5.7 *Two algorithms, $\mathcal{A}_1$ and $\mathcal{A}_2$, are* rank consistent *on the class of graphs $\overline{\mathcal{G}}_n$, if for every graph $G \in \overline{\mathcal{G}}_n$,*

$$d_r^{(0)}(\mathcal{A}_1(G), \mathcal{A}_2(G)) = 0$$

Definition 5.8 *Two algorithms, $\mathcal{A}_1$ and $\mathcal{A}_2$, are* rank equivalent *on the class of graphs $\overline{\mathcal{G}}_n$, if for every graph $G \in \overline{\mathcal{G}}_n$,*

$$d_r^{(1)}(\mathcal{A}_1(G), \mathcal{A}_2(G)) = 0$$

We note that, according to the above definition, every algorithm is rank consistent with the trivial algorithm that gives the same weight to all authorities. Although this may seem somewhat bizarre, it does have an intuitive justification. For an algorithm whose goal is to produce an *ordinal* ranking, the weight vector with all weights equal conveys no information; therefore, it lends itself to all possible ordinal rankings. The weak rank distance counts only the pairs that are weighted inconsistently, and in this case there are none. If a stronger notion of similarity is needed, in the $d_r^{(1)}$ distance, all such pairs contribute to the distance.

From the discussion in Section 5.4.2, it is obvious that if two algorithms are strictly rank similar, then they are similar under all other rank distances defined in Section 5.4.2. Equivalently, if two algorithms are not weakly rank similar, then they are not similar under any of the rank distance measures defined in Section 5.4.2.

The definition of similarity depends on the normalization of the algorithms. In the following, we show that, for the d_1 distance, similarity in the L_p norm implies similarity in the L_q norm, for any $q > p$.

Theorem 5.4 *Let $\mathcal{A}_1$ and $\mathcal{A}_2$ be two algorithms, and let $1 \leq p \leq q \leq \infty$. If the L_p-algorithm $\mathcal{A}_1$ and the L_p-algorithm $\mathcal{A}_2$ are similar, then the L_q-algorithm $\mathcal{A}_1$ and the L_q-algorithm $\mathcal{A}_2$ are also similar.*

Proof: Let G be a graph of size n, and let $\boldsymbol{u} = \mathcal{A}_1(G)$, and $\boldsymbol{v} = \mathcal{A}_2(G)$ be the weight vectors of the two algorithms. Let $\boldsymbol{v}_p$ and $\boldsymbol{u}_p$ denote the weight vectors, normalized in the L_p norm, and let $\boldsymbol{v}_q$ and $\boldsymbol{u}_q$ denote the weight vectors, normalized in the L_q norm. Since the L_p-algorithm $\mathcal{A}_1$ and the L_p-algorithm $\mathcal{A}_2$ are similar, there exist $\gamma_1, \gamma_2 \geq 1$ such that

$$d_1(\boldsymbol{v}_p, \boldsymbol{u}_p) = \sum_{i=1}^{n} |\gamma_1 v_p(i) - \gamma_2 u_p(i)| = o\left(n^{1-1/p}\right)$$

Now, $\boldsymbol{v}_q = \boldsymbol{v}_p / \|\boldsymbol{v}_p\|_q$, and $\boldsymbol{u}_q = \boldsymbol{u}_p / \|\boldsymbol{u}_p\|_q$. Therefore, $\sum_{i=1}^{n} |\gamma_1 \|\boldsymbol{v}_p\|_q v_q(i) - \gamma_2 \|\boldsymbol{u}_p\|_q u_q(i)| = o(n^{1-1/p})$. Without loss of generality assume that $\|\boldsymbol{u}_p\|_q \geq \|\boldsymbol{v}_p\|_q$. Then

$$\|\boldsymbol{v}_p\|_q \sum_{i=1}^{n} \left| \gamma_1 v_q(i) - \gamma_2 \frac{\|\boldsymbol{u}_p\|_q}{\|\boldsymbol{v}_p\|_q} u_q(i) \right| = o\left(n^{1-1/p}\right)$$

We set $\gamma_1' = \gamma_1$ and $\gamma_2' = \gamma_2 \frac{\|\boldsymbol{u}_p\|_q}{\|\boldsymbol{v}_p\|_q}$. Then we have that

$$d_1(\boldsymbol{v}_q, \boldsymbol{u}_q) \leq \sum_{i=1}^{n} |\gamma_1' v_q(i) - \gamma_2' u_q(i)| = o\left(\frac{n^{1-1/p}}{\|\boldsymbol{v}_p\|_q}\right) .$$

In Appendix A, Lemma A.3 we show that $\|\boldsymbol{v}_p\|_q \geq \|\boldsymbol{v}_p\|_p n^{1/q-1/p} = n^{1/q-1/p}$. Hence, $\frac{n^{1-1/p}}{\|\boldsymbol{v}_p\|_q} \leq \frac{n^{1-1/p}}{n^{1/q-1/p}} = n^{1-1/q}$. Therefore, $d_1(\boldsymbol{v}_q, \boldsymbol{u}_q) = o(n^{1-1/q})$, and thus L_q-algorithm $\mathcal{A}_1$, and L_q-algorithm $\mathcal{A}_2$ are similar. □

Theorem 5.4 implies that if two L_1-algorithms are similar, then the corresponding L_p-algorithms are also similar, for any $1 \leq p \leq \infty$. Consequently, if two L_∞-algorithms are dissimilar, then the corresponding L_p-algorithms are also dissimilar, for any $1 \leq p \leq \infty$. Therefore, all dissimilarity results proven for the L_∞ norm hold for any L_p norm, for $1 \leq p \leq \infty$.

5.5.1 Similarity Results

We now consider the similarity of the Hits, InDegree, Salsa, HubAvg and Max algorithms. We will show that no pair of algorithms are similar, or rank similar in the class $\mathcal{G}_n$ of all possible graphs of size n. For the dissimilarity results under the d_1 distance measure, we will assume that the weight vectors are normalized under the L_∞ norm. Dissimilarity between two L_∞-algorithms implies dissimilarity in L_p norm, for $p < \infty$.

The Hits and the InDegree algorithms

Proposition 5.1 *The* Hits *and the* InDegree *algorithms are neither similar, nor weakly rank similar on $\mathcal{G}_n$.*

Proof: Consider a graph G on $n = 7r - 2$ nodes that consists of two disconnected components. The first component C_1 consists of a complete bipartite graph with $2r - 1$ hubs and $2r - 1$ authorities. The second component C_2 consists of a bipartite graph with $2r$ hubs, and r authorities. The graph G for $r = 2$ is shown in Figure 5.2.

Let $\boldsymbol{a}$ and $\boldsymbol{w}$ denote the weight vectors of the Hits, and the InDegree algorithm respectively, on graph G. Then, the Hits algorithm allocates all the weight to the nodes in C_1. After normalization, for all $i \in C_1$, $a_i = 1$, while for all $j \in C_2$, $a_j = 0$. On the other hand, the InDegree algorithm distributes the weight to both components, allocating more

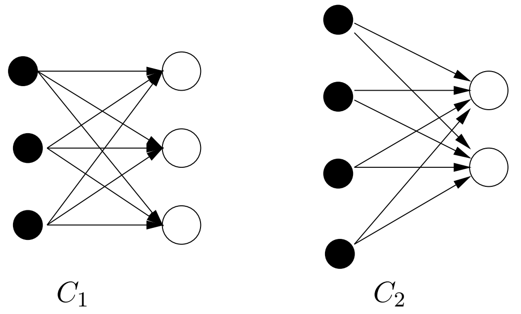


Figure 5.2: Dissimilarity of HITS and INDEGREE. The graph G for $r = 2$.

weight to the nodes in C_2. After the normalization step, for all $j \in C_2$, $w_j = 1$, while for all $i \in C_1$, $w_i = \frac{2r-1}{2r}$.

There are r nodes in C_2 for which $w_i = 1$ and $a_i = 0$. For all $\gamma_1, \gamma_2 \geq 1$, $\sum_{i \in C_2} |\gamma_1 w_i - \gamma_2 a_i| \geq r$. Therefore, $d_1(\boldsymbol{w}, \boldsymbol{a}) = \Omega(r) = \Omega(n)$, which proves that the algorithms are not similar.

The proof for weak rank dissimilarity follows immediately from the above. For every pair of nodes $\{i, j\}$ such that $i \in C_1$ and $j \in C_2$, $a_i > a_j$ and $w_i < w_j$. There are $\Theta(n^2)$ such pairs, therefore, $d_r^{(0)}(\boldsymbol{a}, \boldsymbol{w}) = \Theta(1)$. Thus, the two algorithms are not weakly rank similar. □

The SALSA algorithm

Proposition 5.2 *The* SALSA *algorithm is neither similar, nor weakly rank similar to the* INDEGREE*,* HUBAVG*, or* HITS *algorithms.*

Proof: Consider a graph G on $n = 6r$ nodes, that consists of two components C_1 and C_2. The component C_1 is a complete bipartite graph with $2r$ hubs and $2r$ authorities. The component C_2 is a complete bipartite graph with r hubs and r authorities, with one link (q, p) removed. Figure 5.3 shows the graph G for $r = 3$.

Let $\boldsymbol{u}$, $\boldsymbol{a}$, $\boldsymbol{v}$, and $\boldsymbol{w}$ denote the normalized weight vectors for SALSA, HITS, HUBAVG and INDEGREE algorithms respectively. Also, let $\boldsymbol{u}_1$ denote the SALSA weight vector normalized in the L_1 norm (i.e., as it is computed by the random walk of the SALSA algorithm). The SALSA algorithm allocates weight $u_1(i) = 1/3r$ for all authority nodes $i \in C_1$, and weight $u_1(j) = (r-1)/3(r^2-1)$ for all authority node $j \in C_2 \setminus \{p\}$. Hub nodes receive weight zero

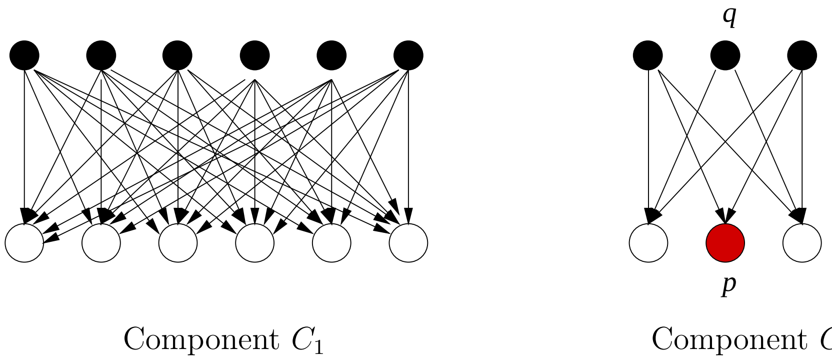


Figure 5.3: Dissimilarity of SALSA with HITS,INDEGREE and HUBAVG. The graph G for $r = 3$.

for all algorithms. It is interesting to note that the removal of the link (q, p) increases the weight of the rest of the nodes in C_2. Since $(r-1)/3(r^2-1) > 1/3r$, after normalization in the L_∞ norm, we have that $u_i = 1 - \frac{1}{r^2}$ for all $i \in C_1$, and $u_j = 1$ for all $j \in C_2 \setminus \{p\}$. On the other hand, both the HITS and HUBAVG algorithms distribute all the weight equally to the authorities in the C_1 component, and allocate zero weight to the nodes in the C_2 component. Therefore, after normalization, $a_i = v_i = 1$ for all nodes $i \in C_1$, and $a_j = v_j = 0$ for all nodes $j \in C_2$. The INDEGREE algorithm allocates weight proportionally to the in-degree of the nodes, therefore, after normalization, $w_i = 1$ for all nodes in C_1, while $w_j = \frac{1}{2}$ for all nodes $j \in C_2 \setminus \{p\}$.

Let $\|\cdot\|$ denote the L_1 norm. For the HITS and HUBAVG algorithm, there are r entries in $C_2 \setminus \{p\}$, for which $a_i = v_i = 0$ and $u_i = 1$. Therefore, for all of $\gamma_1, \gamma_2 \geq 1$, $\|\gamma_1 \boldsymbol{u} - \gamma_2 \boldsymbol{a}\| = \Omega(r) = \Omega(n)$, and $\|\gamma_1 \boldsymbol{u} - \gamma_2 \boldsymbol{a}\| = \Omega(r) = \Omega(n)$. From the above, we have that $d_r^{(0)}(\boldsymbol{u}, \boldsymbol{a}) = \Theta(1)$, and $d_r^{(0)}(\boldsymbol{u}, \boldsymbol{v}) = \Theta(1)$.

The proof for the INDEGREE algorithm, is a little more involved. Let

$$\begin{aligned} S_1 &= \sum_{i \in C_1} |\gamma_1 w_i - \gamma_2 u_i| = 2r \left| \gamma_1 - \gamma_2 - \frac{\gamma_2}{r^2} \right| \\ S_2 &= \sum_{i \in C_2 \setminus \{p\}} |\gamma_1 w_i - \gamma_2 u_i| = r \left| \gamma_1 \frac{1}{2} - \gamma_2 \right| . \end{aligned}$$

We have that $\|\gamma_1 \boldsymbol{w} - \gamma_2 \boldsymbol{u}\| \geq S_1 + S_2$. Unless $\frac{1}{2}\gamma_1 - \gamma_2 = o(1)$, then $S_2 = \Theta(r) = \Theta(n)$. If $\gamma_1 = 2\gamma_2 + o(1)$, since $\gamma_1, \gamma_2 \geq 1$, we have that $S_1 = \Theta(r) = \Theta(n)$. Therefore, $d_1(\boldsymbol{w}, \boldsymbol{u}) = \Omega(n)$. From the above, $d_r^{(0)}(\boldsymbol{w}, \boldsymbol{u}) = \Theta(1)$.

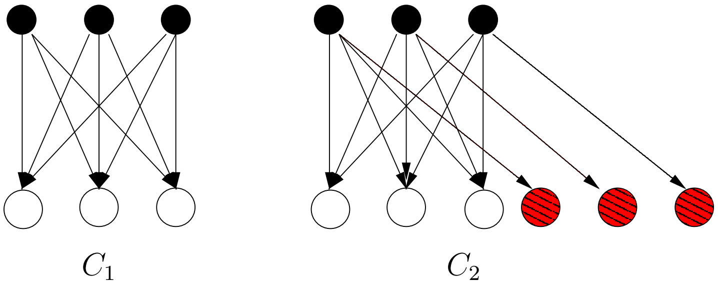


Figure 5.4: Dissimilarity of HubAvg and Hits. The graph G for $r = 3$.

Thus, Salsa is neither similar, nor weakly rank similar to Hits, InDegree, and HubAvg. □

The HubAvg algorithm

Proposition 5.3 *The* HubAvg *and* Hits *algorithms are neither similar, nor weakly rank similar on $\mathcal{G}_n$.*

Proof: Consider a graph G on $n = 5r$ nodes that consists of two disconnected components. The first component C_1 consists of a complete bipartite graph with r hub, and r authority nodes. The second component C_2 consists of a complete bipartite graph C with r hub and r authority nodes, and a set of r "external" authority nodes E, such that each hub node in C points to a node in E, and no two hub nodes in C point to the same "external" node. Figure 5.4 shows the graph G for $r = 3$.

Let $\boldsymbol{a}$ and $\boldsymbol{w}$ denote the weight vectors of the Hits and the HubAvg algorithm respectively, on graph G. It is not hard to see that the Hits algorithm allocates all the weight to the authority nodes in C_2. After normalization, for all authority nodes $i \in C$, $a_i = 1$, for all $j \in E$, $a_j = \frac{1}{r-1}$, and for all $k \in C_1$, $a_k = 0$. On the other hand, the HubAvg algorithm allocates all the weight to the nodes in C_1. After normalization, for all authority nodes $k \in C_1$, $w_k = 1$, and for all $j \in C_2$, $w_j = 0$.

Let $U = C_1 \cup C$. The set U contains $2r$ authority nodes. For every authority $i \in U$, either $a_i = 1$ and $w_i = 0$, or $a_i = 0$ and $w_i = 1$. Therefore, for all $\gamma_1, \gamma_2 \geq 1$, $\sum_{i \in U} |\gamma_1 a_i - \gamma_2 w_i| \geq 2r$. Thus, $d_1(\boldsymbol{a}, \boldsymbol{w}) = \Omega(r) = \Omega(n)$, which proves that the algorithms are not similar.

The proof for weak rank dissimilarity follows immediately from the above. For every pair of authority nodes (i, j) such that $i \in C_1$ and $j \in C_2$, $a_i < w_j$, and $a_i > w_j$. There are

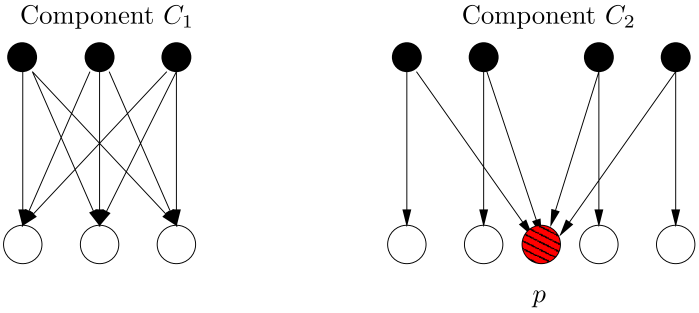


Figure 5.5: Dissimilarity of HubAvg and InDegree. The G_s graph.

$\Theta(n^2)$ such pairs, therefore, $d_r^{(0)}(\boldsymbol{a}, \boldsymbol{w}) = \Theta(1)$. Thus, the two algorithms are not weakly rank similar. □

Proposition 5.4 *The* HubAvg *algorithm and the* InDegree *algorithm are neither similar, nor weakly rank similar on G_n.*

Proof: Consider a graph G with $n = 15r$ nodes. The graph G consists of r copies of a subgraph G_s on 15 nodes. The subgraph G_s contains two components C_1 and C_2. The component C_1 is a complete bipartite graph with 3 hubs and 3 authorities. The component C_2 consists of 4 hubs that all point to an authority node p. Furthermore, each hub points to one more authority, a different one for each hub. The graph G_s is shown in Figure 5.5.

Let $\boldsymbol{a}$ denote the authority weight vector of HubAvg algorithm, and let $\boldsymbol{w}$ denote the authority weight of the InDegree algorithm on graph G. It is not hard to see that for every subgraph G_s, the HubAvg algorithm assigns all the weight to component C_1 and zero weight to component C_2. On the other hand, the InDegree algorithm assigns weight 1 to all nodes with in-degree 4, and weight 3/4 to the authorities in the C_1 components of the G_s subgraphs. Since the graph G contains r copies of the graph G_s, it follows that there are $r = \Theta(n)$ nodes for which $a_i = 0$ and $w_i = 1$. Therefore, $d_r(\boldsymbol{a}, \boldsymbol{w}) = \Theta(1)$. Furthermore, for all $\gamma_1, \gamma \geq 1$, $\|\gamma_1 \boldsymbol{a} - \gamma_2 \boldsymbol{w}\|_1 = \Theta(n)$. Thus, HubAvg and InDegree are neither similar, nor weakly rank similar □

The Max algorithm

Proposition 5.5 *The* Max *algorithm is neither similar, nor weakly rank similar to the* Hits *and the* HubAvg *algorithms on $\mathcal{G}_n$.*

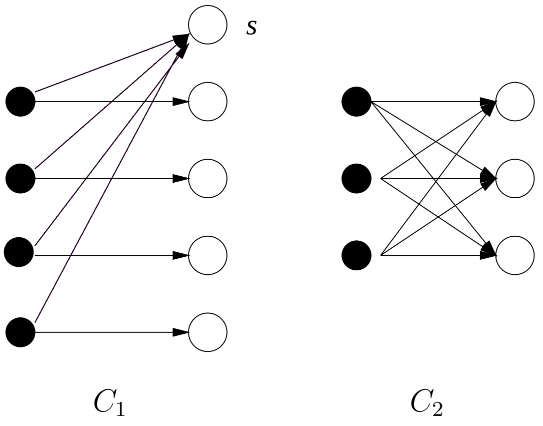


Figure 5.6: Dissimilarity of MAX with HITS and HUBAVG algorithms. The G graph for $r = 4$.

Proof: Consider a graph G on $n = 4r - 1$ nodes that consists of two disconnected components C_1 and C_2. Component C_1 contains r hubs and $r + 1$ authorities. The first authority, denoted by s, is pointed to by all r hubs, while the rest of the authorities are pointed to by exactly one hub, such that no two hubs point to the same authority. The C_2 component consists of $r - 1$ hubs and $r - 1$ authorities, arranged in $(r - 1) \times (r - 1)$ bipartite graph. Figure 5.6 shows graph G for $r = 4$.

Since the C_2 component does not contain a seed node, the MAX algorithm, assigns all weight to the C_1 component, and zero weight to the authorities of the C_2 component. Let $\boldsymbol{a}$ denote the weight of the MAX algorithm. For authority s, $a_s = 1$, while for every other authority j in C_1, $a_j = 1/r$. For every authority i in C_2, $a_i = 0$.

On the other hand, the HITS and the HUBAVG algorithms allocate all the weight to the nodes in the C_2 component, and no weight to the component C_1. If $\boldsymbol{w}$ is the weight vector of the HITS algorithm, and $\boldsymbol{v}$ the weight vector of the HUBAVG algorithm, we have that $w_i = v_i = 1$ for all authorities i in the C_2 component, while $w_j = v_j = 0$ for every authority j in the C_1 component. Since C_1 and C_2 contain $\Theta(n)$ number of authorities, the MAX algorithm and the HITS and HUBAVG algorithms, are neither similar, nor weakly rank similar. □

Proposition 5.6 *The* MAX *algorithm is neither similar, nor weakly rank similar to the* INDEGREE *(and* SALSA*) algorithm on* $\mathcal{G}_n^{AC}$.

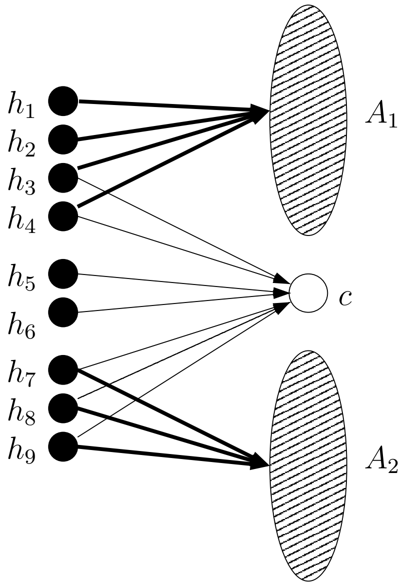


Figure 5.7: Dissimilarity of MAX with INDEGREE and SALSA algorithms.

Proof: Consider a graph G that consists of $2r + 9$ nodes. The graph consists of a "central" authority node c, a set of r authorities A_1, and a set of r authorities A_2. There are also 9 hub nodes $h_1, \ldots, h_9$. Hubs $h_1, \ldots, h_4$ point to all authorities in A_1, while hubs h_6, h_7, h_8 point to all authorities in A_2. Seven hubs, $h_3, \ldots, h_8$, point to the central authority c. The graph is shown in Figure 5.7.

Let $\boldsymbol{a}$ denote the weight vector of the MAX algorithm. The authority node c is the seed node of the algorithm, so $a_c = 1$. For every authority node i in A_1, $a_i = 2/5$, and for every authority node j in A_2, $a_j = 3/7$. Let $\boldsymbol{w}$ denote the weight vector of the INDEGREE algorithm. For the central authority c, $w_c = 1$. For every authority node $i \in A_1$, $w_i = 4/7$, and for all nodes in $j \in A_2$, $a_j = 3/7$.

Therefore, the MAX algorithm ranks the authorities in A_2 ahead of the authorities in A_1, while INDEGREE ranks the authorities in A_1 ahead of the authorities in A_2. Thus, there are r^2 pair of nodes (i, j), for which $\mathcal{I}^{(0)}_{\boldsymbol{aw}}(i, j) = 1$. Since $r = \Theta(n)$, $d^{(0)}_r(\boldsymbol{a}, \boldsymbol{w}) = \Theta(n)$. Thus MAX and INDEGREE are not weakly rank similar.

For the dissimilarity of the MAX and INDEGREE algorithms,

$$\|\gamma_1 \boldsymbol{a} - \gamma_2 \boldsymbol{w}\|_1 = r(4\gamma_1/7 - 2\gamma_2/5) + 3r/7(\gamma_1 - \gamma_2)$$

For all $\gamma_1, \gamma_2 \geq 1$, $\|\gamma_1 \boldsymbol{a} - \gamma_2 \boldsymbol{w}\|_1 = \Theta(r)$. Thus $d_1(\boldsymbol{a}, \boldsymbol{w}) = \Theta(n)$, so Max and InDegree are not similar.

In authority connected graphs the Salsa algorithm produces the same authority weights as the InDegree algorithm. Thus, we conclude that Max and Salsa are not similar or weakly rank similar. □

Other Results

On the positive side, the following lemma follows immediately from the definition of the Salsa algorithm and the definition of the authority-connected class of graphs.

Lemma 5.1 *The* Salsa *algorithm is rank equivalent to the* InDegree *algorithm on the class of authority connected graphs* $\mathcal{G}_n^{AC}$.

In a recent work, Lempel and Moran [65] showed that the Hits, InDegree (Salsa), and PageRank algorithms are not weakly rank similar on the class of authority connected graphs, $\mathcal{G}_n^{AC}$.

5.6 Stability

In the previous section, we examined the similarity of two different algorithms on the same graph G. In this section, we are interested in how the output of a *fixed* algorithm changes, as we alter the graph. We would like small changes in the graph to have a small effect on the weight vector of the algorithm. We capture this requirement by the definition of stability. The notion of stability has been independently considered (but not explicitly defined) in a number of different papers [72, 73, 4, 1]. For the definition of stability, we will use some of the terminology employed by Lempel and Moran [65].

Let $\overline{\mathcal{G}}_n$ be a class of graphs, and let $G = (P, E)$ and $G' = (P, E')$ be two graphs in $\overline{\mathcal{G}}_n$. We define the *link distance* d_ℓ between graphs G and G' as follows.

$$d_\ell\left(G, G'\right) = \left|(E \cup E') \setminus (E \cap E')\right|$$

That is, $d_\ell(G, G')$ is the minimum number of links that we need to add and/or remove so as to change one graph into the other.

The d_ℓ distance can be generalized to the case that G and G' are weighted matrices. Let W and W' denote the adjacency matrices of the graphs G and G', respectively. Then we can define the distance between the graphs as the sum of the L_1 differences of the weights.

$$d_w = \sum_{i=1}^{n} \sum_{j=1}^{n} \left| W[i,j] - W'[i,j] \right|$$

Alternatively, we could use the Frobenius matrix norm. Given a matrix W the Frobenius norm $\|W\|_F$ is defined as follows

$$\|W\|_F = \left(\sum_{i=1}^{n} \sum_{j=1}^{n} W[i,j]^2 \right)^{1/2} .$$

We then define the distance between the two graphs as

$$d_F = \left\| W - W' \right\|_F .$$

We note that in the case that W and W' are 0/1 matrices, the link distance d_ℓ is a special case of both d_w and d_F.

Given a class of graphs $\overline{\mathcal{G}}_n$, we define a *change*, ∂, within class $\overline{\mathcal{G}}_n$ as a pair $\partial = \{G, G'\}$, where $G, G' \in \overline{\mathcal{G}}_n$. The size of the change is defined as $|\partial| = d_\ell(G, G')$. We say that a change ∂ *affects* node i, if the links that point to node i are altered. In algebraic terms, the i-th column vectors of the adjacency matrices W and W' are different. We define the *impact set* of a change ∂, $\{\partial\}$, to be the set of nodes affected by the change ∂.

For a graph $G \in \mathcal{G}_n$, we define the set $\mathcal{C}_k(G) = \{G' \in \overline{\mathcal{G}}_n : d_\ell(G, G') \leq k\}$. The set $C_k(G)$ contains all graphs that have link distance at most k from graph G, that is, all graphs G' that can be produced from G, with a change of size at most k.

We are now ready to define stability. For the following, if $G = (P, E)$ is a graph in $\overline{\mathcal{G}}_n$, then we assume that $E = \omega(1)$. Otherwise, all properties that we discuss below are trivial.

Definition 5.9 *An L-algorithm $\mathcal{A}$ is stable on the class of graphs $\overline{\mathcal{G}}_n$ under distance d if for every fixed positive integer k, we have as $n \to \infty$*

$$\max_{G \in \overline{\mathcal{G}}_n, G' \in \mathcal{C}_k(G)} d(\mathcal{A}(G), \mathcal{A}(G')) = o(M_n)$$

where $M_n = \sup_{\|\boldsymbol{w}_1\|=\|\boldsymbol{w}_2\|=1} d(\boldsymbol{w}_1, \boldsymbol{w}_2)$ *is the maximum distance between any two* n*-dimensional vectors with unit norm* $L = || \cdot ||$.

We now give definitions for stability for the specific distance measures we consider.

Definition 5.10 *An* L_p*-algorithm* $\mathcal{A}$ *is* d_1*-stable (or, stable) on the class of graphs* $\overline{\mathcal{G}}_n$*, if for every fixed positive integer* k*, we have as* $n \to \infty$

$$\max_{G \in \overline{\mathcal{G}}_n, G' \in \mathcal{C}_k(\overline{\mathcal{G}}_n)} d_1(\mathcal{A}(G), \mathcal{A}(G')) = o\left(n^{1-1/p}\right)$$

Definition 5.11 *An algorithm* $\mathcal{A}$ *is* weakly rank stable *on the class of graphs* $\overline{\mathcal{G}}_n$ *if for every fixed positive integer* k*, we have as* $n \to \infty$

$$\max_{G \in \overline{\mathcal{G}}_n, G' \in \mathcal{C}_k(G)} d_r^{(0)}(\mathcal{A}(G), \mathcal{A}(G')) \;=\; o(1)$$

Definition 5.12 *An algorithm* $\mathcal{A}$ *is* strictly rank stable *on the class of graphs* $\overline{\mathcal{G}}_n$ *if for every fixed positive integer* k*, we have as* $n \to \infty$

$$\max_{G \in \overline{\mathcal{G}}_n, G' \in \mathcal{C}_k(G)} d_r^{(1)}(\mathcal{A}(G), \mathcal{A}(G')) \;=\; o(1)$$

As in the case of similarity, strict rank stability implies stability for all rank distance measures we defined in Section 5.4.2, while weak rank instability implies instability for all rank distance measures.

Stability seems to be a desirable property. If an algorithm is not stable, then slight changes in the link structure of the Base Set may lead to large changes in the rankings produced by the algorithm. Given the rapid evolution of the Web, stability is necessary to guarantee consistent behavior of the algorithm. Furthermore, stability may provide some "protection" against malicious spammers.

The following theorem is the analogue of Theorem 5.4 for stability.

Theorem 5.5 *Let* $\mathcal{A}$ *be an algorithm, and let* $1 \leq p \leq q \leq \infty$*. If the* L_p*-algorithm* $\mathcal{A}$ *is stable on class* $\overline{\mathcal{G}}_n$*, then the* L_q*-algorithm* $\mathcal{A}$ *is also stable on* $\overline{\mathcal{G}}_n$*.*

Proof: Let $\partial = \{G, G'\}$ be a change within $\overline{\mathcal{G}}_n$ of size at most k, for a fixed constant k. Set $\boldsymbol{v} = \mathcal{A}(G)$, and $\boldsymbol{u} = \mathcal{A}(G')$, and then the rest of the proof is identical to the proof of Theorem 5.4. □

Theorem 5.5 implies that, if an L_1-algorithm $\mathcal{A}$ is stable, then the L_p-algorithm $\mathcal{A}$ is also stable, for any $1 \leq p \leq \infty$. Consequently, if the L_∞-algorithm $\mathcal{A}$ is unstable, then the L_p-algorithm $\mathcal{A}$ is also unstable, for any $1 \leq p \leq \infty$. Therefore, instability results proven for the L_∞ norm hold for any L_p norm, for $1 \leq p \leq \infty$.

5.6.1 Stability and Similarity

We now prove an interesting connection between stability and similarity.

Theorem 5.6 *Let d be a distance function that is a metric, or a near metric*[1]*, over a class of graphs $\overline{\mathcal{G}}_n$. If two L-algorithms $\mathcal{A}_1$ and $\mathcal{A}_2$ are similar under d on the class $\overline{\mathcal{G}}_n$, and the algorithm $\mathcal{A}_1$ is stable under d on the lass $\overline{\mathcal{G}}_n$, then $\mathcal{A}_2$ is also stable under d on the class $\overline{\mathcal{G}}_n$.*

Proof: Let $\partial = \{G, G'\}$ be a change in $\overline{\mathcal{G}}_n$ of size k, where k is some fixed constant independent of n. Now let $\boldsymbol{w}_1 = \mathcal{A}_1(G)$, $\boldsymbol{w}_2 = \mathcal{A}_2(G)$, $\boldsymbol{w}'_1 = A(G')$, and $\boldsymbol{w}'_2 = A(G')$. Since $\mathcal{A}_1$ and $\mathcal{A}_2$ are similar, we have that $d(\boldsymbol{w}_1, \boldsymbol{w}_2) = o(M_n)$, and $d(\boldsymbol{w}'_1, \boldsymbol{w}'_2) = o(M_n)$. Since $\mathcal{A}_1$ is stable, we have that $d(\boldsymbol{w}_1, \boldsymbol{w}'_1) = o(M_n)$. Since the distance measure d is a metric, or a near metric, we have that

$$d(\boldsymbol{w}_2, \boldsymbol{w}'_2) = O(d(\boldsymbol{w}_1, \boldsymbol{w}_2) + d(\boldsymbol{w}'_1, \boldsymbol{w}'_2) + d(\boldsymbol{w}_1, \boldsymbol{w}'_1)) = o(M_n)$$

Therefore, $\mathcal{A}_2$ is stable on $\overline{\mathcal{G}}_n$. □

In Section A.1 (Appendix A), we show that $d_r^{(p)}$ is a metric for $p \geq 1/2$, and a near metric for $p < 1/2$ such that $p = \Theta(1)$. Also, the d_1 distance measure is a near metric over the set of L_1 unit vectors. The distance function $d_r^{(p)}$ for $p = o(1)$ is not a near metric.

5.6.2 Stability Results

Proposition 5.7 *The* Hits *and* HubAvg *algorithms are neither stable, nor weakly rank stable, on class $\mathcal{G}_n$.*

[1] A near metric is a distance function that is reflexive, and symmetric, and satisfies the following relaxed polygonal inequality. There is a constant c independent of n, such that for all $k > 0$, and all vectors $\boldsymbol{u}, \boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_k, \boldsymbol{v}$, $d(\boldsymbol{u}, \boldsymbol{v}) \leq c(d(\boldsymbol{u}, \boldsymbol{w}_1) + d(\boldsymbol{w}_1, \boldsymbol{w}_2) + \cdots + d(\boldsymbol{w}_k, \boldsymbol{v}))$. We define metrics and near metrics in Section A.1 (Appendix A).

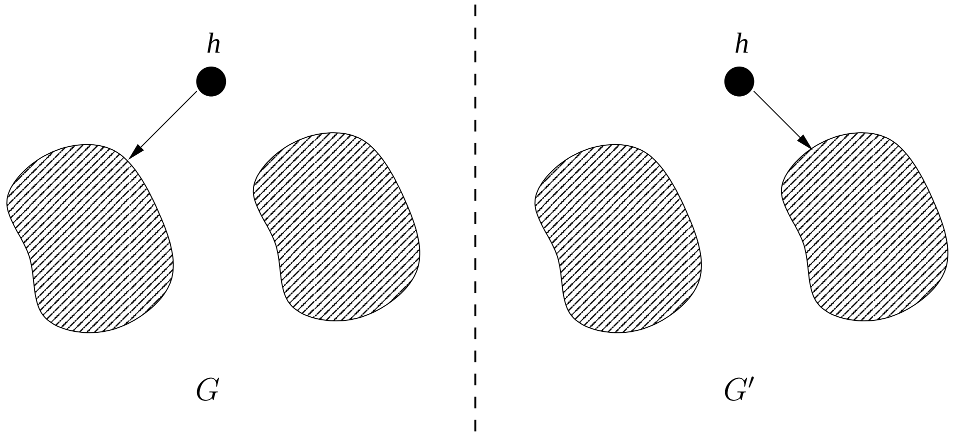


Figure 5.8: Instability of Hits and HubAvg algorithms.

Proof: Consider the graph G of size $n = 2r+1$ that consists of two disjoint components C_1 and C_2, each a complete graph on r nodes. There is also an extra hub node h that points to some node in C_1. For both Hits and HubAvg, in the corresponding matrices M_H and M_{HA}, the eigenvalue of the component C_1 is (slightly) larger than that of C_2. Therefore, both algorithms will allocate all the weight to the nodes of C_1, and zero weight to C_2. Now, construct the graph G' by removing the link from h to C_1 and adding a link to some node in C_2. The graphs G and G' are shown in Figure 5.8. In G' the eigenvalue of C_2 becomes larger than that of C_1, causing the all the weight to shift from C_1 to C_2, and leaving the nodes in C_1 with zero weight. It follows that the two algorithms are neither stable nor weakly rank stable. □

The proof of Proposition 5.7 makes use of a disconnected graph in order to establish the instability of the algorithms. Lempel and Moran [65] have recently proved that the Hits algorithm is weakly rank unstable on the class $\mathcal{G}_n^{AC}$ of authority connected graphs.

Proposition 5.8 *The* Salsa *algorithm, is neither stable, nor weakly rank stable on the class $\mathcal{G}_n$.*

Proof: We first establish the rank instability of the Salsa algorithm. The example is similar to that used in the previous proof. Consider a graph G of size $n = 2r + 1$, which consists of two disjoint components. The first component consists of a complete graph C_1 on r nodes and an extra authority p that is pointed to by a single node of the complete graph C_1. The second component consists of a complete graph C_2 on r nodes.

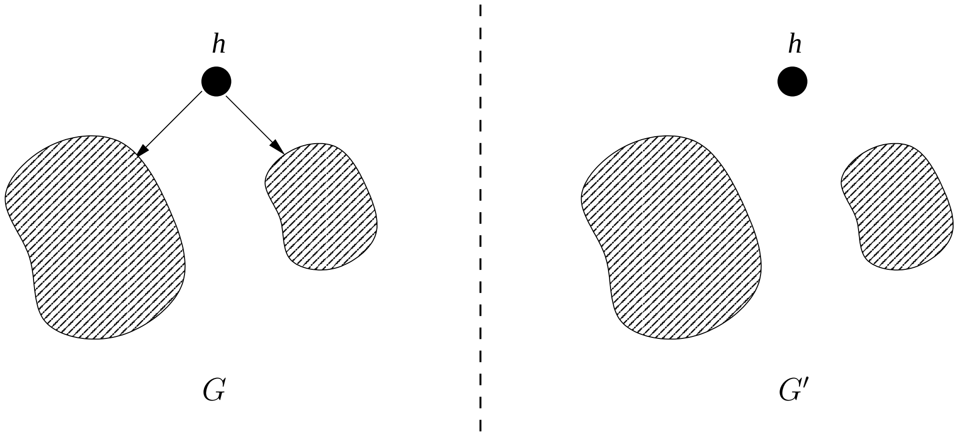


Figure 5.9: Instability of the SALSA algorithm.

Let $\boldsymbol{a}$ denote the weight vector of the SALSA algorithm on the graph G. Then for every node $i \in C_1$, $a_i = \frac{r+1}{2r+1}\frac{r-1}{r(r-1)+1}$. For every node $j \in C_2$, $a_j = \frac{1}{2r+1}$. If $r > 2$, then the SALSA algorithm ranks the r authorities in C_1 higher than those in C_2. We now remove the link from the node in C_1 to node p, and we add a link from a node in C_2 to p. Now, the nodes in C_2 are ranked higher than the nodes in C_1. There are $\Theta(n^2)$ pairs of nodes whose relative order is changed; therefore, SALSA is weakly rank unstable.

The proof of instability is a little more involved. Consider again the graph G that consists of two complete graphs C_1 and C_2 of size n_1 and n_2 respectively, such that $n_2 = cn_1$, where $c < 1$ is a fixed constant. There exists also an extra hub h that points to two authorities p and q from the components C_1 and C_2 respectively. The graph has $n = n_1 + n_2 + 1$ nodes, and $n_a = n_1 + n_2$ authorities. Figure 5.9 shows the example.

The authority Markov chain defined by the SALSA algorithm is irreducible, therefore, the weight of authority i is proportional to the in-degree of node i. Let $\boldsymbol{a}$ be the weight vector of the SALSA algorithm. Node p is the node with the highest in-degree, therefore, after normalizing in the L_∞ norm, $a_p = 1$, $a_i = 1 - 1/n_1$, for all $i \in C_1 \setminus \{p\}$, $a_q = c$, and $a_j = c - 1/n_1$ for all $j \in C_2 \setminus \{q\}$.

Now let G' be the graph G after we remove the two links from hub h to authorities p and q. Let $\boldsymbol{a}'$ denote the weight vector of the SALSA algorithm on graph G'. It is not hard to see that all authorities receive the same weight $1/n_a$ by the SALSA algorithm. After normalization, $a'_i = 1$ for all authorities i in G'.

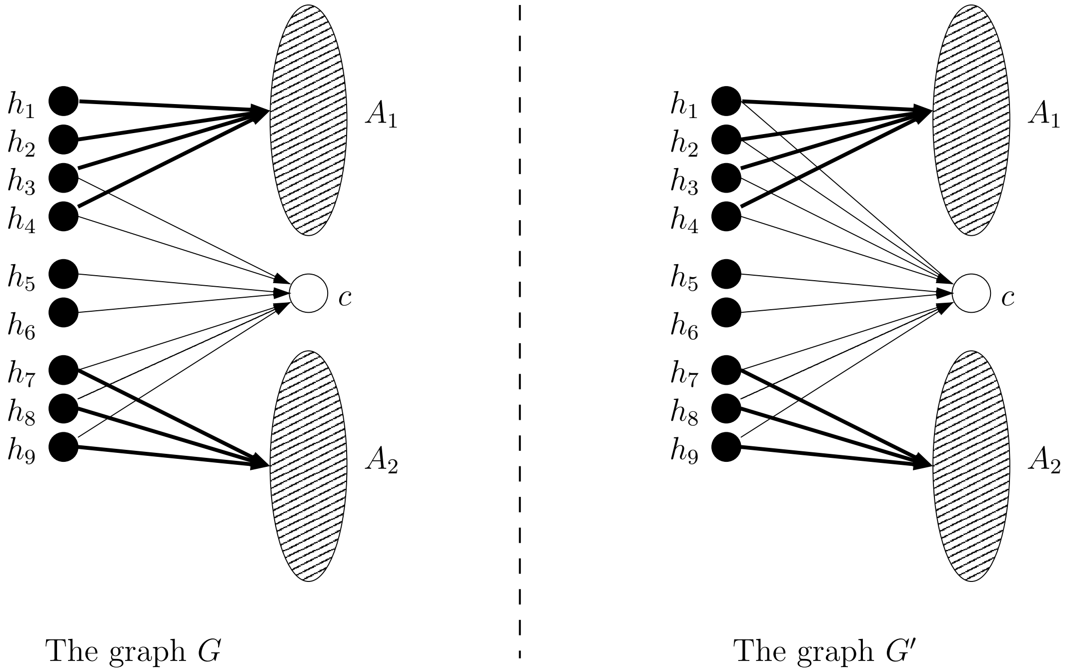


Figure 5.10: Instability of the MAX algorithm.

Consider now the difference $\|\gamma_1\boldsymbol{a} - \gamma_2\boldsymbol{a'}\|_1$. Let

$$\begin{aligned} S_1 &= \sum_{C_1\setminus\{p\}} |\gamma_1 a_i - \gamma_2 a_i'| = (n_1 - 1)\left|\gamma_1 - \gamma_2 - \frac{\gamma_1}{n_1}\right| \\ S_2 &= \sum_{C_2\setminus\{q\}} |\gamma_1 a_i - \gamma_2 a_i'| = (n_2 - 1)\left|c\gamma_1 - \gamma_2 - \frac{\gamma_1}{n_1}\right| . \end{aligned}$$

It holds that $\|\gamma_1\boldsymbol{a} - \gamma_2\boldsymbol{a'}\|_1 \geq S_1 + S_2$. It is not hard to see that unless $\gamma_1 = \frac{1}{c}\gamma_2 + o(1)$, then $S_2 = \Theta(n_2) = \Theta(n)$. If $\gamma_1 = \frac{1}{c}\gamma_2 + o(1)$, then $S_1 = \Theta(n_1) = \Theta(n)$. Therefore, $d_1(\boldsymbol{a}, \boldsymbol{a'}) = \Omega(n)$. Thus, the SALSA algorithm is unstable. □

Proposition 5.9 *The* MAX *algorithm is unstable, and weakly rank unstable, on the class of authority connected graphs* $\mathcal{G}_n^{AC}$.

Proof: For the instability of the MAX algorithm, we use the graph G defined for the dissimilarity of the MAX algorithm with the INDEGREE algorithm. We now add two links from hubs h_1 and h_2 to the central authority c to produce the graph G'. The graphs G and G' are shown in Figure 5.10.

In the graph G', the Max algorithm produces the same authority weights as the InDegree algorithm. The instability and weak rank instability of the Max algorithm follow directly from the dissimilarity of the Max and InDegree algorithms. □

We note that it is possible to cause a complete reversal of the ranking produced by the Max algorithm by just changing a few links. Lempel and Moran [65] present a proof for the weak rank instability of the Hits algorithm on $\mathcal{G}_n^{AC}$. On this graph, the Max algorithm produces exactly the same ranking as the Hits algorithm (albeit, not the same authority weights). In their counter-example, changing two links causes a complete reversal of the ranking produced.

On the positive side, we can prove that the InDegree algorithm is stable.

Theorem 5.7 *The* InDegree *algorithm is stable on the class $\mathcal{G}_n$.*

Proof: Let $\partial = \{G, G'\}$ be a change within $\mathcal{G}_n$ of size k. Let m be the size of the impact set $\{\partial\}$, where $m \leq k$. Without loss of generality assume that $\{\partial\} = \{1, 2, \ldots, m\}$. Let $\boldsymbol{u}$ be the weight vector that assigns to node i weight equal to $|B(i)|$, the in-degree of i. Let $\boldsymbol{w}$ be the weight of the L_1-InDegree algorithm. Then $\boldsymbol{w} = \boldsymbol{u}/\|\boldsymbol{u}\|$, where $\|\cdot\|$ is the L_1 norm. Let $\boldsymbol{u}'$ and $\boldsymbol{w}'$ denote the corresponding weight vectors for the graph ∂G. For all $i \notin \{1, 2, \ldots, m\}$ $u_i' = u_i$. Furthermore, $\sum_{i=1}^{m} |u_i - u_i'| \leq k$. Set $\gamma_1 = 1$ and $\gamma_2 = \frac{\|\boldsymbol{u}'\|}{\|\boldsymbol{u}\|}$. Then

$$\|\gamma_1 \boldsymbol{w} - \gamma_2 \boldsymbol{w}'\|_1 = \frac{1}{\|\boldsymbol{u}\|} \sum_{i=1}^{n} |u_i - u_i'| \leq \frac{k}{\|\boldsymbol{u}\|} \ .$$

We note that $\|\boldsymbol{u}\|$ is equal to the sum of the links in the graph; therefore, $\|\boldsymbol{u}\| = \Omega(1)$. Thus, $d_1(\boldsymbol{w}, \boldsymbol{w}') = o(1)$, which proves that L_1-InDegree, and consequently InDegree is stable. □

We examine the rank stability of InDegree in Section 5.7 where we discuss *locality.*

5.6.3 Other Results

Following the work of Borodin et al. [10], Lempel and Moran [65] proved that the Hits and PageRank algorithms are not stable on the class of authority connected graphs. Recently, Lee and Borodin [63] considered a different definition of stability, where the distance between the weights before and after the change may depend on the weights of the nodes whose in and out links where affected. The intuition is that, if a change is performed on a highly authoritative node, then we expect a large change in the weights. They prove that, under their definition, the PageRank algorithm is stable. They also prove the stability of a randomized version of Salsa, where, similar to PageRank, at each iteration you are a random jump may be performed. On the negative side, they prove that Hits and Salsa remain unstable.

5.7 Locality

We now introduce the concept of "locality". The idea behind locality is that for a local algorithm a change should not affect the relative order of the nodes that are not affected by the change.

Definition 5.13 *An algorithm $\mathcal{A}$ is local if for every change $\partial = \{G, G'\}$ there exists $\lambda > 0$ such that $\mathcal{A}(G')[i] = \lambda \mathcal{A}(G)[i]$, for all $i \notin \{\partial\}$.*

Definition 5.14 *An algorithm $\mathcal{A}$ is* weakly rank local *if for every change $\partial = \{G, G'\}$, if $\boldsymbol{a} = \mathcal{A}(G)$ and $\boldsymbol{a}' = \mathcal{A}(G')$, then, for all $i, j \notin \{\partial\}$, $a_i > a_j \Rightarrow a'_i \geq a'_j$, or $a_i < a_j \Rightarrow a'_i \leq a'_j$. (equivalently, $\mathcal{I}^{(0)}_{\boldsymbol{aa}'}(i,j) = 0$). The algorithm is* strictly rank local *if for all $i, j \notin \{\partial\}$, $a_i > a_j \Leftrightarrow a'_i > a'_j$ (equivalently, $\mathcal{I}^{(1)}_{\boldsymbol{aa}'}(i,j) = 0$).*

We note that locality and rank locality do not depend upon the normalization used by the algorithm. From the definitions, one can observe that if an algorithm is local, then it is also strictly rank local. If it is strictly rank local then it is obviously weakly rank local.

We have the following.

Theorem 5.8 *If an algorithm $\mathcal{A}$ is weakly rank local on the class $\overline{\mathcal{G}}_n$, then it is weakly rank stable on the class $\overline{\mathcal{G}}_n$. If $\mathcal{A}$ is strictly rank local on $\overline{\mathcal{G}}_n$, then it is strictly rank stable on $\overline{\mathcal{G}}_n$.*

Proof: Let $\partial = \{G, G'\}$ be a change within the class $\overline{\mathcal{G}}_n$ of size at most k. Let $\mathcal{A}$ be an algorithm defined on $\mathcal{G}_n$, let $\boldsymbol{a}$ be the weight vector of $\mathcal{A}$ on graph G, and $\boldsymbol{a}'$ be the weight vector of $\mathcal{A}$ on graph G'. Let $T = \{\partial\}$ be the impact set of change ∂, and let m be the size of the set T, where $m \leq k$. If the algorithm $\mathcal{A}$ is weakly rank local, then $\mathcal{I}^{(0)}_{\boldsymbol{aa}'}(i,j) = 0$ for all $i, j \notin T$. Therefore,

$$\begin{aligned} d_r^{(0)}(\boldsymbol{a}_1, \boldsymbol{a}'_1) &= \frac{1}{n(n-1)/2} \sum_{i=1}^{n} \sum_{p \in P} \mathcal{I}^{(0)}_{\boldsymbol{aa}'}(i,p) \\ &\leq \frac{nm}{n(n-a)/2} \leq 2k/(n-1) \\ &= o(1) \end{aligned}$$

Similarly, if the algorithm $\mathcal{A}$ is strictly rank local, $\mathcal{I}^{(1)}_{\boldsymbol{aa}'}(i,j) = 0$ for all $i, j \notin T$, and

$$\begin{aligned} d_r^{(1)}(\boldsymbol{a}, \boldsymbol{a}') &= \frac{1}{n(n-1)/2} \sum_{i=1}^{n} \sum_{p \in P} \mathcal{I}^{(1)}_{\boldsymbol{aa}'}(i,p) \\ &\leq 2k/(n-1) = o(1) \end{aligned}$$

which concludes the proof of the theorem.

□

Therefore, locality implies rank stability. It is not necessarily the case that it also implies stability. For example, consider the algorithm $\mathcal{A}$, which for a graph G on n nodes, assigns weight $n^{|B(i)|}$ to node i. This algorithm is local, but it is not stable.

Theorem 5.9 *The* InDegree *algorithm is local, and consequently strictly rank local, and rank local.*

Proof: Given a graph G, let $\boldsymbol{u}$ be the weight vector that assigns to node i weight equal to $|B(i)|$, the in-degree of i. Let $\boldsymbol{w}$ be the weight vector of the InDegree algorithm; then $w_i = u_i/\|u\| = |B(i)|/\|u\|$, where $\|\cdot\|$ is any norm.

Let $\partial = \{G, G'\}$ be a change within $\mathcal{G}_n$, and let $\boldsymbol{u}'$ and $\boldsymbol{w}'$ denote the corresponding weight vectors on graph G'. For every $i \notin \{\partial\}$, the number of links to i remains unaffected by the change ∂; therefore $u'_i = u_i$. For the InDegree algorithm, $w'_i = u'_i/\|\boldsymbol{u}'\| = u_i/\|\boldsymbol{u}'\|$. For $\lambda = \frac{\|\boldsymbol{u}\|}{\|\boldsymbol{u}'\|}$, it holds that $w'_i = \lambda w_i)$, for all $i \notin \{\partial\}$. Thus, InDegree is local, and consequently strictly rank local, and rank local. □

The following is a direct corollary of the locality of the InDegree algorithm.

Corollary 5.1 *The* InDegree *algorithm is strictly rank stable.*

The following corollary follows from the fact that the Salsa and InDegree algorithms are equivalent on the class $\mathcal{G}_n^{AC}$ of authority connected graphs.

Corollary 5.2 *The* Salsa *algorithm is stable, and strictly rank stable on* $\mathcal{G}_n^{AC}$.

5.8 An axiomatic characterization of the InDegree algorithm

We will now prove that there is a set of properties that characterize the InDegree algorithm. To this end we introduce the property of *label-independence.*

Definition 5.15 *Let* $G \in \mathcal{G}_n$ *be a graph of size* n*, and let* $\{1, 2, \ldots, n\}$ *denote a labeling of the nodes of* G*. Let* $\mathcal{A}$ *be an LAR algorithm, and let* $\boldsymbol{a} = \mathcal{A}(G)$ *denote the weight vector of* $\mathcal{A}$ *on a graph* $G \in \mathcal{G}_n$*. Let* π *denote a permutation of the labels of the nodes of* G*, and let* $\boldsymbol{a}'$ *denote the weight vector of* $\mathcal{A}$ *on the graph with the permuted labels. The algorithm* $\mathcal{A}$ *is label-independent if* $a'(\pi(i)) = a(i)$.

All the algorithms we considered in this thesis (InDegree, PageRank, Hits, Salsa, HubAvg, AT(k), Norm(p), BFS) are clearly label-independent. Label-independence is a reasonable property, but one can define reasonable algorithms that are not label-independent. For example, an algorithm may choose to give more weight to a link from a node with a specific label. The algorithm defined by Bharat and Henzinger [8], when computing the authority weight of a node i, averages the hub weights of the nodes that belong to the same domain. This algorithm is not label-independent, since it takes into account the “label” of the node when computing the authority weights.

We now state the axiomatic characterization of InDegree.

Theorem 5.10 *An algorithm* $\mathcal{A}$ *that is strictly rank local, monotone, and label-independent is rank consistent with the* InDegree *algorithm on the class* $\mathcal{G}_n$*, for any* $n \geq 3$*. If* $\mathcal{A}$ *is strictly rank local, strictly monotone, and label-independent then it is rank equivalent to the* InDegree *algorithm on the class* $\mathcal{G}_n$*, for any* $n \geq 3$.

Proof: Let G be a graph of size $n \geq 3$, and let $\boldsymbol{a} = \mathcal{A}(G)$ be the weight function of algorithm $\mathcal{A}$ on graph G, and $\boldsymbol{w}$ be the weight vector of InDegree. We will modify G to form graphs G_1, and G_2, and we use $\boldsymbol{a}_1$, and $\boldsymbol{a}_2$ to denote (respectively) the weight vector of algorithm $\mathcal{A}$ on these graphs.

Let i and j be two nodes in G. If $w_i = w_j = 0$, that is $B(i) = B(j) = \emptyset$, then from the monotonicity of $\mathcal{A}$, we have that $a_i = a_j$. Therefore $\mathcal{I}^{(0)}_{\boldsymbol{aw}}(i,j) = 0$, and $\mathcal{I}^{(1)}_{\boldsymbol{aw}}(i,j) = 0$. If

$w_i > w_j = 0$, that is, $B(i) = \emptyset$, and $B(j) \neq \emptyset$ then $B(i) \subset B(j)$. If $\mathcal{A}$ is monotone, $a_i \leq a_j$, thus $\mathcal{I}^{(0)}_{\boldsymbol{aw}}(i,j) = 0$. If $\mathcal{A}$ is strictly monotone, $a_i < a_j$, thus $\mathcal{I}^{(1)}_{\boldsymbol{aw}}(i,j) = 0$.

Now, assume that $w_i \geq w_j > 0$, or equivalently that node i has at least as many in-links as node j. The set $B(i) \cup B(j)$ of nodes that point to i or j is decomposed as follows.

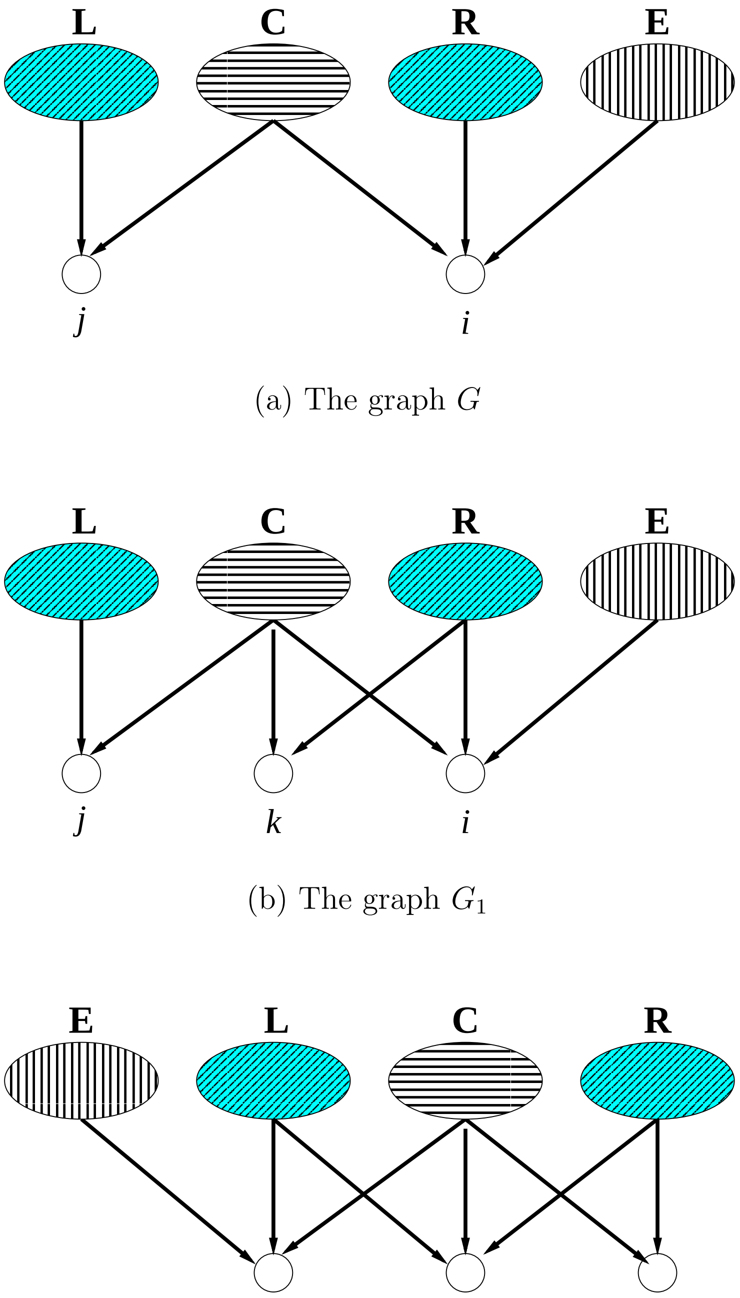
- There exists a set $C = B(i) \cap B(j)$ of nodes, that point to both i and j.
- There exists a set $L = B(j) \setminus C$ of nodes, that point to node j, but not to node i.
- There exists a set $V = B(i) \setminus C$ of nodes, that point to node i, but not to node j. The set V is further decomposed into the sets R and E. The set R is an arbitrary subset of the set V with cardinality equal to that of L. Since the in-degree of node i is at least as large as that of node j the set R is well defined. We also have that, $E = V \setminus R$.

Note that some of these sets may be empty, but not all of them are empty. Specifically, $V \cup C \neq \emptyset$, and $L \cup C \neq \emptyset$. The set E is empty if any only if nodes i and j have equal in-degrees. The links for nodes i and j are shown in Figure 5.11(a). The eliptic shapes denote sets of nodes, while the thick arrows represent the links from a set of nodes to a single node.

Let $k \neq i, j$ be an arbitrary node in the graph. We now perform the following change to graph G. We remove all links that do not point to i or j, and add links from the nodes in R and C to node k. Let G_1 denote the resulting graph. The graph G_1 is shown in Figure 5.11(b). Since $\mathcal{A}$ is strictly rank local, and the links to nodes i and j were not affected by the change, we have that

$$a_1(i) < a_1(j) \Leftrightarrow a(i) < a(j) \tag{5.2}$$

We will now prove that $a_1(k) = a_1(j)$. Assume that $a_1(k) < a_1(j)$. Let G_2 denote the graph that we obtain by removing all the links from set R to node i, and adding links from set L to node i. The graph G_2 is shown in Figure 5.11(c). We observe that the graphs G_1 and G_2 are the same up to a label permutation that swaps the labels of nodes j and k, and the labels of the nodes in L with the labels of the nodes in R. Thus, $a_2(j) = a_1(k)$, and $a_2(k) = a_1(j)$. Therefore, from our assumption that $a_1(k) < a_1(j)$, we have $a_2(j) < a_2(k)$. However, graph G_2 was obtained from graph G_1 by performing a local change to node i.



(a) The graph G

(b) The graph G_1

(c) The graph G_2

Figure 5.11: Axiomatic Characterization of InDegree

Given the strict locality assumption, the relative order of the weights of nodes j and k should remain unaffected, that is, $a_2(j) > a_2(k)$, thus reaching a contradiction. We reach the same contradiction if we assume that $a_1(k) > a_1(j)$. Therefore, it must be that $a_1(k) = a_1(j)$.

In the graph G_1, we have that $B(k) \subseteq B(i)$. We distinguish two cases. If $B(k) = B(i)$ then the set E is empty. Therefore, $w_i = w_j$, since i and j have the same number of in-links. Furthermore, from the monotonicity property (weak or strict) of the algorithm $\mathcal{A}$, we have that $a_1(i) = a_1(k) = a_1(j)$. From Equation 5.2 it follows that $a(i) = a(j)$.

If $B(k) \subset B(i)$, then the set E is not empty, and $w_i > w_j$, since i has more links than j. If $\mathcal{A}$ is monotone, then $a_1(i) \geq a_1(k) = a_1(j)$. From Equation 5.2, we have that $a(i) \geq a(j)$. Therefore, for all i, j $w_i > w_j \Rightarrow a_i \geq a_j$. Thus $d_r^{(0)}(\boldsymbol{w}, \boldsymbol{a}) = 0$, and $\mathcal{A}$ and InDegree are rank consistent. If $\mathcal{A}$ is strictly monotone, then $a_1(i) > a_1(k) = a_1(j)$. From Equation 5.2, we have that $a(i) > a(j)$. Therefore, for all i, j $w_i > w_j \Leftrightarrow a_i > a_j$. Thus $d_r^{(1)}(\boldsymbol{w}, \boldsymbol{a}) = 0$, and $\mathcal{A}$ and InDegree are rank equivalent. □

The conditions of Theorem 5.10 characterize InDegree. All three conditions, label independence, (strict) monotonicity, and strict rank locality, are necessary for the proof of the theorem. Assume that we discard the label independence condition. Now, define algorithm $\mathcal{A}$ that assigns to each link a weight that depends on the label of the node from which the link originates. The algorithm sets the authority weight of each node to be the sum of the link weights that point to this node. This algorithm is clearly monotone and local, however if the link weights are chosen appropriately, it will not be rank consistent with InDegree. Assume now that we discard the monotonicity condition. Define an algorithm $\mathcal{A}$, that assigns weight 1 to each node with odd in-degree, and weight 0 to each node with even in-degree. This algorithm is local and label independent, but it is clearly not rank consistent with InDegree. Monotonicity and label independence are clearly not sufficient for proving the theorem; we have provided examples of algorithms that are monotone and label independent, but not rank consistent with the InDegree (e.g. the Hits algorithm). Strict monotonicity is necessary for proving rank equivalence. The algorithm that assigns equal weight to all nodes is monotone, label independent, and strictly rank local. It is rank consistent with InDegree, but not rank equivalent.

Chapter 6

Experimental Evaluation

In this section we present an experimental evaluation of the algorithms that we propose, as well as some of the existing algorithms discussed in Chapter 2. We study the rankings they produce, and how they relate to each other. The goal of this experimental study is to assess the quality of the algorithms, and, more importantly, to understand how theoretically predicted properties manifest themselves in a practical setting.

6.1 The Experimental Set-Up

6.1.1 The queries

We experiment with our algorithms on the following 34 different queries.

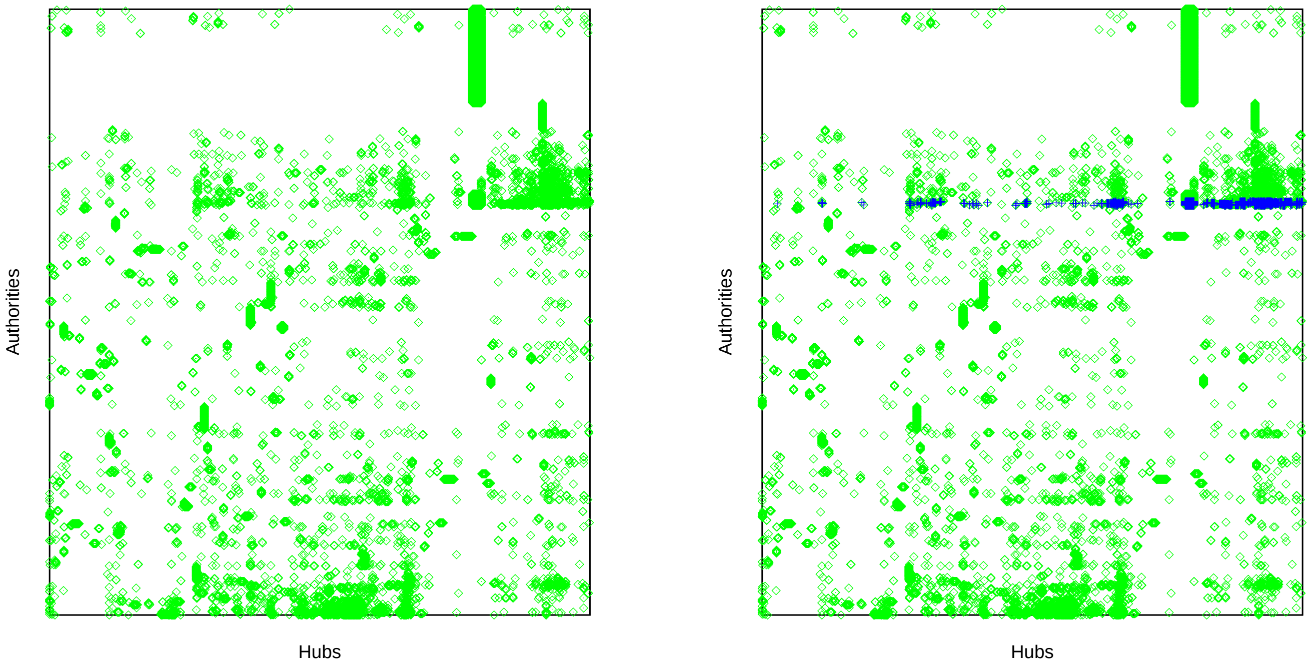
> `abortion, affirmative action, alcohol, amusement parks, architecture, armstrong, automobile industries, basketball, blues, cheese, classical guitar, complexity, computational complexity, computational geometry, death penalty, genetic, geometry, globalization, gun control, iraq war, jaguar, jordan, moon landing, movies, national parks, net censorship, randomized algorithms, recipes, roswell, search engines, shakespeare, table tennis, weather, vintage cars`

Many of these queries have already appeared in previous works [30, 64, 58]. For the remaining queries, we selected queries that correspond to topics for which there are opposing communities, such as, "death penalty", "gun control", "iraq war", "globalization", "moon landing", or queries that are of interest to different communities, depending on the interpretation of the word (for example, "jordan", "complexity", "armstrong"). Our objective is to observe how the different algorithms represent these different (usually unrelated, and sometimes opposing) communities in the top positions of the ranking. We are also interested in observing the behavior of the algorithms when we move from a broad topic ("geometry", "complexity") to a more specific subset of this topic ("computational complexity", "computational geometry"). We also selected some queries for which we expect the most relevant results not to contain the query words. For example, most search engines do not contain the words "search engine". The same for the query "automobile industries" which is a variation of the query "automobile manufacturers" that appears in the work of Kleinberg [58]. The remaining queries were selected as interesting queries on broad topics.

The Base Sets for these queries are constructed in the fashion described by Kleinberg [57]. We start with a Root Set of pages related to the query. This Root Set is obtained by querying the Google[1] search engine. The Root Set consists of the first 200 pages returned by the search engine. This set is then expanded to the Base Set by including nodes that point to, or are pointed to, by the pages in the Root Set. Following the guidelines of Kleinberg [58], for every page in the Root Set, we include only the first 50 pages that point to this page, in the order that they are returned by the Google search engine. We then extract the links between the pages of the Base Set, and we construct the hyperlink graph.

The next step is to eliminate the *navigational* links. These are links that serve solely the purpose of navigating within a Web site, and they do not convey an endorsement for the contents of the target page. Finding navigational links is a non-trivial problem that has received some attention in the literature [21, 9]. We adopt the following heuristics for identifying navigational links. First, we compare the IP addresses of the two links. If the first three bytes are the same then we label the link as navigational. If not, we look at the actual URL. This is of the form "$\texttt{http://string}_1\texttt{/string}_2\texttt{/ ...}$". The domain identifier is $\texttt{string}_1$. This is of the form " $\mathtt{x}_1.\mathtt{x}_2.\cdots.\mathtt{x}_k$". If $k \geq 3$ then we use $\mathtt{x}_2.\cdots.\mathtt{x}_{k-1}$ as the domain identifier. If $k = 2$ we use $\mathtt{x}_1$. If the domain identifiers are the same for the source and target pages of the link, then the link is labeled as navigational, and it is discarded. After the navigational links are removed, we remove any isolated nodes, and we produce the Base Set P and the graph $G = (P, E)$. Unfortunately, our heuristics do not eliminate

[1] http://www.google.com



(a) Matrix Plot

(b) Top-10 results of Hits

Figure 6.1: Matrix plots for query "abortion"

all possible navigational links, which in some cases results in introducing clusters of pages from the same domain.

Table 6.1 presents statistics for our graphs. The "med out" is the median out-degree, the "avg-out" is the average out-degree, where median and average are taken over all hub nodes. The "ACC size" is the size of the largest authority connected component, that is, the size of the largest connected component in the authority graph G_A. Recall that the graph G_A is a graph defined on the authority nodes, where there exists an edge between two authorities if they have a hub in common.

For the purpose of exhibition, we will often represent the graph for a query as a plot. Figure 6.1(a) shows the graph for the query "abortion". This figure plots a matrix, where the rows of the matrix correspond to the authority nodes in the graph, and the columns to the hub nodes. Every point corresponds to an edge in the graph. Each authority node is represented as a row of the matrix plot, that is, as a vector of hubs that point to this authority node.

In order to reveal some of the structure of the graph, the rows and columns are permuted so that similar nodes are brought together. For this, we used LIMBO, an agglom-

query	**nodes**	**hubs**	**authorities**	**links**	**med out**	**avg out**	**ACC size**
abortion	3340	2299	1666	22287	3	9.69	1583
affirmative action	2523	1954	4657	866	1	2.38	752
alcohol	4594	3918	1183	16671	2	4.25	1124
amusement parks	3410	1893	1925	10580	2	5.58	1756
architecture	7399	5302	3035	36121	3	6.81	3003
armstrong	3225	2684	889	8159	2	9.17	806
automobile industries	1196	785	561	3057	2	3.89	443
basketball	6049	5033	1989	24409	3	4.84	1941
blues	5354	4241	1891	24389	2	5.75	1838
cheese	3266	2700	1164	11660	2	4.31	1113
classical guitar	3150	2318	1350	12044	3	5.19	1309
complexity	3564	2306	1951	13481	2	5.84	1860
computational complexity	1075	674	591	2181	2	3.23	497
computational geometry	2292	1500	1294	8189	3	5.45	1246
death penalty	4298	2659	2401	21956	3	8.25	2330
genetic	5298	4293	1732	19261	2	4.48	1696
geometry	4326	3164	1815	13363	2	4.22	1742
globalization	4334	2809	2135	17424	2	8.16	1965
gun control	2955	2011	1455	11738	3	5.83	1334
iraq war	3782	2604	1860	15373	3	5.90	1738
jaguar	2820	2268	936	8392	2	3.70	846
jordan	4009	3355	1061	10937	2	3.25	991
moon landing	2188	1316	1179	5597	2	4.25	623
movies	7967	6624	2573	28814	2	4.34	2409
national parks	4757	3968	1260	14156	2	3.56	1112
net censorship	2598	1618	1474	7888	2	4.87	1375
randomized algorithms	742	502	341	1205	1	2.40	259
recipes	5243	4375	1508	18152	2	4.14	1412
roswell	2790	1973	1303	8487	2	4.30	1186
search engines	11659	7577	6209	292236	5	38.56	6157
shakespeare	4383	3660	1247	13575	2	3.70	1199
table tennis	1948	1489	803	5465	2	3.67	745
weather	8011	6464	2852	34672	3	5.36	2775
vintage cars	3460	2044	1920	12796	3	6.26	1580

Table 6.1: Query statistics

erative hierarchical clustering algorithm [3], which is based on the Information Bottleneck method [88, 86]. The distance between two vectors is measured by normalizing the vectors so that the sum of their entries is 1, and then taking the Jensen-Shannon divergence [68] of the two distributions. This corresponds to the information we lose about the entries of the vectors if we merge them [88]. Any other hierarchical algorithm for clustering binary vectors would also be applicable. Executing the algorithm on the rows of the matrix produces a tree, where each node in the tree corresponds to the merge of two clusters. The leaves of this tree are the rows of the matrix (the authority nodes). If we perform a depth first traversal of this tree and we output the leaves in the order in which we visit them, then we expect similar rows to be brought together. We perform the same operation for the columns of the graph. We do not claim that these permutations of rows and columns are optimal in any sense. The purpose of the permutations is to enhance the visualization of the graph by grouping together some of the similar rows and columns.

The matrix plot representation of the graph is helpful in identifying the parts of the graph on which the various algorithms focus in the top-10 results, by highlighting the corresponding rows. For example, Figure 6.1(b) shows again the plot of the matrix for the "abortion" dataset. The rows in darker color correspond to the top-10 authority nodes of the Hits algorithm. These matrix plots allow us to inspect how the top-10 results of the different LAR algorithms are interconnected with each other and with the rest of the graph, and they yield significant insight in the behavior of the algorithms.

6.1.2 Algorithms

We implemented all the algorithms we described in Chapter 3, namely HubAvg, AT(k), Norm(p), Max and BFS. For the Norm(p) family, we set $p = 2$ and we denote this algorithm as Norm. For the AT(k) family of algorithms, given a graph G, we compute the distribution of the out-degrees in the graph and we experiment with k being the median, and the average out-degree, where median and average are taken over all the hub nodes. We denote these algorithms as AT-med and AT-avg respectively. We also perform a separate study on how the behavior of the AT(k) and Norm(p) algorithms is affected as we vary p and k.

We also implemented the Hits, PageRank, InDegree and Salsa algorithms. For the PageRank algorithm, the jump probability ϵ usually takes values in the interval

$[0.1, 0.2]$ [13, 72]. We observed that the performance of the PageRank algorithm usually improves as we increase the value of ϵ. We set the jump probability ϵ to be 0.2, a value that is sufficiently low, and produces satisfactory results.

For all algorithms, we iterate until the L_1 difference of the authority weight vectors in two successive iterations becomes less than $\delta = 10^{-7}$, or until 1000 iterations have been completed. Although there are more sophisticated methods for testing for convergence, we chose this one for the sake of simplicity. In most cases, the algorithms converge in no more than a hundred iterations.

6.1.3 Measures

The measure that we will use for the evaluation of the quality rankings is *precision over top-10*. This is the fraction of documents in the top 10 positions of the ranking that are relevant to the query. Given the impatient nature of the Web users, we believe that this is an appropriate measure for the evaluation of Web searching and ranking algorithms. Indeed, a search engine is often judged by the first page of results it returns. We will also refer to this fraction as the *relevance ratio* of the algorithm. Similar quality measures are used in the TREC conferences for evaluating Web search algorithms.[2]

We also use a more refined notion of relevance. Given a query, we classify a document as non-relevant, relevant, or highly relevant to the topic of the query. High relevance is meant to capture the notion of authoritativeness. A highly relevant document is one that you would definitely want to be in the few first page of results of a search engine. For example, in the query "movies", the Web page `http://abeautifulmind.com/`, the official site for the movie "A Beautiful Mind", is relevant to the topic of movies, but it cannot be thought of as highly relevant. However the page `http://www.imdb.com`, the Internet Movie Data Base (IMDB) site that contains movie information and reviews, is a page that is highly relevant to the topic. This is a result that a Web user would most likely want to retrieve when posing the query. The notion of high relevance is also employed in the TREC conference for topic distillation queries, where the objective is to find the most authoritative pages for a specific topic. For each algorithm we want to estimate the *high relevance ratio*, the fraction of the top 10 results that are highly relevant. Note that every highly relevant

[2] For TREC data relevance and high relevance is usually predefined by a set of experts.

page is of course relevant, so the high relevance ratio is always less or equal to the relevance ratio.

We are also interested in studying how the algorithms relate to each other. For the comparison of two rankings we will use the geometric distance measure, d_1, defined in Section 5.4.1, and the strict rank distance $d_r^{(1)}$, defined in Section 5.4.2. We do not consider the weak rank distance in this chapter. For brevity, we will refer to the strict rank distance as rank distance, and we will denote it by d_r. When computing the d_1 distance the vectors are normalized so that the entries sum to 1. Thus, the maximum d_1 distance is 2.

We will also consider the following two similarity measures for comparing the top-k results of two algorithms.

- *Intersection over top k, $I(k)$*: The number of documents that the two rankings have in common in the top k results. In our experiments, we use $k = 10$.

- *Weighted Intersection over top k, $WI(k)$*: This is the average intersection over the top k results, where the average is taken over the intersection over the top-1, top-2, up to top-k. The weighted intersection is given by the following formula.

$$WI(k) = \frac{1}{k}\sum_{i=1}^{k} I(i)$$

In our study, we measure the similarity over the top-10 results.

6.1.4 User Study

In order to assess the relevance of the documents we performed a user study. The study was performed on-line.[3] The introductory page contained the queries with links to the results, together with some instructions. By clicking on a query, the union of the top-10 results of all algorithms was presented to the user. The results were permuted, so that they appeared in a random order, and no information was revealed about the algorithm(s) that introduced each result in the collection. The users were then asked to rate each document as “Highly Relevant”, “Relevant”, or “Non-Relevant”. They were instructed to mark a page as “Highly Relevant” any page that they would definitely like to see within the top positions of a search engine. An option “Don’t Know” (chosen as default) was also given,

[3] The URL for the study is `http://www.cs.toronto.edu/~tsap/cgi-bin/entry.cgi`

in the case that the user could not assess the relevance of the result. When pressing the submit button, their feedback was stored into a file. No information was recorded about the users, respecting their anonymity. No queries were assigned to any users, they were free to do whichever ones, and however many they wanted. On average 7 users rated each query. The maximum was 22, for the query "abortion", and the minimum 3, for the query "computational geometry". The number of users per query are shown in Table 6.2. The study was conducted mostly among friends and fellow grad students. Although they are not all experts on all of the topics, we believe that their feedback gives a useful indication about the actual relevance of the documents.[4]

We utilize the user's feedback in the following two ways. First, we compute the average relevance, and high relevance ratios for each algorithm, where the averages are taken over all users and all queries. The ratings of a user for the documents of a query induce a relevance ratio for each algorithm. Taking the average over users of these ratios, we obtain a relevance ratio for the pair (query, algorithm). Taking the average over all queries we obtain an overall relevance ratio for the algorithm.

Since our users are not experts in all topics, their feedback is bound to introduce some noise. For example, a user marked the IMDB site[5] as "Don't Know" for the query "movies", while some other user marked all the pro-life pages as non-relevant. Given that the average number of users per query is low, this introduces a measurable error. In order to reduce the effects of such errors, we also make the following use of the user feedback. Given the users' feedback for a specific document, we rate the document as "Relevant" if the "Relevant" and "Highly Relevant" votes are more than the "Non-Relevant" votes (ties are resolved in favor of "Non-Relevant"). Among the documents that are deemed as "Relevant", we rate as "Highly Relevant" the ones for which the "Highly Relevant" votes are more than the "Relevant" ones. We can now compute the relevance ratios for the algorithms by using the relevance ratings of the documents. We will refer to these measures as *labeled* ratios (since the documents are labeled as relevant or non-relevant), to discriminate them from the *non-labeled* ratios we defined before. Note that in the labeled case the "Don't Knows" do not affect the assessed quality of the algorithm. On the other hand, we get a coarse grain evaluation. A document which was judged as "Not Relevant" by 50% of the users will be

[4] The results of the study may be slightly biased towards the opinion of people with Greek origin.

[5] The Internet Movie Database, `http://www.imdb.com`

query	**users**
abortion	22
affirmative action	7
alcohol	8
amusement parks	8
architecture	7
armstrong	8
automobile industries	7
basketball	12
blues	8
cheese	5
classical guitar	8
complexity	4
computational complexity	4
computational geometry	3
death penalty	9
genetic	7
geometry	7
globalization	5
gun control	7
iraq war	8
jaguar	5
jordan	4
moon landing	8
movies	10
national parks	6
net censorship	4
randomized algorithms	5
recipes	10
roswell	4
search engines	5
shakespeare	6
table tennis	6
weather	9
vintage cars	5
average	7

Table 6.2: Users per query

treated in the same way as a document that was judged "Not Relevant" by all users.

6.2 Evaluation of the LAR algorithms

In this section we study the aggregate behavior of the algorithms. Appendix C contains tables with the top-10 results for all queries. In these tables, the results that are labeled highly relevant appear in boldface, while the relevant ones appear in italics. Due to space constraints we omit the results of the SALSA algorithm. In almost all queries, they are identical to those of INDEGREE. The results for all queries are also posted at `http://www.cs.toronto.edu/~tsap/experiments/thesis/` in a format that is easy to understand and navigate. We strongly encourage the reader to browse through the results while reading this part of the thesis.

Tables 6.3, 6.5, 6.4, 6.6 report the quality ratios of each algorithm for each query. In each row the highlighted value is the best ratio for this query. For each algorithm we also compute the average, the standard deviation, the minimum and the maximum values for all quality ratios. We report both the labeled and non-labeled ratios. In most cases, there is only a small difference between the two measures, and the general trends remain consistent regardless of the measure that we use, so we will not distinguish between the two. Recall, that for the relevance ratio, we consider documents that are marked either "Relevant" or "Highly Relevant", so the relevance ratio is always greater or equal to the high relevance ratio. For the purpose of comparing between algorithms, we also report the average values of all our similarity measures in Appendix B.

The qualitative evaluation reveals that all algorithms fall, to some extent, victim to *topic drift*. That is, they promote pages that are not related to the topic of the query. In terms of high relevance, on average, more than half (and as many as 8 out of 10 for the case of HITS) of the results in the top-10 are not highly relevant. This is significant for the quality of the algorithms, since the highly relevant documents represent the ones that the users would actually want to see in the top positions of the ranking, as opposed to the ones that they found just relevant to the topic. The performance improves when we consider relevance, instead of high relevance. Still, the average relevance ratio is never more than 78%, that is, even for the best algorithm, on average 2 out of the top 10 documents are irrelevant to the query. Furthermore, there exist queries such as "armstrong", and "jaguar",

query	Hits	PageRank	InDegree	Salsa	HubAvg	Max	AT-med	AT-avg	Norm	BFS
abortion	35%	15%	**45%**	**45%**	41%	42%	40%	38%	40%	37%
affirmative action	33%	9%	36%	36%	4%	4%	4%	4%	4%	**51%**
alcohol	55%	40%	56%	56%	54%	50%	50%	49%	49%	**59%**
amusement parks	52%	11%	24%	38%	0%	**65%**	10%	0%	0%	39%
architecture	0%	31%	**51%**	**51%**	0%	44%	47%	0%	0%	37%
armstrong	0%	10%	4%	4%	0%	0%	0%	0%	0%	**14%**
automobile industries	1%	9%	11%	20%	3%	3%	3%	3%	3%	**40%**
basketball	0%	49%	18%	18%	0%	9%	9%	9%	9%	**51%**
blues	40%	36%	40%	40%	**48%**	42%	45%	46%	40%	31%
cheese	0%	4%	8%	8%	4%	0%	0%	4%	0%	**20%**
classical guitar	**41%**	24%	26%	26%	9%	35%	9%	9%	11%	**41%**
complexity	0%	28%	20%	20%	0%	**60%**	45%	0%	0%	45%
computational complexity	38%	40%	42%	42%	**45%**	38%	38%	38%	38%	30%
computational geometry	47%	20%	37%	37%	37%	47%	43%	40%	40%	**50%**
death penalty	68%	43%	58%	58%	42%	64%	64%	68%	68%	**69%**
genetic	**66%**	19%	57%	57%	50%	59%	59%	59%	59%	50%
geometry	**54%**	11%	49%	49%	46%	**54%**	**54%**	49%	**54%**	53%
globalization	2%	28%	22%	22%	4%	4%	4%	4%	4%	**36%**
gun control	0%	33%	**63%**	**63%**	60%	60%	60%	56%	60%	60%
iraq war	12%	11%	15%	15%	0%	10%	0%	0%	0%	**42%**
jaguar	0%	**22%**	2%	2%	2%	2%	2%	0%	0%	8%
jordan	0%	15%	25%	25%	20%	**42%**	**42%**	**42%**	0%	30%
moon landing	0%	20%	12%	12%	0%	0%	0%	0%	0%	**72%**
movies	9%	13%	27%	24%	34%	**55%**	**55%**	53%	**55%**	30%
national parks	0%	38%	7%	7%	**48%**	**48%**	**48%**	0%	0%	**48%**
net censorship	2%	25%	70%	70%	50%	**72%**	**72%**	**72%**	**72%**	70%
randomized algorithms	8%	**30%**	8%	8%	0%	2%	2%	2%	2%	8%
recipes	0%	10%	46%	46%	11%	56%	56%	**65%**	0%	47%
roswell	0%	5%	12%	12%	20%	**25%**	22%	0%	0%	20%
search engines	48%	64%	**84%**	**84%**	**84%**	**84%**	**84%**	82%	38%	74%
shakespeare	28%	35%	60%	60%	62%	**67%**	**67%**	**67%**	63%	63%
table tennis	**55%**	25%	53%	53%	50%	52%	52%	52%	52%	43%
weather	53%	26%	58%	58%	32%	53%	49%	49%	49%	**64%**
vintage cars	0%	2%	34%	34%	0%	**38%**	36%	0%	0%	34%
avg	22%	24%	35%	35%	25%	38%	34%	28%	24%	**43%**
max	68%	64%	**84%**	**84%**	**84%**	**84%**	**84%**	82%	72%	74%
min	0%	2%	2%	2%	0%	0%	0%	0%	0%	**8%**
stdev	24%	14%	22%	21%	25%	25%	26%	28%	26%	18%

Table 6.3: High Relevance Ratio – Non-Labeled case

query	Hits	PageRank	InDegree	Salsa	HubAvg	Max	AT-med	AT-avg	Norm	BFS
abortion	72%	52%	**82%**	**82%**	81%	80%	**82%**	80%	79%	80%
affirmative action	57%	37%	46%	46%	16%	11%	11%	11%	11%	**71%**
alcohol	**84%**	59%	82%	82%	84%	76%	76%	75%	75%	**84%**
amusement parks	**95%**	31%	30%	50%	0%	90%	11%	0%	0%	79%
architecture	7%	67%	**70%**	**70%**	11%	63%	70%	7%	7%	59%
armstrong	19%	**48%**	21%	21%	19%	19%	19%	19%	19%	**48%**
automobile industries	10%	10%	20%	30%	10%	10%	10%	10%	10%	**61%**
basketball	0%	68%	19%	19%	10%	9%	9%	9%	9%	**87%**
blues	61%	**75%**	60%	60%	70%	61%	68%	70%	61%	55%
cheese	0%	26%	30%	30%	14%	0%	0%	14%	0%	**48%**
classical guitar	**89%**	48%	64%	64%	43%	75%	43%	43%	44%	**89%**
complexity	0%	38%	32%	32%	0%	**70%**	62%	0%	0%	65%
computational complexity	85%	68%	85%	85%	**85%**	85%	85%	85%	85%	80%
computational geometry	80%	30%	57%	57%	57%	**87%**	57%	57%	57%	83%
death penalty	**98%**	73%	88%	88%	70%	97%	97%	97%	97%	97%
genetic	**94%**	60%	91%	91%	89%	91%	91%	91%	91%	81%
geometry	**87%**	30%	81%	81%	80%	84%	84%	77%	84%	84%
globalization	82%	54%	**86%**	**86%**	84%	**86%**	**86%**	84%	84%	82%
gun control	0%	51%	**97%**	**97%**	94%	94%	94%	93%	94%	94%
iraq war	39%	35%	39%	39%	21%	32%	25%	20%	25%	**85%**
jaguar	0%	**32%**	4%	4%	10%	8%	4%	0%	0%	16%
jordan	0%	32%	42%	42%	38%	**85%**	**85%**	**85%**	0%	48%
moon landing	0%	31%	19%	19%	0%	0%	0%	0%	0%	**99%**
movies	9%	21%	45%	41%	48%	**70%**	**70%**	69%	**70%**	55%
national parks	0%	57%	10%	10%	**82%**	**82%**	**82%**	0%	0%	78%
net censorship	18%	38%	77%	77%	62%	**85%**	**85%**	**85%**	**85%**	77%
randomized algorithms	66%	**78%**	68%	68%	50%	52%	52%	54%	54%	56%
recipes	0%	27%	69%	69%	29%	89%	89%	**98%**	0%	79%
roswell	12%	20%	38%	38%	52%	**70%**	62%	8%	8%	50%
search engines	76%	86%	94%	94%	94%	94%	94%	**96%**	64%	84%
shakespeare	**100%**	60%	98%	98%	97%	97%	97%	97%	97%	**100%**
table tennis	92%	57%	**97%**	**97%**	93%	93%	93%	93%	93%	88%
weather	80%	51%	82%	82%	59%	80%	76%	76%	76%	**92%**
vintage cars	20%	10%	62%	62%	20%	60%	60%	20%	20%	**64%**
avg	45%	46%	58%	59%	49%	64%	60%	51%	44%	**73%**
max	**100%**	86%	98%	98%	97%	97%	97%	98%	97%	**100%**
min	0%	**10%**	4%	4%	0%	0%	0%	0%	0%	**16%**
stdev	24%	14%	22%	21%	25%	25%	26%	28%	26%	18%

Table 6.4: Relevance Ratio – Non-Labeled case

query	Hits	PageRank	InDegree	Salsa	HubAvg	Max	AT-med	AT-avg	Norm	BFS
abortion	30%	10%	**40%**	**40%**	30%	**40%**	30%	30%	**40%**	30%
affirmative action	30%	0%	40%	40%	0%	0%	0%	0%	0%	**60%**
alcohol	60%	30%	60%	60%	60%	50%	50%	50%	50%	**70%**
amusement parks	50%	10%	30%	40%	0%	**70%**	10%	0%	0%	40%
architecture	0%	30%	**70%**	**70%**	0%	60%	60%	0%	0%	50%
armstrong	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
automobile industries	0%	10%	10%	20%	0%	0%	0%	0%	0%	**40%**
basketball	0%	**60%**	20%	20%	0%	10%	10%	10%	10%	**60%**
blues	**60%**	40%	40%	40%	50%	50%	50%	50%	**60%**	20%
cheese	0%	0%	0%	0%	0%	0%	0%	0%	0%	**10%**
classical guitar	**40%**	30%	30%	30%	0%	**40%**	0%	0%	10%	**40%**
complexity	0%	30%	20%	20%	0%	**70%**	50%	0%	0%	50%
computational complexity	30%	30%	30%	30%	**40%**	30%	30%	30%	30%	20%
computational geometry	40%	20%	40%	40%	40%	40%	**50%**	40%	40%	40%
death penalty	70%	30%	70%	70%	50%	**80%**	**80%**	**80%**	**80%**	**80%**
genetic	**80%**	20%	70%	70%	60%	70%	70%	70%	70%	60%
geometry	60%	10%	50%	50%	40%	**70%**	**70%**	60%	**70%**	60%
globalization	0%	**30%**	20%	20%	0%	0%	0%	0%	0%	**30%**
gun control	0%	50%	**70%**	**70%**	60%	60%	60%	60%	60%	60%
iraq war	0%	10%	10%	10%	0%	10%	0%	0%	0%	**40%**
jaguar	0%	**20%**	0%	0%	0%	0%	0%	0%	0%	10%
jordan	0%	10%	20%	20%	20%	**40%**	**40%**	**40%**	0%	30%
moon landing	0%	20%	10%	10%	0%	0%	0%	0%	0%	**80%**
movies	10%	10%	30%	30%	40%	**70%**	**70%**	**70%**	**70%**	40%
national parks	0%	50%	10%	10%	**60%**	**60%**	**60%**	0%	0%	50%
net censorship	0%	20%	**80%**	**80%**	60%	**80%**	**80%**	**80%**	**80%**	**80%**
randomized algorithms	0%	**40%**	10%	10%	0%	0%	0%	0%	0%	10%
recipes	0%	10%	60%	60%	10%	60%	60%	**70%**	0%	50%
roswell	0%	0%	0%	0%	0%	**10%**	**10%**	0%	0%	**10%**
search engines	60%	70%	**100%**	**100%**	**100%**	**100%**	**100%**	**100%**	40%	90%
shakespeare	0%	20%	50%	50%	50%	**70%**	**70%**	**70%**	60%	60%
table tennis	**50%**	20%	**50%**	**50%**	**50%**	**50%**	**50%**	**50%**	**50%**	40%
weather	60%	20%	60%	60%	30%	60%	50%	50%	50%	**70%**
vintage cars	0%	0%	**40%**	**40%**	0%	**40%**	**40%**	0%	0%	30%
avg	21%	22%	36%	37%	25%	41%	37%	30%	26%	**44%**
max	80%	70%	**100%**	**100%**	**100%**	**100%**	**100%**	**100%**	**100%**	**100%**
min	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
stdev	27%	17%	26%	26%	28%	30%	31%	32%	30%	23%

Table 6.5: High Relevance Ratio – Labeled case

query	Hits	PageRank	InDegree	Salsa	HubAvg	Max	AT-med	AT-avg	Norm	BFS
abortion	90%	70%	**100%**	**100%**	**100%**	**100%**	**100%**	**100%**	**100%**	**100%**
affirmative action	70%	50%	50%	50%	10%	10%	10%	10%	10%	**80%**
alcohol	**90%**	60%	**90%**	**90%**	**90%**	80%	80%	80%	80%	**90%**
amusement parks	**100%**	30%	30%	50%	0%	90%	10%	0%	0%	80%
architecture	10%	**70%**	**70%**	**70%**	10%	60%	**70%**	10%	10%	60%
armstrong	20%	**50%**	20%	20%	20%	20%	20%	20%	20%	**50%**
automobile industries	10%	10%	20%	30%	10%	10%	10%	10%	10%	**60%**
basketball	0%	70%	20%	20%	0%	10%	10%	10%	10%	**100%**
blues	60%	**80%**	60%	60%	70%	60%	70%	70%	60%	50%
cheese	0%	20%	30%	30%	10%	0%	0%	10%	0%	**50%**
classical guitar	**90%**	50%	70%	70%	50%	80%	50%	50%	50%	**90%**
complexity	0%	50%	50%	50%	0%	**90%**	**90%**	0%	0%	80%
computational complexity	**90%**	70%	**90%**	**90%**	**90%**	**90%**	**90%**	**90%**	**90%**	**90%**
computational geometry	**100%**	40%	70%	70%	70%	**100%**	70%	70%	70%	**100%**
death penalty	**100%**	70%	90%	90%	70%	**100%**	**100%**	**100%**	**100%**	**100%**
genetic	**100%**	70%	**100%**	**100%**	**100%**	**100%**	**100%**	**100%**	**100%**	90%
geometry	**90%**	20%	**90%**	**90%**	**90%**	**90%**	**90%**	80%	**90%**	**90%**
globalization	**100%**	70%	90%	90%	**100%**	**100%**	**100%**	**100%**	**100%**	90%
gun control	0%	50%	**100%**	**100%**	**100%**	**100%**	**100%**	**100%**	**100%**	**100%**
iraq war	40%	30%	30%	30%	10%	20%	20%	10%	20%	**90%**
jaguar	0%	**30%**	0%	0%	0%	0%	0%	0%	0%	10%
jordan	0%	30%	30%	30%	40%	**100%**	**100%**	**100%**	0%	40%
moon landing	0%	30%	20%	20%	0%	0%	0%	0%	0%	**100%**
movies	10%	20%	50%	40%	50%	**70%**	**70%**	**70%**	**70%**	60%
national parks	0%	50%	10%	10%	**80%**	**80%**	**80%**	0%	0%	70%
net censorship	0%	30%	80%	80%	60%	**90%**	**90%**	**90%**	**90%**	80%
randomized algorithms	70%	**80%**	**80%**	**80%**	40%	50%	50%	50%	50%	60%
recipes	0%	20%	70%	70%	30%	90%	90%	**100%**	0%	80%
roswell	0%	20%	40%	40%	**70%**	**70%**	60%	0%	0%	60%
search engines	80%	90%	**100%**	**100%**	**100%**	**100%**	**100%**	**100%**	70%	90%
shakespeare	**100%**	70%	**100%**	**100%**	**100%**	**100%**	**100%**	**100%**	**100%**	**100%**
table tennis	90%	60%	**100%**	**100%**	90%	90%	90%	90%	90%	90%
weather	80%	50%	80%	80%	60%	80%	80%	80%	80%	**90%**
vintage cars	20%	10%	60%	60%	20%	60%	60%	20%	20%	**70%**
avg	47%	48%	61%	62%	51%	67%	64%	54%	47%	**78%**
max	**100%**	90%	**100%**	**100%**	**100%**	**100%**	**100%**	**100%**	**100%**	90%
min	0%	**10%**	0%	0%	0%	0%	0%	0%	0%	**10%**
stdev	43%	23%	31%	31%	38%	36%	36%	42%	41%	21%

Table 6.6: Relevance Ratio – Labeled case

for which no algorithm was able to produce satisfactory results.

The algorithm that emerges as the "best" is the BFS algorithm. It exhibits the best high relevance, and relevance ratio on average, and it is the algorithm that most often achieves the maximum ratio (except for the high relevance ratio in the labeled case, a record held by the MAX algorithm). Furthermore, as the low standard deviation indicates, and the study of the results reveals, it exhibits a robust and consistent behavior across queries.

At the other end of the spectrum, the HITS and the PAGERANK algorithms emerge as the "worst" algorithms, exhibiting the lowest ratios, with HITS being slightly worse. However, although they have similar (poor) performance on average, the two algorithms exhibit completely different behaviors. The behavior of the HITS algorithm is erratic. As the standard deviation indicates and the query results reveal, the performance of HITS exhibits large variance. There are queries (such as "amusement parks", "genetic", "classical guitar", "table tennis") for which the HITS algorithm exhibits good relevance and high relevance ratios, but at the same time there are many of queries (such as, "basketball", "gun control", "moon landing","recipes") for which HITS has 0% relevance ratio (even for the non-labeled case). This is related to the *Tightly Knit Community*(TKC) effect. HITS is known to favor nodes that belong to the most tightly interconnected component in the graph. The performance of the algorithm depends on how relevant this component is to the query. We discuss the TKC effect further in Section 6.3.

On the other hand PAGERANK exhibits consistent, albeit poor, performance. It is the only algorithm that never achieves 100% relevance in the labeled measure on any query, always producing at least one non-relevant result. Furthermore, the PAGERANK algorithm is qualitatively different from the rest. It is the algorithm that most often promotes documents (relevant, or not) not considered by the remaining algorithms. The strong individuality of PAGERANK becomes obvious when examining the average distance of PAGERANK to the remaining algorithms (Tables B.1, B.2, B.3, B.4), especially for the I and WI measures. This is something to be expected, since the philosophy of the algorithm is different. Furthermore, the Base Set for each query is constructed in a way that is meant to exploit the mutual reinforcing relation of hubs and authorities, a property not considered by the PAGERANK algorithm. We discuss this more in Section 6.3.

The MAX algorithm emerges as second best option after BFS, and it actually achieves the best high relevance score for the labeled measure. The quality of the MAX algorithm usually depends on the quality of the seed node, and the nodes with the highest in-degree. We actually observed that in most cases, the top-10 nodes returned by Max are a subset of the ten nodes with the highest in-degree, and the ten nodes that are most co-cited with the seed node. The ratios of Max indicate that the seed node is usually relevant to the query. This agrees with the following observations about the quality of the InDegree algorithm.

For the remaining algorithms, it is interesting to observe that the InDegree algorithm performs relatively well. On average, 3 out of 10 of the most popular documents are highly relevant, and 6 out 10 are relevant. Because of its simplicity, one would expect the quality of the InDegree algorithm to set the bar for the remaining algorithms. Surprisingly, in many queries, the InDegree algorithm outperforms some of the more sophisticated algorithms. We should note though that due to the simplicity of the algorithm, the InDegree algorithm is the one that is most affected by the choice of the search engine that it is used for generating the Base Set of Web pages. Therefore, the performance of the InDegree algorithm reflects, in part, the quality of the Google search engine, which uses, to some extent, link analysis techniques. It would be most interesting if one could generate a Base Set using a search engine that does not make use of any link analysis.

In our experiments, in most queries the Salsa algorithm produces the same top-10 pages as the InDegree algorithm. The similarity of the two algorithms becomes obvious in the average values of all similarity measures in the tables of Appendix B, and especially in the WI measure. As can be seen in Table 6.1, the graphs in our experiments contain a giant authority connected component, which attracts most of the authority weight of the Salsa algorithm. As a result, the algorithm reduces to the InDegree algorithm.

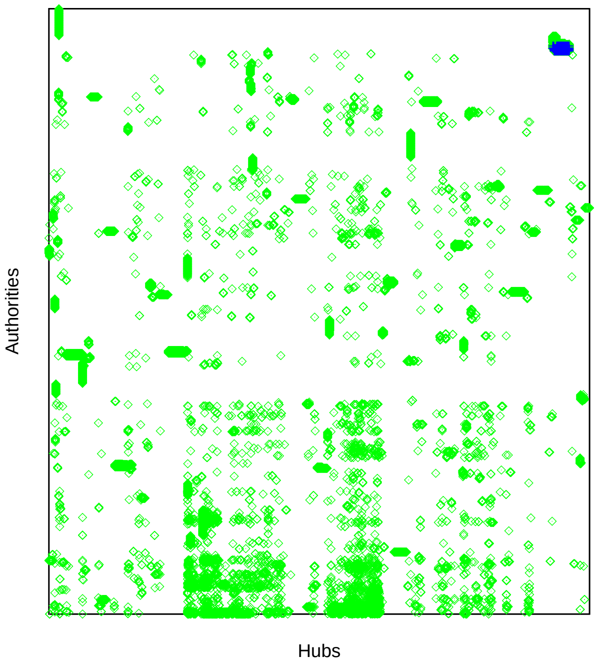
For the other variants of Hits, the HubAvg is the one that performs the worst, being only slightly better than Hits. HubAvg suffers from its own TKC effect that we describe in Section 6.3. For the AT-med, AT-avg and Norm, close examination of the results reveals that they usually have the same ratios as either the Hits or the Max algorithm. In most cases, the top-10 results of these algorithms are a subset of the union of the top-10 results of Hits and Max. Thus it is not surprising that the average performance ratios of AT-med, AT-avg and Norm take values between the ratios of Max and Hits. We also note that all derivatives of Hits (including Max) exhibit similar erratic behavior to that of Hits. This is due to the various TKC phenomena that we describe in the next section.

6.3 Community effects

It has been well documented that the HITS algorithm tends to favor the most "tightly interconnected component" of hubs and authorities in the graph G. This was first stated by Kleinberg [58] in his original paper, and Drineas et al. [29] provided some theoretical analysis to support this observation. Lempel and Moran [64] observed that the side-effect of this property of HITS is that, in a graph that contains multiple communities, the HITS algorithm will only focus on one of them in the top positions of the ranking, the one that contains the hubs and authorities that are most tightly interconnected. They termed this phenomenon the *Tightly Knit Community (TKC)* effect, and they compared the *focused* behavior of the HITS algorithm, against the *mixing* behavior of the SALSA algorithm, which tends to represent different communities in the top positions of the ranking. In this section we study similar *community effects* for all the algorithms that we consider. Our objective is to understand the kind of structures that the algorithms favor, and the effects on the rankings they produce.

The TKC effect is prominent in our experiments with the HITS algorithm, most of the time leading to a severe topic drift. Consider for example the query "gun control". Figure 6.2 shows the plot of the graph, and the top-10 results for HITS. In this query HITS gets trapped in a tightly interconnected component of 69 nodes (63 hubs and 37 authorities) which is completely disconnected from the rest of the graph, and obviously unrelated to the query. Similar phenomena appear in many of the queries that we have tested (examples include, "vintage cars", "recipes", "movies", "complexity").

Consider now the query "abortion". Figure 6.3 shows the plot of the graph for this query. The graph contains two separated, but not completely disconnected, communities of Web pages; the pro-choice community, and the pro-life community. The pro-life community contains a set X of 37 hubs from the domain `http://www.abortion-and-bible.com/`, which form a complete bipartite graph with a set Y of 288 authority nodes. These appear as a vertical strip in the top-right part of the plot. This tightly interconnected set of nodes attracts the HITS algorithm to that community, and it ranks these 37 nodes, as the best hubs. Among the 288 authorities, HITS ranks in the top-10, the authorities that are better interconnected with the remaining hubs in the community. The top-10 results for HITS, and their position in the graph are shown in Figure 6.4. The points in darker color correspond



	Hits
1.	(1.000) Coffee Club URL: www.Batavia-rof.com
2.	(0.982) Hotel and Travel URL: www.bwdriftwood.com
3.	(0.935) Basement Writers URL: www.basement-writers.com
4.	(0.935) Before Today URL: www.beforetoday.com
5.	(0.935) Bennett Boxing URL: www.bennettboxing.com
6.	(0.935) Boeing Mail URL: www.boeingmail.com
7.	(0.935) Burdan USA URL: www.burdanusa.com
8.	(0.935) British Jokes URL: www.callusforfun.com
9.	(0.917) Religious Happenings URL: www.bellbrook-umc.com
10.	(0.917) Blade Liners URL: www.bladeliners.com

Figure 6.2: The TKC effect for the Hits algorithm for the query "gun control"

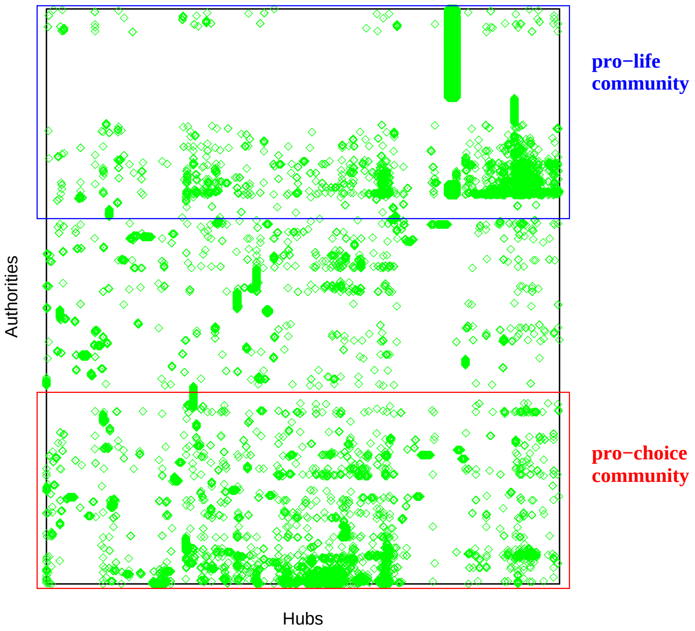
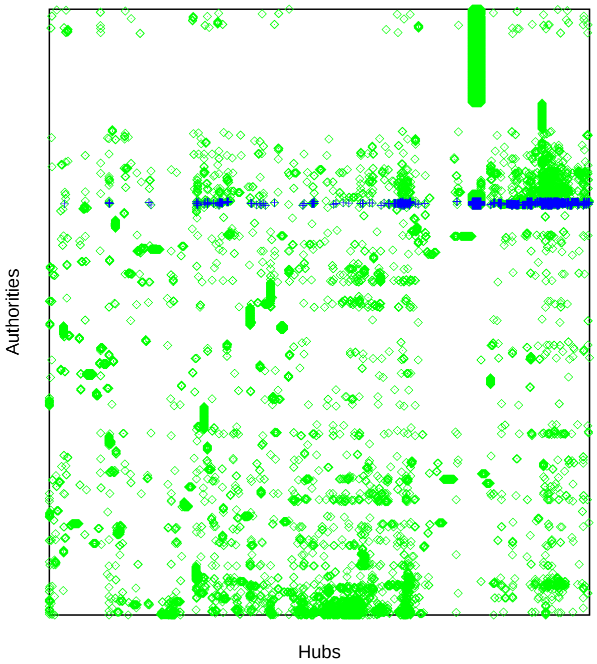


Figure 6.3: The communities of the query "abortion"



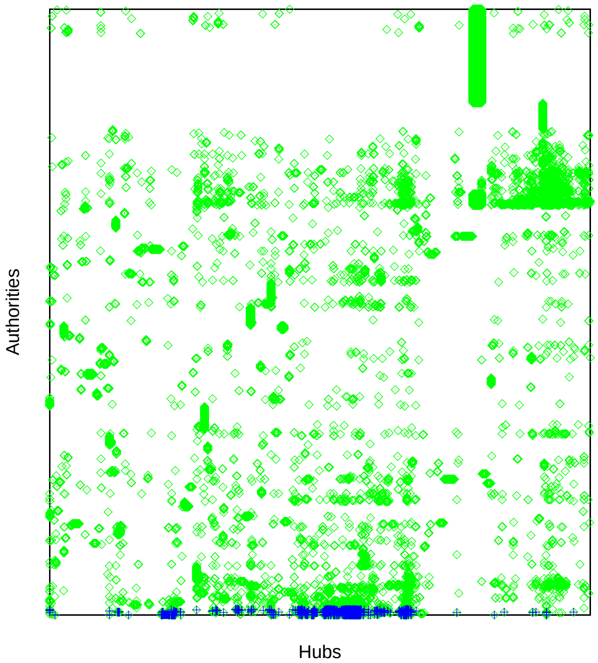
	Hits
1.	(1.000) *Priests for Life Index* *URL:www.priestsforlife.org*
2.	(0.997) *National Right to Life* *URL:www.nrlc.org*
3.	(0.994) **After Abortion: Information** **URL:www.afterabortion.org**
4.	(0.994) *ProLifeInfo.org* *URL:www.prolifeinfo.org*
5.	(0.990) **Pregnancy Centers Online** **URL:www.pregnancycenters.org**
6.	(0.989) *Human Life International* *URL:www.hli.org*
7.	(0.987) *Abortion - Breast Cancer Link* *URL:www.abortioncancer.com*
8.	(0.985) **Abortion facts and information** **URL:www.abortionfacts.com**
9.	(0.981) *Campaign Life Coalition British ...* *URL:www.clcbc.org*
10.	(0.975) Empty title field URL:www.heritagehouse76.com

Figure 6.4: The Hits algorithm for the "abortion" query

to the top-10 results of the Hits algorithm.

Consider now applying the HubAvg algorithm to the same graph. The authority nodes in the set Y are not all of equal strength. Some of them are well interconnected with other hubs in the pro-life community (the ones ranked high by Hits), but the large majority of them are pointed to only by the hubs in the set X. Recall that the HubAvg algorithm requires that the hubs point only (or at least, mainly) to good authorities. As a result, the hubs in X are penalized. The HubAvg algorithm avoids the pro-life community and focuses on the pro-choice one. Note that this is the community that contains the node with the maximum in-degree. The top-10 results of HubAvg are shown in Figure 6.5.

Note that HubAvg does not penalize densely interconnected clusters of pages. On the contrary, the HubAvg algorithm favors tightly knit communities, but it also poses the additional requirement of *exclusiveness*. That is, it requires that the hubs be *exclusive* to the community to which they belong. As a result, the HubAvg algorithm tends to favor tightly knit isolated components in the graph that contain nodes of high in-degree. These correspond to long thin horizontal strips in our plots. If such a strip exists, that is, if there exists a large set of hubs that all point to just a few authorities in the graph, then this



	HubAvg
1.	(1.000) *NARAL Pro-Choice America* *URL:www.naral.org*
2.	(0.935) *Planned Parenthood Federation* *URL:www.plannedparenthood.org*
3.	(0.921) **NAF - The Voice of Abortion Providers** **URL:www.prochoice.org**
4.	(0.625) **Abortion Clinics OnLine** **URL:www.gynpages.com**
5.	(0.516) *FEMINIST MAJORITY FOUNDATION* *URL:www.feminist.org*
6.	(0.484) *The Alan Guttmacher Institute* *URL:www.guttmacher.org*
7.	(0.439) **center for reproductive rights** **URL:www.crlp.org**
8.	(0.416) *The Religious Coalition for ...* *URL:www.rcrc.org*
9.	(0.415) *National Organization for Women* *URL:www.now.org*
10.	(0.408) *Medical Students for Choice* *URL:www.ms4c.org*

Figure 6.5: The HubAvg algorithm for the "abortion" query

community receives most of the weight of the HubAvg algorithm. Unfortunately, this case occurs often in our experiments, resulting in topic drift for HubAvg.

Figure 6.6 shows the plot of the graph for the query "recipes". The communities that attract the top-10 results of Hits and HubAvg are marked on the plot. Table 6.7 shows the top-10 tuples for each algorithm. The community on news and advertising that attracts Hits contains a set of hubs that point to nodes outside the community. This corresponds to the vertical strip above the marked rows of Hits. HubAvg escapes this community, but it assigns almost all weight to a community of just three nodes that are interconnected by 45 hubs. Only two out of these 45 hubs point to nodes other than these three nodes. Note that, in order for such a structure to attract HubAvg, the authorities (or at least some of the authorities) must have sufficiently large in-degree. In this case the top three nodes for for HubAvg correspond to the nodes with the 10th, 11th and 13th highest in-degree in the graph. This is a typical example of the behavior of the HubAvg algorithm. Although the HubAvg algorithm manages to escape the communities that pull Hits into topic drift, it still falls victim to its own TKC effect.

The influence of the various communities on the ranking of the Max algorithm is pri-

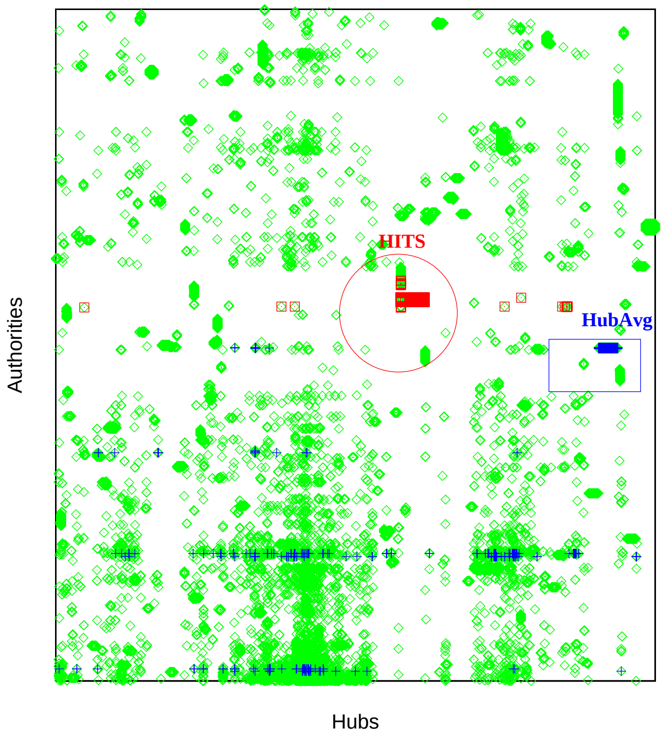


Figure 6.6: The TKC effect for the query “recipes”

	Hits
1.	(1.000) HonoluluAdvertiser.com Hawaiis URL:www.hawaiisclassifieds.com
2.	(0.999) Gannett Company, Inc. URL:www.gannett.com
3.	(0.998) AP MoneyWire URL:apmoneywire.mm.ap.org
4.	(0.990) e.thePeople : Honolulu Advertiser : URL:www.e-thepeople.com/affiliates
5.	(0.989) News From The Associated Press URL:customwire.ap.org/dynamic/fron
6.	(0.987) Honolulu Traffic Cameras, City and URL:www.co.honolulu.hi.us/cameras/
7.	(0.987) News From The Associated Press URL:customwire.ap.org/dynamic/fron
8.	(0.987) News From The Associated Press URL:customwire.ap.org/dynamic/fron
9.	(0.987) News From The Associated Press URL:customwire.ap.org/dynamic/fron
10.	(0.987) News From The Associated Press URL:customwire.ap.org/dynamic/fron

	HubAvg
1.	(1.000) Le Web des iles www.chez.com/zanozile
2.	(0.991) Please stand by.. www.sofcom.com.au
3.	(0.968) Sign in - Yahoo! Groups groups.yahoo.com/group/mauriti
4.	(0.005) Microsoft bCentral - FastCounter fastcounter.bcentral.com/fc-jo
5.	(0.004) **Recipes are Cooking at NetCooks!** **URL:www.netcooks.com**
6.	(0.004) *Mauritian cuisine, cooking and reci* *URL:ile-maurice.tripod.com*
7.	(0.004) Mauritius Australia Connection www.cjp.net
8.	(0.003) Mauritius Australia Connection www.users.bigpond.com/clancy/t
9.	(0.003) SleepAngel.com - Are you snoring yo wcpsecure.com/app/aftrack.asp?
10.	(0.002) *Chef Jobs Foodservice Culinary Inst* *URL:chef2chef.net*

Table 6.7: Hits and HubAvg for the query “recipes”

marily exerted through the seed node. The community that contains the seed node, and the co-citation of the seed node with the remaining nodes in the community determine the focus of the MAX algorithm. For example, in the "movies" query, the seed node is the Internet Movie Database[6] (IMDB), and the algorithm converges to a set of movie databases and movie reviews sites. Similarly for the "net censorship" query, where the seed node is the Electronic Frontier Foundation[7] (EFF), the algorithm outputs highly relevant results. In both these cases, the MAX algorithm manages best to distill the relevant pages from the community to which it converges. On the other hand, in the case of the "affirmative action" query the seed node is a copyright page from the University of Pennsylvania, and as a result the MAX algorithm outputs a community of university home pages.

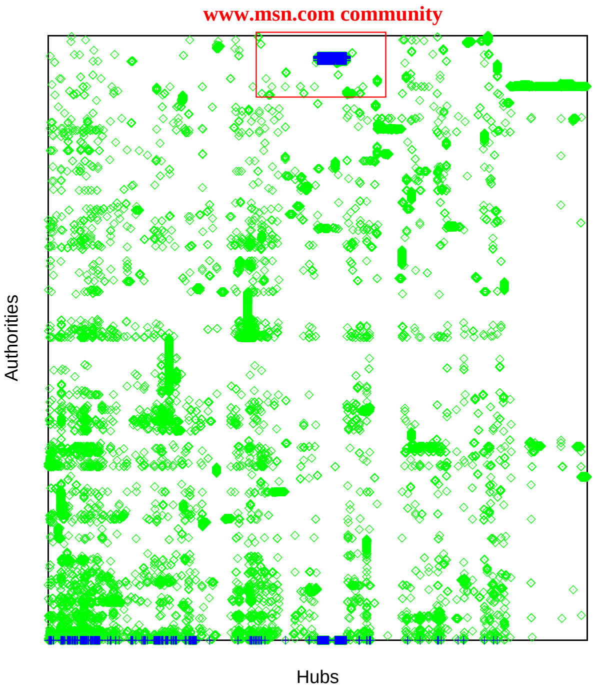
There are cases, where the seed node may belong to more than one community. Consider for example the query "basketball". The plot of the graph is shown in Figure 6.7. The seed node in this case is the NBA official Web page, `http://www.nba.com`. This page belongs to the basketball community, but it has the highest overlap with a community of nodes from `http://www.msn.com`, which causes the MAX algorithm to converge to this community.

It may also be the case that there are multiple seed nodes in the graph. For example for the "randomized algorithms" query, there are two seeds in the graph, one on algorithms, and one on computational geometry. As a result, the algorithm mixes pages of both communities.

It is also interesting to observe the behavior of MAX on the query "abortion". Figure 6.8 shows the output of the algorithm. The seed node in the graph is the "NARAL Pro-Choice" home page. Given that there is only light co-citation between the pro-choice and pro-life communities, one would expect that the algorithm would converge to pro-choice pages. However, the MAX algorithm mixes pages from both communities. The third page in the ranking of MAX is the "National Right To Life" (NRTL) home page, and there are two more in the fifth and seventh positions of the ranking. After examination of the data, we observed that the NRTL page has the second highest in-degree in the graph. Furthermore, its in-degree (189) is very close to that of the seed node (192), and, as we observed before, it belongs to a tightly interconnected community. In this case, the NRTL page acts as a *secondary* seed node for the algorithm, pulling pages from the pro-life community to the

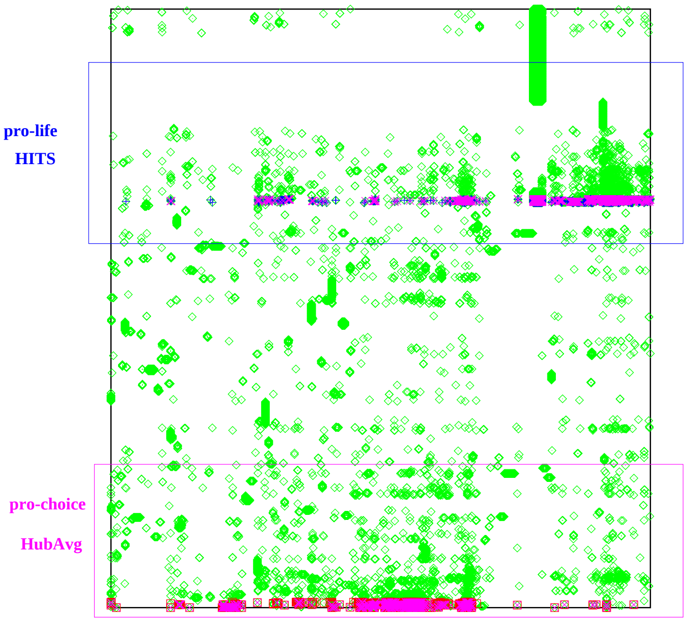
[6] `http://www.imdb.com`

[7] `http://www.eff.org`



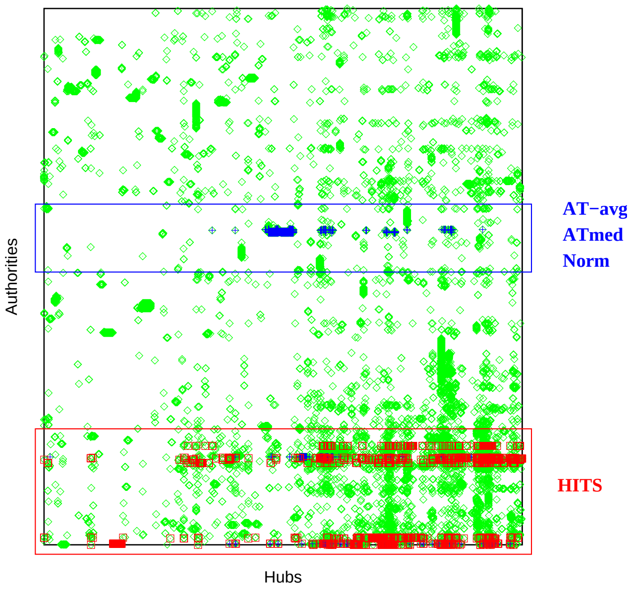
	Max
1.	(1.000) **NBA.com** **URL:www.nba.com**
2.	(0.326) Welcome to MSN.com URL:g.msn.com/0nwenus0/AK/14
3.	(0.322) Welcome to MSN.com URL:g.msn.com/0nwenus0/AK/07
4.	(0.322) Welcome to MSN.com URL:g.msn.com/0nwenus0/AK/08
5.	(0.322) Empty title field URL:g.msn.com/0nwenus0/AK/09
6.	(0.322) MSN Search – More Useful Everyday URL:g.msn.com/0nwenus0/AK/10
7.	(0.322) Welcome to MSN Shopping URL:g.msn.com/0nwenus0/AK/11
8.	(0.322) MSN Money - More Useful Everyday URL:g.msn.com/0nwenus0/AK/12
9.	(0.322) MSN People and Chat - More Useful E URL:g.msn.com/0nwenus0/AK/13
10.	(0.319) Welcome to MSN.com URL:g.msn.com/0nwenus0/AK/00

Figure 6.7: The Max algorithm for the “basketball” query



	Max
1.	(1.000) *NARAL: Pro-Coice America* *URL:www.naral.org*
2.	(0.946) *Planned Parenthood Federation* *URL:www.plannedparenthood.org*
3.	(0.918) *National Right to Life* *URL:www.nrlc.org*
4.	(0.819) **NAF - The Voice of Abortion Provide** **URL:www.prochoice.org**
5.	(0.676) *Priests for Life Index* *URL:www.priestsforlife.org*
6.	(0.624) **Pregnancy Centers Online** **URL:www.pregnancycenters.org**
7.	(0.602) *ProLifeInfo.org* *URL:www.prolifeinfo.org*
8.	(0.557) **Abortion Clinics OnLine** **URL:www.gynpages.com**
9.	(0.551) **After Abortion: Information on the** **URL:www.afterabortion.org**
10.	(0.533) *FEMINIST MAJORITY FOUNDATION* *URL:www.feminist.org*

Figure 6.8: The Max algorithm for the “abortion” query



	AT-AVG
1.	(1.000) *Hitsquad.com - Musicians Web Center* *URL:www.hitsquad.com*
2.	(0.986) Hitsquad Privacy Policy URL:www.hitsquad.com/privacy.shtml
3.	(0.986) Advertising on Hitsquad Music Indus URL:www.hitsquad.com/advertising.s
4.	(0.906) *Empty title field* *URL:www.vicnet.net.au/ easyjamn*
5.	(0.210) *AMG All Music Guide* *URL:www.allmusic.com*
6.	(0.199) Free Music Download, MP3 Music, Mus URL:ubl.com
7.	(0.179) *2000 Guitars Database* *URL:dargo.vicnet.net.au/guitar/lis*
8.	(0.135) CDNOW URL:www.cdnow.com/from=sr-767167
9.	(0.122) Guitar Alive - GuitarAlive - guitar URL:www.guitaralive.com
10.	(0.080) *OLGA - The On-Line Guitar Archive* *URL:www.olga.net*

Figure 6.9: The AT-MED,AT-AVG algorithms for the “classical guitar” query

top-10. As a result, the algorithm mixes pages from both communities.

For the AT-MED, AT-AVG and NORM algorithms, we observed that in most cases their rankings are the same as either that of HITS or that of MAX. The cases that deviate from these two reveal some interesting properties of the algorithms. Consider the query “classical guitar”. The plot of the graph and the top-10 results of the algorithms are shown in Figure 6.9. For this graph $k = 3$ for AT-MED, and $k = 5$ for AT-AVG. In this query, the AT-MED, AT-AVG and NORM algorithms allocate most of the weight to just four nodes, which are ranked lower by HITS and MAX. We discovered that these four nodes belong to a complete bipartite graph pointed to by approximately 120 hubs. It appears that the authorities in the community favored by HITS are drawn together by a set of strong hubs. As we decrease k or increase p, the effect of the strong hubs decreases, and other structures emerge as the most tightly connected ones. Surprisingly, the MAX algorithm, which corresponds to the extreme values of k and p, produces a set of top-10 results that are more similar to that of HITS than to that of AT-MED, AT-AVG and NORM. We observed the same phenomenon for the queries “cheese” and “amusement parks”.

In order to better understand the mechanics of the threshold algorithms, we consider the “roswell” query. In this case the AT-AVG produces the same top-4 authorities as HITS, but

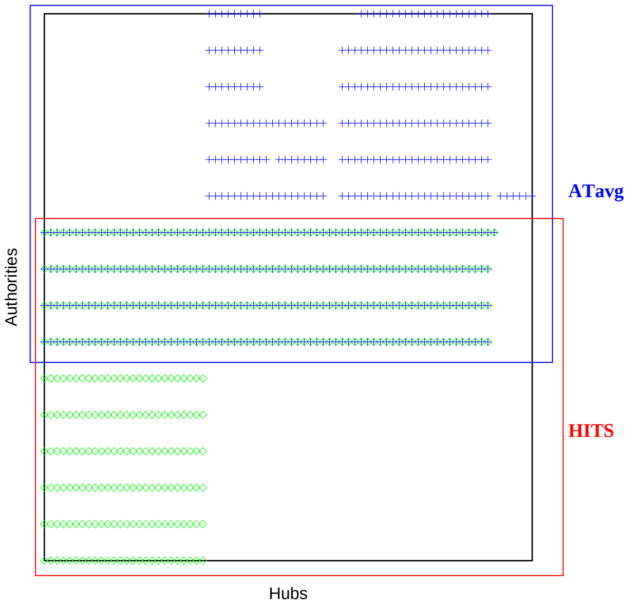


Figure 6.10: The top-10 authorities for AT-AVG,and HITS for the query "roswell"

the lower part of the ranking is completely different. In this graph $k = 4$ for the AT-AVG algorithm. Figure 6.10 shows the plot for the top-10 authorities of both algorithms. The middle part corresponds to the common authorities, the bottom rows are the authorities of HITS, and the upper rows are the authorities of AT-AVG. Note that the authorities of HITS are more tightly interconnected. However, given that $k = 4$, all hubs receive the same weight, since they all point to the middle authorities. These four authorities play a role similar to that of the seed in the MAX algorithm. Therefore, the authorities of AT-AVG are ranked higher because they have higher co-citation with the "seed" nodes, despite the fact that they are not as well interconnected. It seems that as we decrease the value of k, the importance of the in-degree increases. It is interesting that the NORM algorithm converges to the same authorities as AT-AVG, indicating that in this case for $p = 2$ the two algorithms behave similarly.

An interesting query that reveals the behavior of the algorithms in extreme settings is the "amusement parks" query. The plot of the graph for this query is shown in Figure 6.11. In this graph, the node with the maximum in-degree is a completely isolated node. As a result, both MAX and HUBAVG allocate all their weight to this node, and zero to the rest of the nodes. The HITS algorithm escapes this component easily, and converges to the most

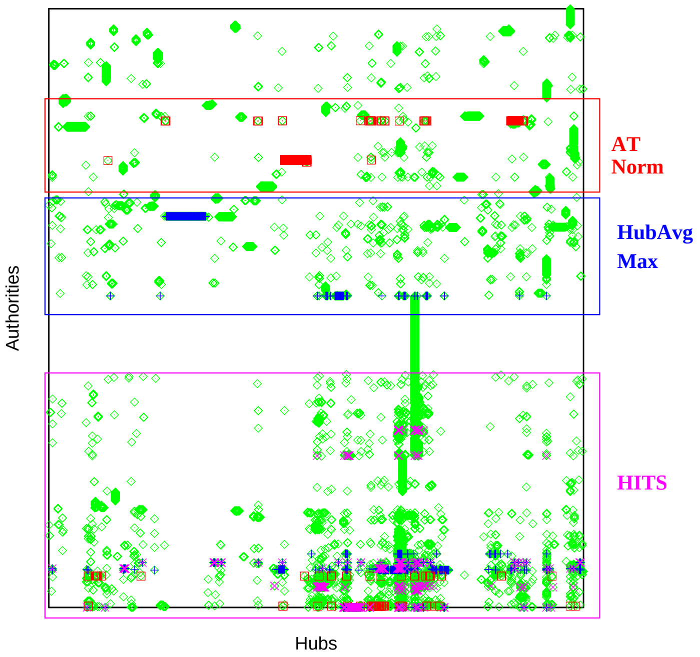


Figure 6.11: The query "amusement parks"

relevant community. The AT-MED, AT-AVG and NORM algorithms converge again to a small tight bipartite graph different from that of HITS.

The "amusement parks" query is one of the rare cases where the ranking of the SALSA algorithm differs from that of INDEGREE. SALSA is designed to handle such situations, that is, nodes that are very popular, but belong to weak communities, so the high in-degree isolated node does not appear in the top-10 of the SALSA algorithm. The premise of SALSA is interesting. It sets the weight of a node as a combination of the popularity of the community it belongs to, and its popularity within the community. However, a community in this case is defined as an authority connected component (ACC) of the graph. This is a very strong definition, since if a node shares even just one hub with the ACC, it immediately becomes part of the community. It would be interesting to experiment with SALSA on graphs that contain multiple ACCs of comparable size, and observe how the algorithm mixes between the different communities. In our experiments, all graphs contain a giant ACC, and many small ACCs that do not contribute to the ranking. Thus, the SALSA algorithm reduces to INDEGREE with the additional benefit of avoiding the occasional isolated high in-degree node for the queries "amusement parks", "automobile industries", "moon landing", "movies" and "national parks".

The effect of different communities to the InDegree algorithm is straightforward. Communities that contain nodes with high in-degree will be promoted, while communities that do not contain "popular" nodes are not represented, regardless of how tightly interconnected they are. As a result, InDegree usually mixes the results of various communities in the top-10 results. One characteristic example is the "genetic" query, where the InDegree algorithm is the only one to introduce a page from the Genetic Algorithms community. The simplistic approach of the InDegree algorithm appears to work relatively well in practice. However, it has no defense mechanism against *spurious* authorities, that is, nodes with high in-degree that are not related to the query, as in the case of the "amusement parks" query. Another such example is the "basketball" query, where the algorithm is drawn to the set of spurious authorities from `http://www.msn.com`.

The BFS algorithm counteracts the effect of spurious authorities by considering the popularity of a node in a neighborhood of larger radius. Therefore, the ranking of a node depends on how well interconnected the node is with the nodes of the community to which it belongs. Note that for the BFS algorithm when computing the weight of node i, we count the number of nodes that are reachable from node i (weighted with respect to the distance from node i). This is in contrast to the Hits algorithm, where for node i we count the number of paths that leave node i. Highly interconnected components (i.e., components with high reachability) influence the ranking, but *tightly* interconnected components (components with large number of *paths* between nodes) do not have a significant effect on BFS, since the weight of a node depends on the number of neighbors that are reachable from that node, and not on the number of paths that lead to them. As a result, the BFS algorithm avoids strong TKC effects and strong topic drift.

The existence of dense communities of hubs and authorities does not have a significant effect on the PageRank algorithm, since it does not rely on mutual reinforcement for computing authority weights. However, we observed that there are certain structures to which the PageRank algorithm is sensitive. For example, in the query "amusement parks", the PageRank algorithm assigns a large weight to the isolated node with the maximum in-degree. We observed that in general PageRank tends to favor isolated nodes of high in-degree. In this case, the hubs that point to the isolated node transfer all their weight directly to that node, since they do not point anywhere else, thus increasing its weight.

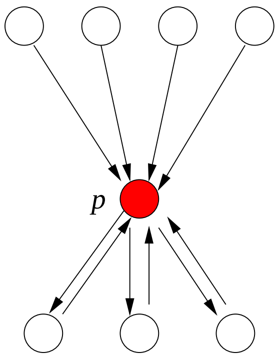


Figure 6.12: Topic drift for PageRank

Furthermore, the PageRank algorithm favors structures like the one shown in Figure 6.12. There exists a node p that is pointed to exclusively by a set of hubs (not necessarily many of them), and it creates a two link cycle with one or more nodes. In this case the node p reinforces itself, since the random walk will iterate within this cycle until it performs a random jump. This explains the fact that in certain cases the performance of the algorithm improves when we increase the jump probability. Note that such structures are very common in the Web, where p may be the entry point to some Web site, and all pages within this site point back home. They appeared very often in our experiments (even with pages that are not in the same site), and they account for most of the topic drift of the PageRank algorithm.

Overall, the PageRank algorithm appears to be mixing between different communities. This should probably be attributed to the random jumps that the algorithm performs. The random jumps are probably also responsible for the fact that the algorithm performs better than the rest of the algorithms on very sparse graphs (like in the "jaguar" and "randomized algorithms" queries).

6.4 Similarity and Stability

The definitions of similarity and stability in Chapter 5 are meant to capture the asymptotic worst case behavior of the algorithms as the size of the input graph grows. Although experiments cannot capture the asymptotic behavior of the algorithms, it is still instructive to observe how these two properties manifest themselves in practice. In this section we

	PageRank	InDegree	Salsa	HubAvg	Max	AT-med	AT-avg	Norm	BFS
d_1	1.93	1.78	1.95	2.00	2.00	2.00	2.00	2.00	1.99
d_r	0.94	0.73	0.77	0.99	0.98	0.99	0.99	0.99	0.99

(a) Distances between Hits and the remaining algorithms for the query "gun control".

	Hits	PageRank	InDegree	Salsa	AT-med	AT-avg	Norm	BFS
d_1	2.00	1.96	1.97	1.99	2.00	2.00	2.00	1.99
d_r	0.97	0.95	0.83	0.86	0.97	0.95	0.97	0.98

(b)Distances between Max and HubAvg and the remaining algorithms for the query "amusement parks".

	Hits	PageRank	InDegree	Salsa	Max	AT-med	AT-avg	Norm	BFS
d_1	0.08	1.92	1.49	1.51	1.99	1.99	0.17	0.06	1.97
d_r	0.28	0.67	0.47	0.50	0.70	0.70	0.11	0.13	0.73

(c) Distances between HubAvg and the remaining algorithms for the query "complexity".

Table 6.8: Instances of dissimilarity

examine how our theoretical results relate to the experimental observations.

In Appendix B we present the average values for the distance measures we considered in this chapter. One obvious observation is that the theoretically proved dissimilarity between Salsa and InDegree is not validated in practice. This is due to the fact that the graphs in our experiments contain a giant authority connected component. The nodes in the giant component receive almost the same weight by the two algorithms. As a result the distance between the algorithms is low.

Another interesting observation is that in our experiments we often encounter highly fragmented (although not always disconnected) graphs which are the cause for dissimilarity between the algorithms in a way very similar to that we described in the proofs of dissimilarity in Chapter 5. Consider for example the query "gun control". As discussed previously, for this query there exists a tightly interconnected component that is completely disconnected from the rest of the graph. The Hits algorithm allocates all weight to this component, while the other algorithms focus on the remainder of the graph. In particular, Max, HubAvg, AT-med, AT-avg, and Norm allocate zero weight to the component favored by Hits. Table 6.8(a) shows the d_1 and d_r distances between Hits and the remaining algorithms. We observe that in the case of the Max, HubAvg, AT-med, AT-avg, and Norm algorithms the d_1 distance is maximized.

A similar phenomenon occurs for the Max and HubAvg algorithms for the "amusement parks" query. In this case both algorithms allocate all weight to a single node, the node with the highest in-degree. Table 6.8(b) shows the distances between the Max and HubAvg algorithms and the remaining of the algorithms. Note that the d_1 distance with Hits, AT-med, and AT-avg is maximum.

The HubAvg and Max algorithms are not always close as in the case of the "amusement parks" query. For the query "complexity" the HubAvg algorithm focuses on an isolated component of a few nodes, while Max allocates the weight to other parts of the graph. Table 6.8(c) shows the distances of HubAvg to the rest of the algorithms. The d_1 distance with Max, AT-med, AT-avg, Norm, and BFS is close to the maximum. All distances, for all pairs of algorithms, all queries, and all distance measures can be found in the Web page[8] with the results.

Stability is harder to analyze experimentally, since for every graph we must identify the links that must be added or removed from the graph to cause instability. However, in the case that an algorithm allocates all weight to an isolated component, as it is the case for Hits for the "gun control" query, and for Max and HubAvg for the "amusement parks" query, we can cause the algorithm to shift its weight to a different component by removing enough links (if necessary all) from the component the algorithm favors. For example, for the "amusement parks" query, we can cause a shift in the weighting of the Max algorithm by removing just 51 links. This causes the seed node to change, and the algorithm to focus on a different component of the graph. Removing a few more links has a similar effect on the HubAvg algorithm. We can cause a similar weight shift to the Hits algorithm, however the number of links that need to be altered is significantly higher.

In general, for the algorithms that can be described as a linear dynamical systems (such as Hits, HubAvg), the stability of the algorithm depends upon the *eigengap* of the matrix that defines the dynamical system, that is, the difference between the first and the second eigenvalues. It is interesting to investigate experimentally the average value of this eigengap. This will give a good indication for the average stability of the algorithms in practice.

[8] http://www.cs.toronto.edu/∼tsap/experiments/thesis

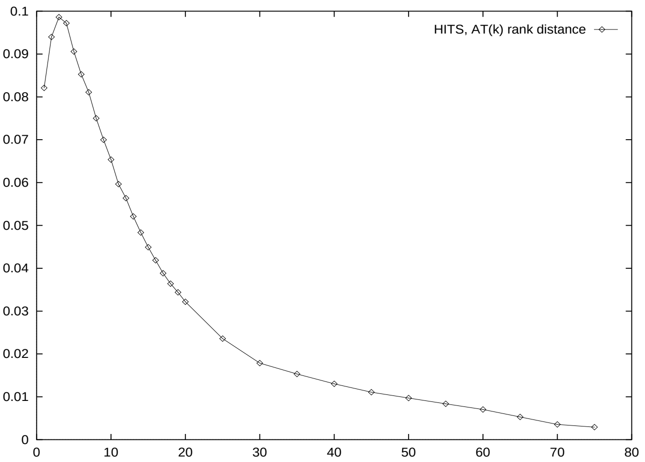
6.5 The AT(k) and Norm(p) algorithms

For the AT(k) family of algorithms, for $k = 1$ we obtain the Max algorithm, while for $k = d_{out} \leq n$ we obtain the Hits algorithm. Similarly for the Norm(p) algorithm, when we set $p = 1$, we obtain the Hits algorithm, while for $p = \infty$ we obtain the Max algorithm. An intriguing question is what happens for the intermediate values of k and p. Thus far, we cannot even guarantee that these algorithms will converge. Furthermore, although the algorithms are rather well understood for the extreme values of k and p, we do not know what kind of ranking they produce for the intermediate values.

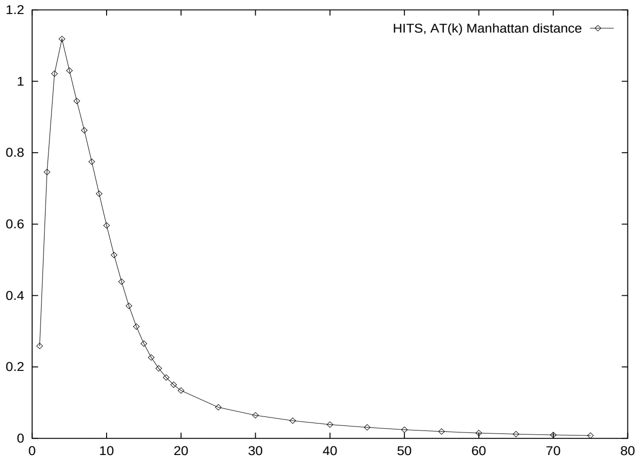
A number of initial conjectures were experimentally disproved. For example, given two nodes i and j, such that AT(k) (or Norm(p)), ranks i above j, it is *not* the case that either Max or Hits rank the pair in the same order. Furthermore, it is not the case that the top-10 results of AT(k) or Norm(p) are a subset of the union of the top-10 of Hits and Max.

We also observed that the transition between the Max and the Hits algorithm as we increase the value of k is *not* monotone. For any of the similarity/distance measures we consider, we found cases where increasing the value of k increases the distance of AT(k) from Hits. Figures 6.13(a) and 6.13(b) show how the rank, and d_1 distances change as we increase k for the query “classical guitar”. The transition is neither monotone nor smooth. In fact we were surprised to observe that in certain cases Max is more similar to Hits than AT(k) for some $k > 1$. We make similar observations for the Norm(p) algorithm. Figures 6.14(a) and 6.14(b) plot the distance of Norm(p) and Max for the query “roswell”.

Another question that remains unresolved is the relation between Norm(p) and AT(k). For the (opposite) extreme values of k and p, the two algorithms meet. A natural question is how the algorithms relate to each other for the other values of k and p, and whether there are other points where the algorithms meet, or produce similar rankings. A bold conjecture would be that for every value of k there is a value of p such that the rankings of AT(k) and Norm(p) are the same, or at least similar. We do not have yet any experimental evidence for that, although we were surprised that in certain cases the Norm(p) algorithm favored the same community as the AT-avg.

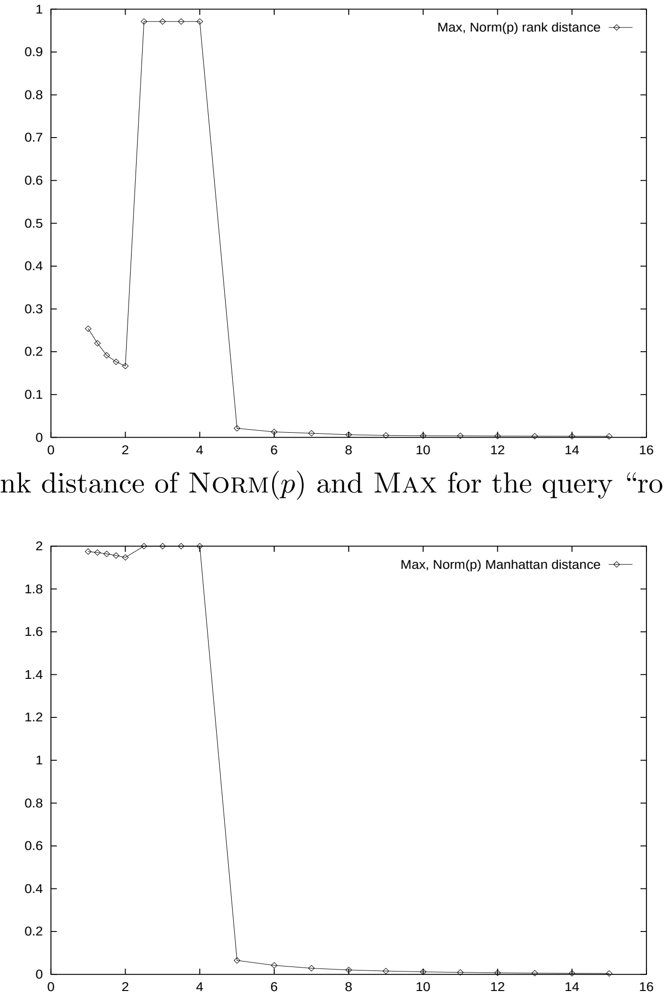


(a) Rank distance of AT(k) and HITS for the query "classical guitar"



(b) The d_1 distance of AT(k) and HITS for the query "classical guitar"

Figure 6.13: Monotonicity of the AT(k) algorithm



(a) Rank distance of $\textsc{Norm}(p)$ and $\textsc{Max}$ for the query "roswell"

(b) The d_1 distance of $\textsc{Norm}(p)$ and $\textsc{Max}$ for the query "roswell"

Figure 6.14: Monotonicity of the $\textsc{Norm}(p)$ algorithm

6.6 Application of Max algorithm to finding related pages

The property of the Max algorithm to diffuse the weight from the seed node to the remainder of the graph has the effect that the pages that are ranked highly are usually "related" to the seed node. Therefore, if we could set the seed node to some selected page, then we could use the Max algorithm to find pages *related* to that page. Finding pages related to a query Web page is a standard feature of most modern search engines. This is an active research area with a growing literature [58, 22, 45]. The current techniques use content analysis, link analysis, or a combination of both. We propose the use of the Max algorithm as a tool for discovering Web pages, related to a query Web page.

The idea of using link analysis algorithms for finding related pages was fist suggested by Kleinberg [57], and it was later extended by Dean and Henzinger [22]. In this section, we use the terminology of Dean and Henzinger [22]. First, we note that we need a different algorithm for constructing the hyperlink graph that will be given as input to the LAR algorithm. Given a query page q, Dean and Henzinger propose to construct a "vicinity graph" around q as follows. Let B denote a step that follows a link backwards, and let F denote a step that follows a link forward. Starting from the query page q, collect a set of pages that can be reached by following B, F, BF, and FB paths. The vicinity graph is the underlying hyperlink graph of this set of pages. The authors then propose to run the Hits algorithm, or other heuristics for discovering related pages.

We propose the Max algorithm as a novel alternative for discovering related pages. In order for the algorithm to work, the query page q must be the seed of the algorithm. The rest of the nodes will then be ranked according to their relation to q, where relation is defined naturally by the Max algorithm. However, it may not always be the case that the page q is the seed of the vicinity graph. In these cases, we engineer the graph, so as to make sure that the page q has the highest in-degree. We go through the nodes of the graph and find the node with the highest in-degree d. We then add enough extra "dummy" nodes in the graph, that point only to node q, so that the in-degree of q becomes greater than d. Thus the page q becomes the seed node for the Base Set. The Max algorithm will assign maximum weight 1 to page q. Following the discussion in Chapter 4, the weight will be diffused from the seed node to the remaining nodes of the graph, through the hubs. The amount of weight that reaches node i will be used as a measure of its relatedness to the seed node.

We note that link analysis by itself is not always sufficient to produce a good set of related pages. Obviously, for a page with no in and out links, the algorithm will fail. Moreover, it is important that the query page has high in-degree in the vicinity graph, even if it does not have the maximum in-degree. In this case, more weight is transferred from the seed node to the remaining nodes, and the ranking is more meaningful. In the case of a "weak" seed, we expect a secondary seed to take over. Hopefully, this secondary seed will be related to the query page, and our algorithm will still be able to produce a good set of results.

	Max		Hits		InDegree		CoCitation		PageRank		Google	
	HR	R	HR	R	HR	R	HR	R	HR	R	HR	R
www.travelocity.com	70%	100%	40%	100%	80%	100%	70%	100%	40%	50%	100%	100%
www.allmovie.com	90%	100%	90%	100%	50%	80%	90%	100%	40%	40%	100%	100%
www.cs.toronto.edu/∼bor	100%	100%	100%	100%	90%	90%	90%	90%	100%	100%	70%	100%

Table 6.9: Relevance Statistics for Related Pages queries

In order to evaluate the performance of the Max algorithms on the task of finding related pages we perform a new experimental study. We consider three different query pages: "www.travelocity.com" (an electronic travel agency – Table 6.10), "www.allmovie.com" (a site with movie information and movie reviews – Table 6.11), and "www.cs.toronto.edu/∼bor" (the home page of Allan Borodin – Table 6.12). For the construction of the vicinity graph, for the "travelocity" and "allmovie" queries, we added one dummy node, while for the "Borodin" query we needed to add 50 dummy nodes. Although the number of queries is small, we believe that the study is indicative of the trends of the algorithms. An extensive study is planned for future work.

Other than Max, we also experiment with the Hits, InDegree, and PageRank algorithms. We also consider the CoCitation heuristic which is defined as follows. Given the query page q, for each page p in the vicinity graph compute the number of hubs that point to both q and p. Then, rank the pages according to the number of hubs that they have in common with the query page. Furthermore, we also performed a comparison with the actual Google[9] search engine.

[9] http://www.google.com

For the relevance evaluation we used our (subjective) judgment, and we classified pages as highly related, related, and unrelated. In the tables with the results, the highly related pages appear boldface, and the related pages appear in italics. Table 6.9 shows the high relevance, and relevance ratios for each algorithm, for each query.

The first observation is that the link analysis algorithms perform surprisingly well. Compared to the results of Google, a search engine that uses a combination of link analysis, text analysis, and (possibly) user statistics, pure link analysis algorithms are competitive, and in the case of the "Borodin" query (Table 6.12), they outperform Google. For this query, the top 10 results of the Google search engine contain four (marginally) weakly related results, while the link analysis algorithms output results of higher quality.

The MAX algorithm performs well in all three queries. Although it reports some weakly related pages in the "travelocity"[10] (Table 6.10) and "allmovie" (Table 6.11) queries, it never reports completely unrelated pages, and it consistently retrieves highly relevant pages. The best case for the MAX algorithm is the "Borodin" query (Table 6.12), where it outputs pages on University of Toronto, a page on the book authored by Allan Borodin, pages from ex-students (Dimitris Achlioptas, Ran El Yaniv) and collaborators of Allan Borodin, as well as researchers with similar background and interests.

From the remaining link analysis algorithms PAGERANK has the worst performance. With the exception of the "Borodin" query (Table 6.12), it outputs many unrelated and weakly related pages. It appears that mutual reinforcement is a desirable property for this type of queries. This is a possible explanation for the improved performance of the HITS algorithm. We are interested in discovering dense clusters around the seed node, and these are exactly the structures that the HITS algorithm tends to favor. However, the TKC effect becomes obvious in the "travelocity" query (Table 6.10), where the algorithm favors (weakly related) home pages of search engines. Also, in the "Borodin" query, HITS focuses on a set of (related) pages of researchers in Theoretical Computer Science, missing pages like the home page of University of Toronto, or the page for the book authored by Allan Borodin.

Another interesting observation is that the INDEGREE algorithm performs surprisingly well, indicating that the Web pages with high in-degree are related to the query page. However, the limitations of the algorithm become obvious in the "allmovie" query (Ta-

[10] For the "travelocity" query, the Web pages of "Yahoo", "Excite", "HotBot", and "CNN" were deemed as weakly related since all of these pages contain numerous links about travel.

ble 6.11), where it outputs two unrelated, and many weakly related pages, indicating that the in-degree by itself is not sufficient for determining relatedness.

The CoCitation algorithm performs well on all three queries, and follows closely the Max algorithm. For the "travelocity" (Table 6.10) and "allmovie" (Table 6.11) queries, the two algorithms have 100% overlap in the top 10 results. However, the qualitative difference of the two algorithms becomes obvious in the "Borodin" query (Table 6.12). In this data set, the seed node has very little co-citation with the remaining nodes. Namely, the maximum amount of co-citation is 6, for the home page of Ran El Yaniv, while the home page of University of Toronto, and that of the Department of Computer Science, have just one hub in common with the query home page. This indicates that co-citation cannot always produce meaningful results. At the same time, in the Max algorithm, the University of Toronto home page acts as a secondary seed. The top-10 include both the the University of Toronto home page and the Department of Computer Science. It appears that the algorithm manages to combine various factors of the data set to discover related pages.

6.7 Summary and concluding Remarks

In this chapter, we performed an experimental analysis of Link Analysis Ranking. The objective was to study the effectiveness of pure Link Analysis Ranking in identifying and ranking high relevant documents to the query. We experimented with the Hits, PageRank, InDegree, Salsa, HubAvg, AT(k), Norm(p), Max, and BFS algorithms on multiple queries. The BFS and Max algorithms emerged as the best algorithms, followed by the InDegree (and Salsa) algorithms. The Hits and PageRank algorithms exhibited the lowest relevance ratios, followed by the HubAvg algorithm that performed only slightly better.

In order to better understand the performance of the algorithms, we also studied how the existence of various communities in the graphs affect the behavior of the algorithms. For each algorithm, we tried to identify the type of graph structures that the algorithm tends to favor. We discovered that the Hits algorithm tends to favor tightly knit communities of hubs and authorities, which confirmed previous experimental results and the theoretical analysis of the algorithm. The HubAvg algorithm also favors tightly knit communities, but it also imposes the additional requirement that the hubs are exclusive to the authorities in the community. As a result it favors isolated components that contain nodes with high in-degree. The PAGERANK algorithm is strongly affected by the existence of cycles, and isolated nodes of high in-degree. The MAX algorithm favors the community that contains the seed node, and the ranking of the nodes is determined to a great extent by the connectivity of the nodes with the seed node. The BFS algorithm favors popular nodes that belong to large communities that are well interconnected (but not necessarily tightly interconnected).

The experimental analysis revealed several limitations and weaknesses of Link Analysis Ranking. Overall, Link Analysis Ranking algorithms were not effective in retrieving the most relevant pages in the dataset within the top few positions of the ranking. The main reason for the shortcomings of the LAR algorithms appears to be the various community effects that we described in Section 6.3. The structures that are promoted by the various algorithms are usually not relevant to the query at hand. Tightly knit communities, favored by the HITS algorithm and variants of the HITS, such as HUBAVG, AT-MED, AT-AVG and NORM, are usually off-topic. The same is true for isolated components or nodes, favored by HUBAVG and PAGERANK, as well as for nodes that belong to cycles, which accounts to some extent for the poor behavior of the PAGERANK algorithm. On the other hand, although on average 4 out of the 10 nodes with the highest in-degree are non-relevant, nodes with high in-degree are more likely to be relevant. This explains the relatively good behavior of the INDEGREE algorithm. Furthermore, relevant high degree nodes also have a positive effect on the rankings of the MAX and BFS algorithms.

Overall, we observed that the algorithms that exhibit a "mixing" behavior, that is, they allocate the weight across different communities in the graph, tend to perform better than more "focused" algorithms, that tend to allocate all weight to the nodes of a single community. In our experiments, the graphs were often fragmented. As a result, focused algorithms often produced a lopsided weighting scheme, where almost all weight was allocated to just a few nodes. This suggests that in the future we should consider relaxations of the existing algorithms that employ more moderate approaches. One possibility is to combine multiple eigenvectors for computing the authority weights [1, 73].

Alternatively, we could consider improving the input graph so that it does not include structures that cause topic drift. Our experimental study indicates that the properties of the graph that is given as input to the LAR algorithm are critical to the quality of the output of the LAR algorithm. Little research [40, 8] has been devoted to the problem of improving the algorithm that generates the input graph. During the expansion of the Root set many non-relevant pages are introduced into the Base set. Content analysis could be applied to filter some of the noise introduced in the graph, either by pruning some of the non-relevant pages, or by downweighting their effect by adding weights to the links. Other approaches include grouping similar nodes together, so as to avoid extreme TKC phenomena, as described in the work of Roberts and Rosenthal [80]. The problem of understanding the input graph is of fundamental importance for the study of Link Analysis Ranking.

We believe that our in-depth study of the properties of the algorithms and of the effect of these properties on the rankings of the algorithms suggests a different approach for the evaluation of the LAR algorithms. An LAR algorithm is a mapping from a graph to a set of weights. Therefore, in order to evaluate an LAR algorithm we need to understand the interplay between the graph and the algorithm, as well as the connection between graph structures and topical relevance. For a fixed LAR algorithm, we need to understand how the structural properties of the graph affect the ranking of the algorithm. Then, we need to study how the relevance of the Web pages relates to these structural properties of the graph by analyzing the statistics of the graph. For example, assume that we observe that the graphs are likely to contain cycles. Then, we need to understand which algorithms are affected by the existence of cycles in the graph, and how likely it is for the nodes that belong to the cycles to be relevant to the query. Alternatively, if we know that an algorithm favors cycles, then we need to estimate how often cycles appear in the graphs, and, again, how likely it is to be relevant to the query. Performing such analysis will enable us to predict the combinations of graphs and algorithms that we expect to perform well, and work towards improving the algorithms, or the construction of the input graph. For example, we know that the performance of the Max algorithm is strongly influenced by the quality of the node with the highest in-degree. In our experiments, we observed that it is likely that the node with the highest in-degree is relevant to the query. Therefore, we expect the Max algorithm to perform well, as it is the case in practice.

The study that we performed in Section 6.3 is a first step towards such an evaluation of LAR algorithms. Ideally, we would like to be able to characterize the structures that an LAR algorithm favors within the theoretical framework we introduced in Chapter 5. Then, we would be able to argue formally about the performance of the LAR algorithms on specific families of graphs.

Although it appears that the LAR algorithms cannot always capture the true quality of the documents, it was interesting to observe that the LAR algorithms were very successful in discovering the “greater picture” behind the topic of the query. For example, for the query “computational geometry”, the algorithms returned pages on math. For the query “armstrong” we got many pages on jazz music, even if they did not contain any reference to Louis Armstrong. For the query “globalization” the algorithms returned independent media sites, anti-Bush sites, and workers’ movements sites. It is questionable whether the users would like such a wide perspective of the query (and our user study proved that they usually do not), however it is important to have a tool that can provide the “Web context” of a query. The algorithms for finding related pages to a query page build upon exactly this property of Link Analysis. Preliminary experiments indicate that LAR algorithms are successful in handling such queries. Furthermore, as a result of this generalization property of Link Analysis, LAR algorithms proved successful in finding highly related pages that do not contain the actual query words. This was the case in the “search engines” query, the “automobile industries” query, and the “globalization” query, where the algorithms discover and rank highly pages like the World Trade Organization site. Despite its limitations, Link Analysis is a useful tool for mining and understanding the Web.

	MAX	HITS	INDEGREE
1	**Travelocity.com** **www.travelocity.com**	**Travelocity.com** **www.travelocity.com**	**Travelocity.com** **www.travelocity.com**
2	**Expedia Travel** **www.expedia.com**	**Expedia Travel** **www.expedia.com**	**Expedia Travel** **www.expedia.com**
3	**MapQuest: Home** **www.mapquest.com**	**MapQuest: Home** **www.mapquest.com**	**MapQuest: Home** **www.mapquest.com**
4	**Orbitz: Airline Tickets ...** **www.orbitz.com**	*Yahoo!* *www.yahoo.com*	*Google* *www.google.com*
5	**Priceline.com** **www.priceline.com**	*Google* *www.google.com*	**Orbitz: Airline Tickets ...** **www.orbitz.com**
6	*Google* *www.google.com*	**weather.com - Index** **www.weather.com**	*Yahoo!* *www.yahoo.com*
7	*Yahoo!* *www.yahoo.com*	*CNN.com* *www.cnn.com*	**weather.com - Index** **www.weather.com**
8	w**eather.com – Index** **www.weather.com**	*Lycos* *www.lycos.com*	**Lonely Planet Thorn Tree** **thorntree.lonelyplanet.com**
9	*CNN.com* *www.cnn.com*	*My Excite* *www.excite.com*	**Amtrak - ... Train Travel** **www.amtrak.com**
10	**Trip.com** **www.trip.com**	*HotBot* *www.hotbot.com*	**Priceline.com** **www.priceline.com**

	COCITATION	PAGERANK	GOOGLE
1	**Travelocity.com** **www.travelocity.com**	First Click to the US Government firstgov.gov	**Travelocity.com** **www.travelocity.com**
2	**Expedia Travel** **www.expedia.com**	**Travelocity.com** **www.travelocity.com**	**Expedia Travel** **www.expedia.com**
3	**MapQuest: Home** **www.mapquest.com**	**Travelocity: Last minute deals** **travelocity.lmdeals.com**	**MapQuest: Home** **www.mapquest.com**
4	**Orbitz: Airline Tickets ...** **www.orbitz.com**	The IT Industry Portal www.earthweb.com	**Priceline.com** **www.priceline.com**
5	**Priceline.com** **www.priceline.com**	University of Wisconsin-Madison www.wisc.edu	**Orbitz: Airline Tickets ...** **www.orbitz.com**
6	*Yahoo!* *www.yahoo.com*	**The Industry Standard Archives** **www.thestandard.net**	**Trip.com** **www.trip.com**
7	*Google* *www.google.com*	**Travelocity** **travelocity.lmdeals.com/...**	**weather.com - Index** **www.weather.com**
8	*CNN.com* *www.cnn.com*	*nz search .. .. SearchNOW.co.nz* *searchnow.co.nz/*	**Fodor's Travel Online** **www.fodors.com**
9	w**eather.com – Index** **www.weather.com**	**U of Wisconsin-Madison Libraries** **www.library.wisc.edu**	**Lonely Planet Online** **www.lonelyplanet.com**
10	**Trip.com** **www.trip.com**	**Marriott Rewards** **www.Marriottrewards.com**	**AA.com** **www.aa.com**

Table 6.10: Related pages to "www.travelocity.com"

	Max	Hits	InDegree
1	**All Movie Guide** **www.allmovie.com**	**All Movie Guide** **www.allmovie.com**	**All Movie Guide** **www.allmovie.com**
2	**The Internet Movie Database** **www.imdb.com**	**The Internet Movie Database** **www.imdb.com**	**The Internet Movie Database** **www.imdb.com**
3	**Movie Review Query Engine** **www.mrqe.com**	**Movie Review Query Engine** **www.mrqe.com**	**Movie Review Query Engine** **www.mrqe.com**
4	**TV Guide Online - [Movies]** **www.tvguide.com/movies**	**TV Guide Online - [Movies]** **www.tvguide.com/movies**	*ABCNEWS.com: Home* *www.abcnews.com*
5	**Academy of Motion Pictures** **www.oscars.org**	**The Miramax Cafe** **www.miramax.com**	*AltaVista* *www.altavista.com*
6	*ABCNEWS.com: Home* *www.abcnews.com*	**Academy of Motion Pictures** **www.oscars.org**	*Google* *www.google.com*
7	**Real.com - Guide** **www.film.com**	**FINE LINE FEATURES** **www.flf.com**	**Academy of Motion Pictures** **www.oscars.org**
8	**TV Guide Online** **www.tvguide.com**	*ABCNEWS.com: Home* *www.abcnews.com*	**Real.com - Guide** **www.film.com**
9	**Your entertainment source** **www.hollywood.com**	**Paramount Pictures** **www.paramount.com**	**newspaper.info** **www.classifiedsatoz.com**
10	**The Miramax Cafe** **www.miramax.com**	**Bright Lights Film Journal** **www.brightlightsfilm.com**	**find.info @ Find AtoZ.com** **www.FindAtoZ.com**

	CoCitation	PageRank	Google
1	**All Movie Guide** **www.allmovie.com**	**The Internet Movie Database** **www.imdb.com**	**All Movie Guide** **www.allmovie.com**
2	**The Internet Movie Database** **www.imdb.com**	**All Movie Guide** **www.allmovie.com**	**The Internet Movie Database** **www.imdb.com**
3	**Movie Review Query Engine** **www.mrqe.com**	The Industry Desktop www.ifilmpro.com	**AMG All Music Guide** **www.allmusic.com**
4	**TV Guide Online - [Movies]** **www.tvguide.com/movies**	**The Internet Movie Guide** **www.ifilm.com**	**Movie Review Query Engine** **www.mrqe.com**
5	**Academy of Motion Pictures** **www.oscars.org**	NewspapersAtoZ.com www.classifiedsatoz.com	**Your entertainment source** **www.hollywood.com**
6	*ABCNEWS.com: Home* *www.abcnews.com*	**find.info @ Find AtoZ.com** **www.FindAtoZ.com**	**Real.com - Guide** **www.film.com**
7	**Real.com - Guide** **www.film.com**	PetsAtoZ www.PetsAtoZ.com	**Movie Reviews** **www.rottentomatoes.com**
8	**TV Guide Online** **www.tvguide.com**	TravelA-Z www.TravelA-Z.com	**Non Stop Festivals** **www.filmfestivals.com**
9	**Bright Lights Film Journal** **www.brightlightsfilm.com**	**MoviesAtoZ** **www.moviesatoz.com**	**The Internet Movie Database** **www.imdb.com/search**
10	**FINE LINE FEATURES** **www.flf.com**	RealOne Player www.real.com	**Academy of Motion Pictures** **www.oscars.org**

Table 6.11: Related pages to “www.allmovie.com”

	Max	Hits	InDegree
1	**Allan Borodin's Home Page** **www.cs.toronto.edu/∼bor**	**Allan Borodin's Home Page** **www.cs.toronto.edu/∼bor**	**Allan Borodin's Home Page** **www.cs.toronto.edu/∼bor**
2	**University of Toronto Home Page** **www.toronto.edu**	**Ran El-Yaniv's Home Page** **www.cs.technion.ac.il/∼rani**	**University of Toronto Home Page** **www.toronto.edu**
3	**Department of Computer Science** **www.cs.toronto.edu**	**Anna R. Karlin** **www.cs.washington.edu/homes/karlin**	**Department of Computer Science** **www.cs.toronto.edu**
4	**Ran El-Yaniv's Home Page** **www.cs.technion.ac.il/∼rani**	**Dimitris' Home Page** **research.microsoft.com/∼optas**	**University of Toronto Home Page** **www.utoronto.ca/uoft.html**
5	**Anna R. Karlin** **www.cs.washington.edu/homes/karlin**	**Avrim Blum's home page** **www.cs.cmu.edu/∼avrim**	**Hebrew University of Jerusalem** **www.huji.ac.il**
6	**Avrim Blum's home page** **www.cs.cmu.edu/∼avrim**	**Michael A. Bender's Homepage** **www.cs.sunysb.edu/∼bender**	**Online computation and competitive analysis** **www.cs.technion.ac.il/∼rani/book.html**
7	**Online computation and competitive analysis** **www.cs.technion.ac.il/∼rani/book.html**	**Mark Overmars homepage** **www.cs.ruu.nl/people/markov**	**Tel Aviv University** **www.tau.ac.il**
8	**Baruch Awerbuch's home page** **www.cs.jhu.edu/∼baruch**	**Bernard Chazelle's Home Page** **www.cs.princeton.edu/∼chazelle**	**Technion - Israel Institute of Technology** **www.technion.ac.il**
9	**Yossi Azar** **www.math.tau.ac.il/∼azar**	**Baruch Awerbuch's home page** **www.cs.jhu.edu/∼baruch**	**Ran El-Yaniv's Home Page** **www.cs.technion.ac.il/∼rani**
10	**Dimitris' Home Page** **research.microsoft.com/∼optas**	**Erik Demaine** **db.uwaterloo.ca/∼eddemain**	Web-Counter www.digits.com

	CoCitation	PageRank	Google
1	**Allan Borodin's Home Page** **www.cs.toronto.edu/∼bor**	**Allan Borodin's Home Page** **www.cs.toronto.edu/∼bor**	**Allan Borodin's Home Page** **www.cs.toronto.edu/∼bor**
2	**Ran El-Yaniv's Home Page** **www.cs.technion.ac.il/∼rani**	**University of Toronto Home Page** **www.toronto.edu**	**Papers and presentations** **www.users.csbsju.edu/∼cburch/pub/**
3	**Anna R. Karlin** **www.cs.washington.edu/homes/karlin**	**University of Toronto Home Page** **www.utoronto.ca/uoft.html**	ECE311S *www.control.utoronto.ca/.../ece311s.html*
4	**Avrim Blum's home page** **www.cs.cmu.edu/∼avrim**	**Online computation and competitive analysis** **www.cs.technion.ac.il/∼rani/book.html**	*3rd Year ECE Course Descriptions - Electrical* *www.ece.utoronto.ca/undergrad/courses...*
5	**Online computation and competitive analysis** **www.cs.technion.ac.il/∼rani/book.html**	**Department of Computer Science** **www.cs.toronto.edu**	*ECE310F - Linear Systems and Communications* *www.dsp.toronto.edu/∼anv/ece310f/*
6	**F. T. Leighton** **theory.lcs.mit.edu/∼ftl**	**Hebrew University of Jerusalem** **www.huji.ac.il**	**Ran El-Yaniv's Home Page** **www.cs.technion.ac.il/∼rani/**
7	WebSTAT hits.webstat.com	**Tel Aviv University** **www.tau.ac.il**	**Yossi Azar** **www.math.tau.ac.il/∼azar/**
8	**Dimitris' Home Page** **research.microsoft.com/∼optas**	**University of Toronto - Faculty of Arts and Science** **www.artsandscience.utoronto.ca**	**Carl Burch** **www.users.csbsju.edu/∼cburch/**
9	**Baruch Awerbuch's home page** **www.cs.jhu.edu/∼baruch**	**Technion - Israel Institute of Technology** **www.technion.ac.il**	**Online computation and competitive analysis** **www.cs.technion.ac.il/∼rani/book.html**
10	**Yossi Azar** **www.math.tau.ac.il/∼azar**	*UT-SFS* *www.utaps.utoronto.ca/financial_aid*	**Department of Computer Science** **www.cs.toronto.edu/**

Table 6.12: Related pages to "www.cs.toronto.edu/∼bor"

Chapter 7

Conclusions

7.1 Summary of the thesis

The rapid growth of the Web, and its increasingly wide accessibility has created the need for better and more accurate search capability. Search engines are required to produce the web pages that the users are searching for within the first pages of results. In this setting, the role of ranking becomes critical.

Link Analysis Ranking can be described as the use of hyperlink information for the purpose of ranking web documents. Link Analysis Ranking operates under the assumption that, given a collection of hyperlinked web documents, the underlying hyperlink graph contains useful information about the authority of the pages. The goal of Link Analysis Ranking is to extract this information, and use these latent authority values to rank the web documents.

In this thesis we studied the problem of Link Analysis Ranking. The objectives in this study can be summarized as follows.

1. Extend the existing techniques in order to produce new algorithms.
2. Develop a formal framework for analyzing LAR algorithms.
3. Experiment with LAR algorithms, and understand how they perform in practice.

For the first objective, we worked within the hubs and authorities paradigm, defined by Kleinberg [58]. We proposed new ways of computing hub and authority weights, namely the HubAvg and BFS algorithms (first presented in the collaborative work with A. Borodin,

G. Roberts, and J. Rosenthal [10]), the AT(k) family of algorithms (also co-created with A. Borodin, G. Roberts, and J. Rosenthal [10]), and the Norm(p) family of algorithms. A feature of the AT(k) and Norm(p) families of algorithms is that they no longer enjoy the linearity property of the previous definitions. As a result, it is harder to argue about their mathematical properties. Still, for the Max algorithm, a special case of both of these families of algorithms, we were able to prove that the algorithm converges, and we provided a rigorous characterization of the combinatorial properties of the weights it assigns. Our work has interesting connections with the study of discrete non-linear dynamical systems.

For our second objective, we performed a formal study of Link Analysis Ranking. We introduced a theoretical framework that allows us to define specific properties of LAR algorithms, such as monotonicity, distance, similarity, stability, and locality. The objective of this theoretical framework is to provide the means for analyzing and comparing algorithms, and define properties that characterize their behavior. Some of these properties, such as stability, appear to be desirable. For other properties, such as label independence, it is not obvious if and when they are desirable. However, they are important since they characterize the algorithm. We were thus able to provide an axiomatic characterization of the InDegree algorithm. We proved that any algorithm that is monotone, label independent, and local produces the same ranking as the InDegree algorithm. This result indicates that our framework and the properties we defined are both meaningful and useful.

For the third objective, we experimented with the algorithms that we proposed, as well as some of the existing ones. We performed an extensive experimental study on multiple queries, using user feedback. The objective of the study was to determine the quality of the algorithms, but also to understand how the theoretically predicted properties of the algorithms affect their ranking behavior in practice. We observed that some of these properties (for example, the TKC effect for the Hits algorithm) were indeed prominent in our experiments. We were surprised to discover that some of the "simpler" algorithms, such as InDegree and BFS, appear to perform better than more sophisticated algorithms, such as PageRank. We also examined an interesting application of the Max algorithm for finding related pages with promising results.

One interesting observation that emerged from the experimental study was that many of the LAR algorithms (Hits, HubAvg, AT(k), Norm(p)) act as clustering algorithms, that is, they tend to promote a certain cluster of nodes in the graph. As a result, when the clusters are not related to the query at hand they fall victims to topic drift. Therefore, for an LAR algorithm, it is important to identify the structures that it promotes, and understand if such structures are expected to be related to the query. For example, the structures promoted by HITS and HUBAVG are most often non-relevant, while the ones favored by the MAX algorithm are usually relevant to the query at hand. Furthermore, our study poses the problem of improving the algorithm for constructing the Base Set, so that some of the noise is eliminated and the structures promoted by the LAR algorithms become related to the query at hand.

Finally, in the process of defining distance measures between rankings, we provided generalizations of distance measures between total orderings to the case of partial orderings. Our work extends the results of Fagin et al. [34] on comparing top-k lists to the case of rankings.

7.2 Future Work

In this section we discuss some interesting possible future research directions opened in our research.

7.2.1 Further extensions of hubs and authorities

Extending the definitions of hubs and authorities can be carried further. One possible extension that we plan to consider is to define a good hub as a node that has *paths* to many good authorities, and a good authority as a node that is reachable from many good hubs. Note that the authority value a_i that the PAGERANK algorithm assigns to node i captures the number of paths that end up at node i. This suggests the following interesting combination of the HITS and PAGERANK algorithms.Run the PAGERANK algorithm with a uniform “jump” probability distribution to obtain a set of authority weights. Now, invert the links of the graph G, and run the PAGERANK algorithm again, this time using the authority weights as the “jump” probability distribution. This will produce a set of hub weights that will be used as the “jump” probability distribution for the next execution of the PAGERANK algorithm on the graph G (not inverted). Iterate this process until, hopefully, it converges.

Furthermore, the idea of treating weights preferentially can be extended further. The threshold operation and the L_p norm are just one possibility for enforcing this idea. We could explore other possibilities. For example, we could apply some predetermined weighting scheme on the weights, or we could set a threshold to the weight, instead of setting a threshold to the number of weights we retain. Another possible generalization of hubs and authorities, similar in spirit to the work of Roberts and Rosenthal [80], is to cluster together nodes and apply Link Analysis Ranking on the graph of clusters. Then, the ranking algorithm, instead of returning a ranked list of pages, will return a ranked list of clusters.

7.2.2 Study of non-linear dynamical systems

In Chapter 4 we proved that the Max algorithm converges. The Max algorithm is a special case of Norm(p) and AT(k), for $p = \infty$, and $k = 1$ respectively. An intriguing question that remains to be resolved is the convergence of the Norm(p) and AT(k) algorithms, for the other values of p and k. Given that the algorithms converge for the two extreme values of p and k, it seems reasonable that the algorithms will converge for the intermediate values as well. However, the experimental results indicate that the algorithms do not make a monotone transition from Hits to Max as we as we vary p and k. Therefore, their behavior may be far less predictable than what we expect. It is an interesting research problem to understand the properties of the algorithms for these values.

7.2.3 Theoretical analysis of LAR algorithms

We consider our theoretical framework as a first step towards the formal analysis of LAR algorithms. There are plenty of research questions that emerge within this framework. First, it would be interesting to consider other distance measures and understand how they relate to the existing ones. For example, if we view the weight vectors as distributions, then we may apply information theoretic distance measures (such as the *Jensen-Shannon* divergence [68]). Furthermore, it would be interesting to define other properties for LAR algorithms. One possible property is *spam sensitivity.* Spam sensitivity captures how susceptible a ranking algorithm is against a malicious node, or a coalition of malicious nodes that want to artificially boost their ranking. This notion is closely related to stability, only in this setting we are interested in the change to the ranking of a single node, rather than to the whole ranking. Another notion that is interesting to define is that of *focus.* Given a graph that has some underlying structure with multiple clusters, we would like to be able to argue about the way the algorithm distributes the weight across clusters.

We are also interested in investigating further the stability and similarity of LAR algorithms. Although most of the stability and similarity results were negative, it is possible that, if we restrict ourselves to smaller classes of graphs, we can obtain positive results. We are interested in pursuing the following research issues.

- Consider a class of graphs $\overline{\mathcal{G}}_n$, and assume that there is a probability distribution over the graphs in $\overline{\mathcal{G}}_n$. Extend the definition of stability and similarity to capture similarity and stability in expectation (or with high probability).

- Consider the class of graphs $\mathcal{G}^\delta$ that consists of the set of graphs whose adjacency matrix has a large eigengap between the first and the second singular values. The results of Ng, Zheng and Jordan [72] suggest that the Hits algorithm is stable on this class. Formulate their results in our framework, and, if possible, prove that Hits is stable on $\overline{\mathcal{G}}_n$ if and only if $\overline{\mathcal{G}}_n \subseteq \mathcal{G}^\delta$.

- Consider the class of random graphs that are generated as follows. Every node i has a predefined authority and hub value, a_i and h_i respectively, and a link is generated from i to j with probability proportional to $h_i a_j$. We refer to this class of graphs as *product* graphs, $\mathcal{G}^P$. This is a model that has received some limited attention in the literature [1]. Combining the results of Achlioptas et al. [1] and Ng, Zheng, and Jordan [72], it seems promising to prove that Hits is stable (in expectation) on $\mathcal{G}^P$. The results of Achlioptas et al. [1] suggest that the graphs in $\mathcal{G}^P$ are expected to have large eigengap, and the Hits algorithm is expected to produce authority weights that are close to the predefined authority values. Note that the expected authority weights of the InDegree algorithm are also these predefined authority values. Therefore, it seems possible that Hits and InDegree algorithms are similar (in expectation) on the class $\mathcal{G}^P$.

Another possible research direction for obtaining some positive results for rank similarity between LAR algorithms is to consider weaker notions of similarity. Rather than classifying two algorithms as rank similar, or rank dissimilar, we could instead try to characterize the degree of (dis)similarity of the two algorithms. For two algorithms $\mathcal{A}_1, \mathcal{A}_2$, we can define the *similarity coefficient* [59] as follows.

$$\lim_{n\to\infty} \sup \max_{G\in\overline{\mathcal{G}}_n} d^{(1)}(\mathcal{A}_1(G), \mathcal{A}_2(G))$$

This is the maximum fraction of pairs of nodes that are ranked in a different order by the two algorithms. When this is $o(1)$ the two algorithms are rank similar. If the coefficient is constant the algorithms are rank dissimilar. An interesting problem is to identify the degree of dissimilarity of the two algorithms. That is, identify how close to 1 the similarity coefficient can be, in which case the two algorithms produce two completely reversed rankings.

Finally, the formal study of ranking algorithms may be the forerunner for the study of other types of algorithms. An interesting candidate is clustering where it seems reasonable to define notions of similarity and stability for clustering algorithms.

Bibliography

[1] D. Achlioptas, A. Fiat, A. Karlin, and F. McSherry. Web search through hub synthesis. In *Proceedings of the 42nd Foundation of Computer Science (FOCS 2001)*, Las Vegas, Nevada, 2001.

[2] D. Achlioptas and F. McSherry. Fast computation of low rank matrix approximations. In *ACM Symposium on Theory of Computing (STOC)*, 2001.

[3] P. Andritsos, P. Tsaparas, R. Miller, and K. Sevcik. LIMBO: Scalable clustering of categorical data. In *International Conference on Extending DataBase Technology (EDBT)*, Heraklion, Crete, Greece, 2004.

[4] Y. Azar, A. Fiat, A. Karlin, F. McSherry, and J. Saia. Spectral analysis of data. In *Proceedings of the 33rd Symposium on Theory of Computing (STOC 2001)*, Hersonissos, Crete, Greece, 2001.

[5] R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval.* Addison-Wesley-Longman, 1999.

[6] M. W. Berry and S. T. Dumais. Using linear algebra for intelligent Information Retrieval. *SIAM Review*, 37(4):573–595, 1995.

[7] G. Bhalotia, C. Nakhe, A. Hulgeri, S. Chakrabarti, and S. Sudarshan. Keyword searching and browsing in databases using BANKS. In *ICDE*, 2002.

[8] K. Bharat and M. R. Henzinger. Improved algorithms for topic distillation in a hyperlinked environment. In *Research and Development in Information Retrieval*, pages 104–111, 1998.

[9] K. Bharat and G. A. Mihaila. When experts agree: Using non-affiliated experts to rank popular topics. In *Proceedings of the 10th International World Wide Web Conference*, 2001.

[10] A. Borodin, G. O. Roberts, J. S. Rosenthal, and P. Tsaparas. Finding authorities and hubs from link structures on the World Wide Web. In *Proceedings of the 10th International World Wide Web Conference*, Hong Kong, 2001.

[11] A. Borodin, G. O. Roberts, J. S. Rosenthal, and P. Tsaparas. Link analysis ranking: Algorithms, experiments, and theory. *ACM Transactions on Internet Technology*, 2003.

[12] R. Botafogo, E. Rivlin, and B. Shneiderman. Structural analysis of hypertext: Identifying hierarchies and useful metrics. *ACM Transactions on Information Systems*, 10:142 – 180, 1992.

[13] S. Brin and L. Page. The anatomy of a large-scale hypertextual Web search engine. In *Proceedings of the 7th International World Wide Web Conference*, Brisbane, Australia, 1998.

[14] A. Broder. Web searching technology overview. In *Advanced school and Workshop on Models and Algorithms for the World Wide Web*, Udine, Italy, 2002.

[15] J. Carrière and R. Kazman. WebQuery: Searching and visualizing the Web through connectivity. In *Proceedings of the 6th International World Wide Web Conference*, 1997.

[16] S. Chakrabarti, B. Dom, D. Gibson, J. Kleinberg, P. Raghavan, and S. Rajagopalan. Automatic resource compilation by analysing hyperlink structure and associated text. In *Proceedings of the 7th International World Wide Web Conference*, 1998.

[17] S. Chakrabarti, B. E. Dom, and P. Indyk. Enhanced hypertext categorization using hyperlinks. In *Proceedings of SIGMOD, ACM International Conference on Management of Data*, Seattle, US, 1998.

[18] S. Chien, C. Dwork, R. Kumar, D. Simon, and D. Sivakumar. Towards exploiting link evolution. In *Workshop on Algorithms for the Web*, Vancuver, 2002.

[19] D. Cohn and H. Chang. Learning to probabilistically identify authoritative documents. In *Proceedings of the 17th International Conference on Machine Learning*, pages 167–174, Stanford University, 2000.

[20] N. Craswell and D. Hawking. Overview of the TREC-11 Web track. In *Proceedings of the Eleventh Text Retrieval Conference (TREC-11)*, 2002.

[21] B. Davison. Recognizing nepotistic links on the web. In *AAAI-2000 Workshop on Artificial Intelligence for Web Search*, Austin Texas, 2000. AAAI Press.

[22] J. Dean and M. R. Henzinger. Finding related pages in the world wide web. In *Proceedings of the Eighth International World-Wide Web Conference (WWW9)*, 1999.

[23] S. Deerwerster, S. T. Dumais, G. W. Furnas, T. K. Landaur, and Richard Harshman. Indexing by Latent Semantic Analysis. *Journal of American Society for Information Science*, 41(6):391–407, 1990.

[24] A. Dempster, N. Laird, and D. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 39:1–38, 1977.

[25] R. L. Devaney. *An Introduction to Chaotic Dynamical Systems.* W. Benjamin, New York, 1986.

[26] P. Diaconis and R. Graham. Spearman's footrule as a measure of disarray. *Journal of the Royal Statistical Society*, 39(2):262 – 268, 1977.

[27] P. Doreian. Measuring the relative standing of disciplinary journals. *Information Processing and Management*, 24:45–56, 1988.

[28] P. Doreian. A measure for standing for citation networks within a wider environment. *Information Processing and Management*, 30:21–31, 1994.

[29] P. Drineas, A. Frieze, R. Kannan, S. Vempala, and V. Vinay. Clustering in large graphs and matrices. In *ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 1999.

[30] C. Dwork, R. Kumar, M. Naor, and D. Sivakumar. Rank aggregation methods for the Web. In *Proceedings of the 10th International World Wide Web Conference*, Hong Kong, 2001.

[31] L. Eggh and R. Rousseau. *Introduction to Infometrics.* Elsevier, 1990.

[32] R. Fagin, R. Kumar, M. Mahdian, D. Sivakumar, and E. Vee. Comparing and aggregating rankings with ties, 2003. To appear.

[33] R. Fagin, R. Kumar, K. McCurley, J. Novak, D. Sivakumar, J. A. Tomlin, and D. P. Williamson. Searching the workplace web. In *Proceedings of the 12th International World Wide Wed Conference (WWW2003)*, Budapest, 2003.

[34] R. Fagin, R. Kumar, and D. Sivakumar. Comparing top k lists. In *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2003.

[35] G. Flake, S. Lawrence, and C. L. Giles. Efficient identification of Web communities. In *Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Boston, MA, 2000.

[36] G. W. Flake, S. Lawrence, C. L. Giles, and F. Coetzee. Self-organization and identification of Web communities. *IEEE Computer*, 35(3), 2002.

[37] M. E. Frisse. Searching for information in a hypertext medical book. *Communications of ACM*, 31(7):880–886, 1988.

[38] N. Geller. On the citation influence methodology of Pinski and Narin. *Information Processing and Management*, 14:93–95, 1978.

[39] D. Gibson, J. Kleinberg, and P. Raghavan. Clustering categorical data: An approach based on dynamical systems. In *Proceedings of the 24th Intl. Conference on Very Large Databases (VLDB)*, 1998.

[40] D. Gibson, J. Kleinberg, and P. Raghavan. Inferring web communities from link topology. In *Proceedings of 9th ACM Conference on Hypertext and Hypermedia*, 1998.

[41] G. H. Golub and C. F. Van Loan. *Matrix Computations,* 2nd ed. Johns Hopkins University Press, Baltimore, 1989.

[42] E. Grafield. Citation analysis as a tool in journal evaluation. *Science*, 178:471–479, 1972.

[43] L. Guo, F. Shao, C. Botev, and J. Shanmugasundaram. XRANK: Ranked keyword search over XML documents. In *SIGMOD*, San Diego, CA, USA, 2003.

[44] T. H. Haveliwala. Topic sensitive page rank. In *Proceedings of the 11th International Word Wide Web Conference (WWW 2002)*, Hawai, 2002.

[45] T. H. Haveliwala, A. Gionis, D. Klein, and P. Indyk. Similarity search on the Web: Evaluation and scalability considerations. In *Proceedings of the 28th International Conference on Very Large Data Bases (VLDB)*, 2002.

[46] D. Hawking, N. Craswell, P. Thistlewaite, and D. Harman. Results and challenges in Web seach evaluation. In *Proceedings of the Eighth International World Wide Web Conference*, 1999.

[47] D. Hawking, E. Voorhes, N. Craswell, and P. Bailey. Overview of the TREC-8 Web track. In *Proceedings of the Eighth Text Retrieval Conference (TREC-8)*, 1999.

[48] Thomas Hofmann. Probabilistic Latent Semantic Analysis. In *Proc. of Uncertainty in Artificial Intelligence, UAI'99*, Stockholm, 1999.

[49] Thomas Hofmann. Learning probabilistic models of the web. In *Proceedings of the 23rd International Conference on Research and Development in Information Retrieval (ACM SIGIR'00)*, 2000.

[50] Richard A. Holmgren. *A First Course in Discrete Dynamical Systems.* Springer-Verlag, Berlin, Germany, 1994.

[51] C. H. Hubbell. An input-output approach to clique identification. *Sociometry*, 28:377–399, 1965.

[52] B. J. Jansen, A. Spink, J. Bateman, and T. Saracevic. Real life Information Retrieval: A study of user queries on the Web. *ACM SIGIR Forum*, 32:5–17, 1998.

[53] G. Jeh and J. Widom. Scaling personalized Web search. In *Proceedings of the 12th International World Wide Wed Conference(WWW2003)*, Budapest, 2003.

[54] R. Kannan, S. Vempala, and A. Vetta. On clusterings: Good, bad and spectral. In *Proceedings of the 41st Foundation of Computer Science (FOCS 2000)*, Redondo Beach, 2000.

[55] L. Katz. A new status index derived from sociometric analysis. *Psychometrika*, 18:39–43, 1953.

[56] M. G. Kendall. *Rank Correlation Methods.* Griffin, London, 1970.

[57] J. Kleinberg. Authoritative sources in a hyperlinked environment. In *Proceedings of the Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 668–677, 1998.

[58] J. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of ACM (JASM)*, 46, 1999.

[59] Jon Kleinberg. Personal communication, 2003.

[60] F. Korn, H. V. Jagadish, and C. Faloutsos. Efficiently supporting ad hoc queries in large datasets of time sequences. In *ACM SIGMOD*, Tuscon, Arizona, 1997.

[61] R. Kumar, P. Raghavan, S. Rajagopalan, and A. Tomkins. Trawling the Web for emerging cyber-communities. In *Proceedings of the 8th International World Wide Web Conference (WWW 1999)*, 1999.

[62] H. C. Lee. Metasearch using the co-citation graph. In *Proceedings of Internet Computing (IC2003)*, 2003.

[63] H. C. Lee and A. Borodin. Perturbation of the hyperlinked environment. In *Proceedings of the Ninth International Computing and Combinatorics Conference*, 2003.

[64] R. Lempel and S. Moran. The stochastic approach for link-structure analysis (SALSA) and the TKC effect. In *Proceedings of the 9th International World Wide Web Conference*, May 2000.

[65] R. Lempel and S. Moran. Rank stability and rank similarity of web link-based ranking algorithms. Technical Report CS-2001-22, Technion - Israel Institute of Technology, 2001.

[66] R. Lempel and A. Soffer. PicASHOW: Pictorial authority search by hyperlinks on the Web. In *Proceedings of the 10th International World Wide Web Conference (WWW 2002)*, Hong Kong, 2001.

[67] L. Li, Y. Shang, and W. Zhang. Improvement of HITS-based algorithms on Web documents. In *Proceedings of the 11th International Word Wide Web Conference (WWW 2002)*, Hawai, 2002.

[68] J. Lin. Divergence measures based on the Shannon entropy. *Machine Learning*, 37(1): 145–151, 1991.

[69] M. Marchiori. The quest for correct information on Web: Hyper search engines. In *Proceedings of the 6th International World Wide Web Conference*, 1997.

[70] A. Mendelzon and D. Rafiei. What do the neighbours think? Computing Web page reputations. *IEEE Data Engineering Bulletin*, 23(3):9–16, 2000.

[71] A. Y. Ng, M. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems (NIPS)*, 2002.

[72] A. Y. Ng, A. X. Zheng, and M. I. Jordan. Link analysis, eigenvectors, and stability. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, Seattle, Washington, USA, 2001.

[73] A. Y. Ng, A. X. Zheng, and M. I. Jordan. Stable algorithms for link analysis. In *Proceedings of the 24th International Conference on Research and Development in Information Retrieval (SIGIR 2001)*, New York, 2001.

[74] 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.

[75] C. H. Papadimitriou, H. Tamaki, P. Raghavan, and S. Vempala. Latent Semantic Indexing: A probabilistic analysis. In *17th Annual Symposium on Principles of Database Systems*, Seattle, 1998.

[76] G. Pinski and F. Narin. Citation influence for journal aggreagates of scientific publications: Theory with applications to the literature of physics. *Information Processing and Management*, 12:297–312, 1976.

[77] V. Poosala and Y. E. Ioannidis. Selectivity estimation without the attribute value independence assumption. In *Very Large DataBase (VLDB) Conference*, 1997.

[78] D. Rafiei and A. Mendelzon. What is this page known for? Computing web page reputations. In *Proceedings of the 9th International World Wide Web Conference*, Amsterdam, Netherlands, 2000.

[79] M. Richardson and P. Domingos. The intelligent surfer: Probabilistic combination of link and content information in PageRank. In *Advances in Neural Information Processing Systems (NIPS) 14*, 2002.

[80] G. O. Roberts and J. S. Rosenthal. Downweighting tightly knit communities in World Wide Web rankings. Submitted for publication, 2003.

[81] J. T. Sandefur. *Discrete dynamical systems.* Oxford: Clarendon Press, 1990.

[82] J. Savoy and J. Picard. Report on the TREC-8 experiment: Searching on the Web and in distributed collections. In *Proceedings of the Eighth Text Retrieval Conference (TREC-8)*, Gaithersburg, Maryland, 1999.

[83] J. Savoy and Y. Rasolofo. Report on the TREC-9 experiment: Link-based retrieval and distributed collections. In *Proceedings of the Ninth Text Retrieval Conference (TREC-9)*, Gaithersburg, Maryland, 2000.

[84] C. Silverstein, M. Henzinger, H. Marais, and M. Moricz. Analysis of a very large AltaVista query log. Technical Report 1998-014, Digital SRC, 1998.

[85] A. Singhal and M. Kaszkiel. A case study in web search using TREC algorithms. In *Proceedings of the Tenth International World Wide Web Conference*, 2001.

[86] N. Slonim and N. Tishby. Agglomerative Information Bottleneck. In *Advances in Neural Information Processing Systems (NIPS)*, Breckenridge, CO, 1999.

[87] G. Strang. *Linear Algebra.* Academic Press, 1980.

[88] N. Tishby, F. C. Pereira, and W. Bialek. The Information Bottleneck method. In *37th Annual Allerton Conference on Communication, Control and Computing*, Urban-Champaign, IL, 1999.

[89] J. A. Tomlin. An entropy approach to unitrusive targeted advertising on the web. In *Proceedings of the 9th International World Wide Wed Conference (WWW2000)*, Amsterdam, 2000.

[90] J. A. Tomlin. A new paradigm for ranking pages on the World Wide Web. In *Proceedings of the 12th International World Wide Wed Conference (WWW2003)*, Budapest, 2003.

[91] M. Turk and A. Pentland. Eigenfaces and recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.

Appendix A

Supplementary material for Chapter 5

A.1 Distance metrics and near metrics

Let D be a domain of elements, and let $d : D \times D \to \mathbb{R}$ be a distance function defined over set D. The function d is a *metric* if it satisfies the following conditions.

1. For all $x, y \in D$, $d(x, y) = 0$ if and only if $x = y$ (reflexivity).

2. For all $x, y \in D$, $d(x, y) = d(y, x)$ (symmetry).

3. For all $x, y, z \in D$, $d(x, y) \leq d(x, z) + d(z, y)$ (triangle inequality).

The following definitions are taken from the work of Fagin, Kumar and Sivakumar [34].

Definition A.1 (Relaxed polygonal inequality) *For some $c > 0$, a distance function $d : D \times D \to \mathbb{R}$ satisfies the* c-polygonal inequality*, if for all $k > 0$ and all $x, y_1, \ldots, y_k, z \in D$, $d(x, y) \leq c(d(x, y_1) + d(y_1, y_2) + \cdots d(y_k, z))$.*

Definition A.2 (Near Metric) *A distance measure $d : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ is a* near metric *if it is reflexive and symmetric, and there is a constant $c > 0$ independent of n, such that the distance measure d satisfies the c-polygonal inequality.*

Definition A.3 (Equivalent distance measures) *Two distance measures d and d' between n-dimensional vectors are equivalent if there exist constants c_1 and c_2 independent of n, such that $c_1 d'(\boldsymbol{w}_1, \boldsymbol{w}_2) \leq d(\boldsymbol{w}_1, \boldsymbol{w}_2) \leq c_2 d'(\boldsymbol{w}_1, \boldsymbol{w}_2)$, for all vectors $\boldsymbol{w}_1, \boldsymbol{w}_2$.*

We now examine which of the distance measures we defined in Section 5.4 are metrics, or near metrics.

A.1.1 The d_1 distance measure

We will prove that the d_1 distance measure is a near metric over the set of L_1 unit vectors, by proving that it is equivalent to the L_1 distance between n-dimensional vectors.

Lemma A.1 *Let $\boldsymbol{w}_1$ and $\boldsymbol{w}_2$ be two unit vectors under the L_1 norm. We have that*

$$d_1(\boldsymbol{w}_1, \boldsymbol{w}_2) \leq \|\boldsymbol{w}_1 - \boldsymbol{w}_2\|_1 \leq 2d_1(\boldsymbol{w}_1, \boldsymbol{w}_2)$$

Proof: For the following we use $\|\cdot\|$ to denote the L_1 norm. Obviously, $d_1(\boldsymbol{w}_1, \boldsymbol{w}_2) = \min_{\gamma_1,\gamma_2 \geq 1} \|\gamma_1 \boldsymbol{w}_1 - \gamma_2 \boldsymbol{w}_2\| \leq \|\boldsymbol{w}_1 - \boldsymbol{w}_2\|$. We will now prove that $\|\boldsymbol{w}_1 - \boldsymbol{w}_2\|_1 \leq 2d_1(\boldsymbol{w}_1, \boldsymbol{w}_2)$. We first prove that at least one of γ_1, γ_2 has to be equal to 1. Assume that this is not true. Then, let $\gamma_1^*, \gamma_2^* = \arg\min_{\gamma_1,\gamma_2 \geq 1} \|\gamma_1 \boldsymbol{w}_1, \gamma_2 \boldsymbol{w}_2\|$, where $\gamma_1^*, \gamma_2^* > 1$. Without loss of generality assume that $\gamma_1^* \geq \gamma_2^*$. We have that

$$\|\gamma_1^* \boldsymbol{w}_1, \gamma_2^* \boldsymbol{w}_2\| = \sum_{i=1}^{n} |\gamma_1^* w_1(i) - \gamma_2^* w_2(i)| = \gamma_2^* \sum_{i=1}^{n} |\frac{\gamma_1^*}{\gamma_2^*} w_1(i) - w_2(i)| > \sum_{i=1}^{n} |\frac{\gamma_1^*}{\gamma_2^*} w_1(i) - w_2(i)| \,,$$

where the last inequality follows from the fact that $\gamma_2^* > 1$. Now let $\gamma_1 = \frac{\gamma_1^*}{\gamma_2^*}$ and $\gamma_2 = 1$. We have that $\|\gamma_1 \boldsymbol{w}_1, \gamma_2 \boldsymbol{w}_2\| < \|\gamma_1^* \boldsymbol{w}_1, \gamma_2^* \boldsymbol{w}_2\|$, thus reaching a contradiction. Therefore, at least one of γ_1^*, γ_2^* is equal to 1. Note that this is true irrespective of the fact that the vectors are L_1 unit vectors.

Now, without loss of generality assume that $w_1(i) \geq w_2(i)$ for all $i = 1, \ldots, k$, and that $w_1(i) \leq w_2(i)$, for all $i = k+1, \ldots, n$, for some k. Also let $\sum_{i=1}^{k} w_1(i) = X$ and $\sum_{i=1}^{k} w_2(i) = Y$. Since $\boldsymbol{w}_1$ and $\boldsymbol{w}_2$ are L_1 unit vectors, we have that $\sum_{i=k+1}^{n} w_1(i) = 1 - X$ and $\sum_{i=k+1}^{n} w_2(i) = 1 - Y$. We have that

$$\begin{aligned} \|\boldsymbol{w}_1 - \boldsymbol{w}_2\| &= \sum_{i=1}^{n} |w_1(i) - w_2(i)| \\ &= \sum_{i=1}^{k} (w_1(i) - w_2(i)) + \sum_{i=k+1}^{n} (w_2(i) - w_1(i)) \\ &= \sum_{i=1}^{k} w_1(i) - \sum_{i=1}^{k} w_2(i) + \sum_{i=k+1}^{n} w_2(i) - \sum_{i=k+1}^{n} w_1(i) \end{aligned}$$

$$
\begin{aligned}
&= X - Y + (1 - Y) - (1 - X) \\
&= 2(X - Y)
\end{aligned}
$$

Assume now that $\gamma_2 = 1$, and let $\gamma = \arg\min_{\gamma \geq 1} \|\gamma \boldsymbol{w}_1, \boldsymbol{w}_2\|$. Then

$$
\begin{aligned}
d_1(\boldsymbol{w}_1, \boldsymbol{w}_2) &= \sum_{i=1}^{n} |\gamma w_1(i) - w_2(i)| \\
&= \sum_{i=1}^{k} |\gamma w_1(i) - w_2(i)| + \sum_{i=k+1}^{n} |\gamma w_1(i) - w_2(i)| \\
&\geq \sum_{i=1}^{k} |\gamma w_1(i) - w_2(i)| \\
&= \sum_{i=1}^{k} (\gamma w_1(i) - w_2(i)) \\
&= \gamma \sum_{i=1}^{k} w_1(i) - \sum_{i=1}^{k} w_2(i) = \gamma X - Y \\
&\geq X - Y
\end{aligned}
$$

The proof works similarly if we assume that $\gamma_1 = 1$. In this case it is easy to see that $d_1(w_1, w_2) \geq \gamma(1-Y) - (1-X)$, and the proof follows. Therefore, $\|\boldsymbol{w}_1 - \boldsymbol{w}_2\| \leq 2 d_1(\boldsymbol{w}_1, \boldsymbol{w}_2)$. □

A.1.2 Rank distance measures

It is obvious that $d_r^{(0)} = K^{(0)}$ is not a metric, or a near metric, since all vectors have distance zero from the uniform vector, that assigns the same weight to all nodes. The Hausdorff distance measure is well known to be a metric. For $K^{(p)}$ we prove the following theorem.

Lemma A.2 *For $p \geq 1/2$ the $K^{(p)}$ rank distance satisfies the triangle inequality. For $p < 1/2$, $K^{(p)}$ distance measure does not satisfy the triangle inequality.*

Proof: Let $\boldsymbol{a}_1, \boldsymbol{a}_2$, and $\boldsymbol{a}_3$ denote three authority weight vectors. Let $\mathcal{X}_1 = \mathcal{X}(\boldsymbol{a}_1, \boldsymbol{a}_2)$, $\mathcal{X}_2 = \mathcal{X}(\boldsymbol{a}_2, \boldsymbol{a}_3)$ and $\mathcal{X}_3 = \mathcal{X}(\boldsymbol{a}_1, \boldsymbol{a}3)$. We use similar indices for the sets $\mathcal{E}, \mathcal{U}, \mathcal{Y}$, and $\mathcal{Z}$.

First, we will prove that if $p < 1/2$ then $K^{(p)}$ does not satisfy the triangle inequality. Consider two vectors $\boldsymbol{a}_1$ and $\boldsymbol{a}_3$ that assign distinct weights to all nodes, and produce the reverse rankings. Then, we have that $K^{(p)}(\boldsymbol{a}_1, \boldsymbol{a}_3) = n(n-1)/2$. Consider now a vector $\boldsymbol{a}_2$

that assigns the same weight to all nodes. Then, we have that $K^{(p)}(\boldsymbol{a}_1, \boldsymbol{a}_2) = pn(n-1)/2$, and $K^{(p)}(\boldsymbol{a}_2, \boldsymbol{a}_3) = pn(n-1)/2$. If $p < 1/2$ then $K^{(p)}(\boldsymbol{a}_1, \boldsymbol{a}_3) > K^{(p)}(\boldsymbol{a}_1, \boldsymbol{a}_2) + K^{(p)}(\boldsymbol{a}_2, \boldsymbol{a}_3)$.

Assume now that $p \geq 1/2$. We have that $K^{(p)}(\boldsymbol{a}_1, \boldsymbol{a}_3) = |\mathcal{X}_3| + p(|\mathcal{Y}_3| + |Z_3|)$. For all $\{i,j\} \in \mathcal{Y}_3$, $a_1(i) = a_1(j)$ and $a_3(i) \neq a_3(j)$, and $\mathcal{I}^{(p)}_{\boldsymbol{a}_1\boldsymbol{a}_3}(i,j) = p$. If $a_2(i) = a_2(j)$ then $\{i,j\} \in \mathcal{Y}_2$, and $\mathcal{I}^{(p)}_{\boldsymbol{a}_2\boldsymbol{a}_3}(i,j) = p$. If $a_2(i) \neq a_2(j)$ then $\{i,j\} \in \mathcal{Y}_1$, and $\mathcal{I}^{(p)}_{\boldsymbol{a}_1\boldsymbol{a}_2}(i,j) = p$. Thus, in all cases, $\mathcal{I}^{(p)}_{\boldsymbol{a}_1\boldsymbol{a}_2}(i,j) + \mathcal{I}^{(p)}_{\boldsymbol{a}_2\boldsymbol{a}_3}(i,j) \geq \mathcal{I}^{(p)}_{\boldsymbol{a}_1\boldsymbol{a}_3}(i,j)$.

For all $\{i,j\} \in \mathcal{Z}_3$, $a_1(i) \neq a_1(j)$ and $a_3(i) = a_3(j)$, and $\mathcal{I}^{(p)}_{\boldsymbol{a}_1\boldsymbol{a}_3}(i,j) = p$. If $a_2(i) \neq a_2(j)$ then $\{i,j\} \in \mathcal{Z}_2$, and $\mathcal{I}^{(p)}_{\boldsymbol{a}_2\boldsymbol{a}_3}(i,j) = p$. If $a_2(i) = a_2(j)$ then $\{i,j\} \in \mathcal{Z}_1$, and $\mathcal{I}^{(p)}_{\boldsymbol{a}_1\boldsymbol{a}_2}(i,j) = p$. Thus, in all cases, $\mathcal{I}^{(p)}_{\boldsymbol{a}_1\boldsymbol{a}_2}(i,j) + \mathcal{I}^{(p)}_{\boldsymbol{a}_2\boldsymbol{a}_3}(i,j) \geq \mathcal{I}^{(p)}_{\boldsymbol{a}_1\boldsymbol{a}_3}(i,j)$.

For all $\{i,j\} \in \mathcal{X}_3$, without loss of generality assume that $a_1(i) < a_1(j)$ and $a_3(i) > a_3(j)$. The other case is treated symmetrically. For all $\{i,j\} \in \mathcal{X}_3$ we have that $\mathcal{I}^{(p)}_{\boldsymbol{a}_1\boldsymbol{a}_3}(i,j) = 1$. If $a_2(i) \neq a_2(j)$ then $\{i,j\} \in \mathcal{X}_1 \cup \mathcal{X}_2$, and $\mathcal{I}^{(p)}_{\boldsymbol{a}_1\boldsymbol{a}_2}(i,j) + \mathcal{I}^{(p)}_{\boldsymbol{a}_2\boldsymbol{a}_3}(i,j) = 1$. If $a_2(i) = a_2(j)$ then $\{i,j\} \in \mathcal{Z}_1 \cup \mathcal{Y}_2$, and $\mathcal{I}^{(p)}_{\boldsymbol{a}_1\boldsymbol{a}_2}(i,j) + \mathcal{I}^{(p)}_{\boldsymbol{a}_2\boldsymbol{a}_3}(i,j) = 2p$. Since $p \geq 1/2$, it follows that $\mathcal{I}^{(p)}_{\boldsymbol{a}_1\boldsymbol{a}_2}(i,j) + \mathcal{I}^{(p)}_{\boldsymbol{a}_2\boldsymbol{a}_3}(i,j) \geq I_{\boldsymbol{a}_1\boldsymbol{a}_3}(i,j)$.

Therefore, $K^{(p)}(\boldsymbol{a}_1, \boldsymbol{a}_2) + K^{(p)}(\boldsymbol{a}_2, \boldsymbol{a}_3) \geq K^{(p)}(\boldsymbol{a}_1, \boldsymbol{a}_3)$, so $K^{(p)}$ satisfies the triangle inequality. □

We now prove the following theorem.

Theorem A.1 *The $K^{(p)}$ distance measure is a metric for all $p \geq 1/2$.*

Proof: In order to prove that $K^{(p)}$ satisfies the reflexive property we need to redefine the domain of $K^{(p)}$. Obviously, if $K^{(p)}$ is defined over $\mathbb{R} \times \mathbb{R}$, there are vectors that are not equal, yet they produce the same ranking. Every vector defines a partial ordering of the nodes in P. Let $\mathcal{O}_P$ denote the set of all possible partial orderings of the elements in P. A partial ordering can be represented as a DAG, or the topological sort of the elements of the DAG. We define the $K^{(p)}$ over $\mathcal{O}_P \times \mathcal{O}_P$. For two partial orderings $O_1, O_2 \in \mathcal{O}_P$, $K^{(p)}(O_1, O_2) = 0$ if and only if $O_1 = O_2$. □

For $p < 1/2$, $K^{(p)}$ is obviously not a metric, since it does not satisfy the triangle inequality. However, we can prove the following theorem.

Theorem A.2 *For $p < 1/2$, such that $p = \Theta(1)$, $K^{(p)}$ is a near metric.*

Proof: From Theorem 5.3, we have that for any vectors $\boldsymbol{w}_1, \boldsymbol{w}_2$,

$$K^{(p)}(\boldsymbol{w}_1, \boldsymbol{w}_2) \leq K^{(1/2)}(\boldsymbol{w}_1, \boldsymbol{w}_2) \leq \frac{1}{2p} K^{(p)}(\boldsymbol{w}_1, \boldsymbol{w}_2) \ .$$

Therefore, for $p = \Theta(1)$, $K^{(p)}$ is equivalent to $K^{(1/2)}$. Since $K^{(1/2)}$ is a metric, $K^{(p)}$ is a near metric. □

A.2 Norms and vectors

In this section we prove a useful lemma relating different norms. We also compute the maximum L_q-distance between any two L_p-unit vectors. We first prove the following auxiliary lemma.

Lemma A.3 *Let $\boldsymbol{v}$ be a vector of length n, and suppose $1 \leq p < q \leq \infty$. Then $\|\boldsymbol{v}\|_p \leq \|\boldsymbol{v}\|_q n^{1/p-1/q}$.*

Proof: Assume first that $s < \infty$. We use Hölder's inequality, which states that for any p and q such that $1 < r, s < \infty$ and $1/r + 1/s = 1$, if $\boldsymbol{x}$ and $\boldsymbol{y}$ are two n-dimensional vectors, then

$$\sum_{i=1}^{n} |x_i y_i| \leq \left(\sum_{i=1}^{n} |x_i|^r \right)^{1/r} \left(\sum_{i=1}^{n} |y_i|^s \right)^{1/s} \ .$$

Set $r = q/p$ and $s = 1/(1-1/r)$. Also, set $x_i = v_i^r$ and $y_i \equiv 1$, and let $\|\boldsymbol{v}\|_p^p$ denote $(\|\boldsymbol{v}\|_p)^p$, and $\|\boldsymbol{v}\|_q^p$ denote $(\|\boldsymbol{v}\|_q)^p$. We have that

$$\begin{aligned}
\|\boldsymbol{v}\|_p^p &= \sum_{i=1}^{n} |v_i|^p = \sum_{i=1}^{n} |v_i|^p \, 1 \\
&\leq \left(\sum_{i=1}^{n} (|v_i|^p)^r \right)^{1/r} \left(\sum_{i=1}^{n} 1^s \right)^{1/s} \\
&= \left(\sum_{i=1}^{n} |v_i|^{p(q/p)} \right)^{1/(q/p)} n^{1/s} \\
&= \|\boldsymbol{v}\|_q^p n^{1-1/(q/p)} = \|\boldsymbol{v}\|_q^p n^{1-p/q} \ .
\end{aligned}$$

Taking p-th roots of both sides, we obtain $\|\boldsymbol{v}\|_p \leq \|\boldsymbol{v}\|_q n^{1/p-1/q}$, as claimed.

For the case $q = \infty$, we compute that

$$\|\boldsymbol{v}\|_p^p = \sum_{i=1}^{n} |v_i|^p \leq \sum_{i=1}^{n} \max_i |v_i|^p = n \max_i |v_i|^p = n\|\boldsymbol{v}\|_\infty^p \; .$$

Thus, $\|\boldsymbol{v}\|_p \leq n^{1/p}\|\boldsymbol{v}\|_\infty$. □

Lemma A.4 *The maximum L_q-distance d_q, between two L_p-unit n-dimensional vectors is $\Theta(n^{1/q-1/p})$, if $q \leq p$, and $\Theta(1)$ if $q > p$.*

Proof: Let $\boldsymbol{w}_1, \boldsymbol{w}_2$ be two n-dimensional L_p-unit vectors.

$$\begin{aligned} d_q(\boldsymbol{w}_1, \boldsymbol{w}_2) &= \min_{\gamma_1,\gamma_2} \|\gamma_1\boldsymbol{w}_1 - \gamma_2\boldsymbol{w}_2\|_q \leq \|\boldsymbol{w}_1 - \boldsymbol{w}_2\|_q \\ &\leq \|\boldsymbol{w}_1\|_q + \|\boldsymbol{w}_2\|_q \end{aligned}$$

The last inequality follows from the triangle inequality. From Lemma A.3, we have that if $q \leq p$, then $\|\boldsymbol{w}_1\|_q \leq n^{1/q-1/p}$. Thus, $d_q(\boldsymbol{w}_1, \boldsymbol{w}_2) = O(n^{1/q-1/p})$. This upper bound is achieved if $\boldsymbol{w}_1$ and $\boldsymbol{w}_2$ are uniform vectors with opposing values.

If $q > p$ then $\|\boldsymbol{w}_1\|_q \leq \|\boldsymbol{w}_1\|_p = 1$. Therefore, $d_q(\boldsymbol{w}_1, \boldsymbol{w}_2) \leq 2$. This upper bound is achieved if the vectors are two different standard basis vectors. □

Appendix B

Similarity measures

In this appendix we present the average values for all the similarity measures that we consider.

	Hits	PageRank	InDegree	Salsa	HubAvg	Max	AT-med	AT-avg	Norm	BFS
Hits	10.0	1.1	4.1	4.1	3.4	4.3	3.9	5.2	6.6	2.8
PageRank	1.1	10.0	3.2	3.1	2.2	2.1	2.1	1.8	1.6	2.2
InDegree	4.1	3.2	10.0	9.8	5.2	6.3	6.2	6.0	5.2	5.6
Salsa	4.1	3.1	9.8	10.0	5.1	6.3	6.2	5.9	5.2	5.6
HubAvg	3.4	2.2	5.2	5.1	10.0	5.5	6.2	6.5	5.6	3.3
Max	4.3	2.1	6.3	6.3	5.5	10.0	8.7	6.7	6.3	5.5
AT-med	3.9	2.1	6.2	6.2	6.2	8.7	10.0	7.5	6.8	5.0
AT-avg	5.2	1.8	6.0	5.9	6.5	6.7	7.5	10.0	8.3	4.1
Norm	6.6	1.6	5.2	5.2	5.6	6.3	6.8	8.3	10.0	3.4
BFS	2.8	2.2	5.6	5.6	3.3	5.5	5.0	4.1	3.4	10.0

Table B.1: Average Intersections over top-10

	Hits	PageRank	InDegree	Salsa	HubAvg	Max	AT-med	AT-avg	Norm	BFS
Hits	10.0	1.0	3.7	3.6	3.3	4.2	3.7	5.2	6.6	2.9
PageRank	1.0	10.0	2.8	2.7	2.2	2.1	2.0	1.5	1.4	1.6
InDegree	3.7	2.8	10.0	9.7	5.1	7.2	6.8	5.9	5.2	5.6
Salsa	3.6	2.7	9.7	10.0	5.0	7.1	6.9	5.8	5.2	5.6
HubAvg	3.3	2.2	5.1	5.0	10.0	5.2	5.9	6.2	5.4	2.7
Max	4.2	2.1	7.2	7.1	5.2	10.0	8.8	6.6	6.3	5.1
AT-med	3.7	2.0	6.8	6.9	5.9	8.8	10.0	7.4	6.7	4.7
AT-avg	5.2	1.5	5.9	5.8	6.2	6.6	7.4	10.0	8.2	4.0
Norm	6.6	1.4	5.2	5.2	5.4	6.3	6.7	8.2	10.0	3.5
BFS	2.9	1.6	5.6	5.6	2.7	5.1	4.7	4.0	3.5	10.0

Table B.2: Average Weighted Intersections over top-10

	Hits	PageRank	InDegree	Salsa	HubAvg	Max	AT-med	AT-avg	Norm	BFS
Hits	0.00	0.53	0.42	0.45	0.34	0.24	0.20	0.16	0.14	0.24
PageRank	0.53	0.00	0.32	0.30	0.49	0.45	0.44	0.46	0.46	0.46
InDegree	0.42	0.32	0.00	0.08	0.42	0.36	0.36	0.37	0.37	0.39
Salsa	0.45	0.30	0.08	0.00	0.46	0.40	0.40	0.41	0.41	0.42
HubAvg	0.34	0.49	0.42	0.46	0.00	0.25	0.27	0.25	0.26	0.38
Max	0.24	0.45	0.36	0.40	0.25	0.00	0.07	0.13	0.13	0.19
AT-med	0.20	0.44	0.36	0.40	0.27	0.07	0.00	0.08	0.08	0.16
AT-avg	0.16	0.46	0.37	0.41	0.25	0.13	0.08	0.00	0.04	0.19
Norm	0.14	0.46	0.37	0.41	0.26	0.13	0.08	0.04	0.00	0.20
BFS	0.24	0.46	0.39	0.42	0.38	0.19	0.16	0.19	0.20	0.00

Table B.3: Average Rank distances

	Hits	PageRank	InDegree	Salsa	HubAvg	Max	AT-med	AT-avg	Norm	BFS
Hits	0.00	1.64	1.22	1.25	1.32	1.11	1.12	0.84	0.61	1.46
PageRank	1.64	0.00	0.94	0.93	1.64	1.38	1.38	1.49	1.54	1.13
InDegree	1.22	0.94	0.00	0.11	1.42	0.86	0.87	1.01	1.09	0.96
Salsa	1.25	0.93	0.11	0.00	1.43	0.89	0.89	1.03	1.11	0.96
HubAvg	1.32	1.64	1.42	1.43	0.00	1.07	1.06	0.97	1.02	1.67
Max	1.11	1.38	0.86	0.89	1.07	0.00	0.21	0.57	0.65	1.19
AT-med	1.12	1.38	0.87	0.89	1.06	0.21	0.00	0.42	0.55	1.20
AT-avg	0.84	1.49	1.01	1.03	0.97	0.57	0.42	0.00	0.27	1.35
Norm	0.61	1.54	1.09	1.11	1.02	0.65	0.55	0.27	0.00	1.42
BFS	1.46	1.13	0.96	0.96	1.67	1.19	1.20	1.35	1.42	0.00

Table B.4: Average d_1 distances

Appendix C

Experiments

Hits	PageRank	InDegree
1. (1.000) *Priests for Life Index* *URL:www.priestsforlife.org*	1. (1.000) *WCLA Feedback* *URL:www.janeylee.com/wcla*	1. (1.000) *prochoiceamerica.org : NARAL* *URL:www.naral.org*
2. (0.997) *National Right to Life* *URL:www.nrlc.org*	2. (0.911) *Planned Parenthood Action Net* *URL:www.ppaction.org/ppaction/prof*	2. (0.984) *National Right to Life* *URL:www.nrlc.org*
3. (0.994) **After Abortion: Information o** **URL:www.afterabortion.org**	3. (0.837) *Welcome to the Westchester Coalit* *URL:www.wcla.org*	3. (0.969) *Planned Parenthood Federation of* *URL:www.plannedparenthood.org*
4. (0.994) *ProLifeInfo.org* *URL:www.prolifeinfo.org*	4. (0.714) *Planned Parenthood Federation of* *URL:www.plannedparenthood.org*	4. (0.865) **NAF - The Voice of Abortion** **URL:www.prochoice.org**
5. (0.990) **Pregnancy Centers Online** **URL:www.pregnancycenters.org**	5. (0.633) GeneTree.com Page Not Found URL:www.qksrv.net/click-1248625-91	5. (0.823) *Priests for Life Index* *URL:www.priestsforlife.org*
6. (0.989) *Human Life International* *URL:www.hli.org*	6. (0.630) Bible.com Prayer Room URL:www.bibleprayerroom.com	6. (0.807) **Pregnancy Centers Online** **URL:www.pregnancycenters.org**
7. (0.987) *Abortion - Breast Cancer Link –* *URL:www.abortioncancer.com*	7. (0.609) *United States Department of Healt* *URL:www.dhhs.gov*	7. (0.740) *ProLifeInfo.org* *URL:www.prolifeinfo.org*
8. (0.985) **Abortion facts and informatio** **URL:www.abortionfacts.com**	8. (0.538) **Pregnancy Centers Online** **URL:www.pregnancycenters.org**	8. (0.734) **After Abortion: Information o** **URL:www.afterabortion.org**
9. (0.981) *Campaign Life Coalition British C* *URL:www.clcbc.org*	9. (0.517) Bible.com Online World URL:bible.com	9. (0.672) **Abortion Clinics OnLine** **URL:www.gynpages.com**
10. (0.975) Empty title field URL:www.heritagehouse76.com	10. (0.516) *National Organization for Women* *URL:www.now.org*	10. (0.625) *Abortion - Breast Cancer Link –* *URL:www.abortioncancer.com*

HubAvg	Max	AT-med
1. (1.000) *prochoiceamerica.org : NARAL* *URL:www.naral.org*	1. (1.000) *prochoiceamerica.org : NARAL* *URL:www.naral.org*	1. (1.000) *prochoiceamerica.org : NARAL* *URL:www.naral.org*
2. (0.935) *Planned Parenthood Federation of* *URL:www.plannedparenthood.org*	2. (0.946) *Planned Parenthood Federation of* *URL:www.plannedparenthood.org*	2. (0.933) *Planned Parenthood Federation of* *URL:www.plannedparenthood.org*
3. (0.921) **NAF - The Voice of Abortion** **URL:www.prochoice.org**	3. (0.918) *National Right to Life* *URL:www.nrlc.org*	3. (0.837) **NAF - The Voice of Abortion** **URL:www.prochoice.org**
4. (0.625) **Abortion Clinics OnLine** **URL:www.gynpages.com**	4. (0.819) **NAF - The Voice of Abortion** **URL:www.prochoice.org**	4. (0.717) *National Right to Life* *URL:www.nrlc.org*
5. (0.516) *FEMINIST MAJORITY* *URL:www.feminist.org*	5. (0.676) *Priests for Life Index* *URL:www.priestsforlife.org*	5. (0.552) *FEMINIST MAJORITY* *URL:www.feminist.org*
6. (0.484) *The Alan Guttmacher Institute: Ho* *URL:www.guttmacher.org*	6. (0.624) **Pregnancy Centers Online** **URL:www.pregnancycenters.org**	6. (0.545) **Abortion Clinics OnLine** **URL:www.gynpages.com**
7. (0.439) **center for reproductive right** **URL:www.crlp.org**	7. (0.602) *ProLifeInfo.org* *URL:www.prolifeinfo.org*	7. (0.538) *The Alan Guttmacher Institute: Ho* *URL:www.guttmacher.org*
8. (0.416) *The Religious Coalition for Repro* *URL:www.rcrc.org*	8. (0.557) **Abortion Clinics OnLine** **URL:www.gynpages.com**	8. (0.523) **center for reproductive right** **URL:www.crlp.org**
9. (0.415) *National Organization for Women* *URL:www.now.org*	9. (0.551) **After Abortion: Information o** **URL:www.afterabortion.org**	9. (0.518) *Priests for Life Index* *URL:www.priestsforlife.org*
10. (0.408) *Medical Students for Choice* *URL:www.ms4c.org*	10. (0.533) *FEMINIST MAJORITY* *URL:www.feminist.org*	10. (0.478) *The Religious Coalition for Repro* *URL:www.rcrc.org*

AT-avg	Norm	BFS
1. (1.000) *National Right to Life* *URL:www.nrlc.org*	1. (1.000) *National Right to Life* *URL:www.nrlc.org*	1. (1.000) *National Right to Life* *URL:www.nrlc.org*
2. (0.905) *Priests for Life Index* *URL:www.priestsforlife.org*	2. (0.966) *Priests for Life Index* *URL:www.priestsforlife.org*	2. (0.930) *Priests for Life Index* *URL:www.priestsforlife.org*
3. (0.844) *ProLifeInfo.org* *URL:www.prolifeinfo.org*	3. (0.929) **Pregnancy Centers Online** **URL:www.pregnancycenters.org**	3. (0.928) **After Abortion: Information o** **URL:www.afterabortion.org**
4. (0.785) **Pregnancy Centers Online** **URL:www.pregnancycenters.org**	4. (0.927) *ProLifeInfo.org* *URL:www.prolifeinfo.org*	4. (0.905) *Pro-life news and information fro* *URL:www.all.org*
5. (0.778) **After Abortion: Information o** **URL:www.afterabortion.org**	5. (0.914) **After Abortion: Information o** **URL:www.afterabortion.org**	5. (0.893) *ProLifeInfo.org* *URL:www.prolifeinfo.org*
6. (0.777) *prochoiceamerica.org : NARAL* *URL:www.naral.org*	6. (0.865) *Human Life International* *URL:www.hli.org*	6. (0.869) **Pregnancy Centers Online** **URL:www.pregnancycenters.org**
7. (0.741) *Human Life International* *URL:www.hli.org*	7. (0.860) *Abortion - Breast Cancer Link –* *URL:www.abortioncancer.com*	7. (0.860) *Human Life International* *URL:www.hli.org*
8. (0.704) *Planned Parenthood Federation of* *URL:www.plannedparenthood.org*	8. (0.848) **Abortion facts and informatio** **URL:www.abortionfacts.com**	8. (0.852) **Abortion facts and informatio** **URL:www.abortionfacts.com**
9. (0.683) **Abortion facts and informatio** **URL:www.abortionfacts.com**	9. (0.825) *Campaign Life Coalition British C* *URL:www.clcbc.org*	9. (0.847) *prochoiceamerica.org : NARAL* *URL:www.naral.org*
10. (0.677) *Abortion - Breast Cancer Link –* *URL:www.abortioncancer.com*	10. (0.787) **Coalition on Abortion/Breast** **URL:www.abortionbreastcancer.com**	10. (0.839) *Planned Parenthood Federation of* *URL:www.plannedparenthood.org*

Table C.1: Query "abortion"

Hits	PageRank	InDegree
1. (1.000) **Affirmative Action and Divers** **URL:aad.english.ucsb.edu**	1. (1.000) *TEXT* *URL:www.eoaa.vt.edu*	1. (1.000) Copyright Information URL:www.psu.edu/copyright.html
2. (0.719) **American Association for Affi** **URL:www.affirmativeaction.org**	2. (0.814) Faculty Jobs URL:www.ps.vt.edu/employment/facjo	2. (0.875) **Affirmative Action and Divers** **URL:aad.english.ucsb.edu**
3. (0.700) **U.S. Equal Employment** **URL:www.eeoc.gov**	3. (0.752) Adobe Acrobat Reader - Download URL:www.adobe.com/products/acrobat	3. (0.852) Adobe Acrobat Reader - Download URL:www.adobe.com/products/acrobat
4. (0.428) *National Organization for Women* *URL:www.now.org*	4. (0.696) *National Organization for Women* *URL:www.now.org*	4. (0.716) **U.S. Equal Employment** **URL:www.eeoc.gov**
5. (0.339) *The United States Department of L* *URL:www.dol.gov*	5. (0.507) Copyright Information URL:www.psu.edu/copyright.html	5. (0.682) **American Association for Affi** **URL:www.affirmativeaction.org**
6. (0.326) *DiversityWeb - A Resource Hub for* *URL:www.diversityweb.org*	6. (0.468) *The United States Department of L* *URL:www.dol.gov*	6. (0.568) Site Meter - Counter and Statis URL:sm6.sitemeter.com/stats.asp?si
7. (0.315) *Diversity Database, University of* *URL:www.inform.umd.edu/EdRes/Topic*	7. (0.421) *Bureau of Labor Statistics Home P* *URL:www.bls.gov*	7. (0.568) Free web counter - Site access URL:cqcounter.com/?_id=nsnewman_l
8. (0.285) Site Meter - Counter and Statis URL:sm6.sitemeter.com/stats.asp?si	8. (0.389) Search at SMSU URL:www.search.smsu.edu	8. (0.568) Free web counter - Site access URL:cqcounter.com
9. (0.285) Free web counter - Site access URL:cqcounter.com/?_id=nsnewman_l	9. (0.375) *National Organization for Women* *URL:www.nowpacs.org*	9. (0.545) *National Organization for Women* *URL:www.now.org*
10. (0.285) Free web counter - Site access URL:cqcounter.com	10. (0.337) The Truth About George W. Bush URL:www.thetruthaboutgeorge.com	10. (0.534) **Affirmative Action Register** **URL:www.aar-eeo.com**

HubAvg	Max	AT-med
1. (1.000) Copyright Information URL:www.psu.edu/copyright.html	1. (1.000) Copyright Information URL:www.psu.edu/copyright.html	1. (1.000) Copyright Information URL:www.psu.edu/copyright.html
2. (0.310) *PSU Affirmative Action* *URL:www.psu.edu/dept/aaoffice*	2. (0.447) *PSU Affirmative Action* *URL:www.psu.edu/dept/aaoffice*	2. (0.447) *PSU Affirmative Action* *URL:www.psu.edu/dept/aaoffice*
3. (0.193) Welcome to Penn State's Home on URL:www.psu.edu	3. (0.314) Welcome to Penn State's Home on URL:www.psu.edu	3. (0.314) Welcome to Penn State's Home on URL:www.psu.edu
4. (0.001) PSU Office for Disability Servi URL:www.lions.psu.edu/ods	4. (0.010) University of Illinois URL:www.uiuc.edu	4. (0.010) University of Illinois URL:www.uiuc.edu
5. (0.000) University of Illinois URL:www.uiuc.edu	5. (0.009) Purdue University-West Lafayett URL:www.purdue.edu	5. (0.009) Purdue University-West Lafayett URL:www.purdue.edu
6. (0.000) Purdue University-West Lafayett URL:www.purdue.edu	6. (0.008) UC Berkeley home page URL:www.berkeley.edu	6. (0.008) UC Berkeley home page URL:www.berkeley.edu
7. (0.000) University of Michigan URL:www.umich.edu	7. (0.008) University of Michigan URL:www.umich.edu	7. (0.008) University of Michigan URL:www.umich.edu
8. (0.000) UC Berkeley home page URL:www.berkeley.edu	8. (0.008) The University of Arizona URL:www.arizona.edu	8. (0.008) The University of Arizona URL:www.arizona.edu
9. (0.000) The University of Arizona URL:www.arizona.edu	9. (0.008) The University of Iowa Homepage URL:www.uiowa.edu	9. (0.008) The University of Iowa Homepage URL:www.uiowa.edu
10. (0.000) The University of Iowa Homepage URL:www.uiowa.edu	10. (0.008) Penn: University of Pennsylvani URL:www.upenn.edu	10. (0.008) Penn: University of Pennsylvani URL:www.upenn.edu

AT-avg	Norm	BFS
1. (1.000) Copyright Information URL:www.psu.edu/copyright.html	1. (1.000) Copyright Information URL:www.psu.edu/copyright.html	1. (1.000) **Affirmative Action and Divers** **URL:aad.english.ucsb.edu**
2. (0.568) *PSU Affirmative Action* *URL:www.psu.edu/dept/aaoffice*	2. (0.484) *PSU Affirmative Action* *URL:www.psu.edu/dept/aaoffice*	2. (0.845) **American Association for Affi** **URL:www.affirmativeaction.org**
3. (0.396) Welcome to Penn State's Home on URL:www.psu.edu	3. (0.343) Welcome to Penn State's Home on URL:www.psu.edu	3. (0.780) **U.S. Equal Employment** **URL:www.eeoc.gov**
4. (0.010) University of Illinois URL:www.uiuc.edu	4. (0.011) University of Illinois URL:www.uiuc.edu	4. (0.737) *National Organization for Women* *URL:www.now.org*
5. (0.009) Purdue University-West Lafayett URL:www.purdue.edu	5. (0.009) Purdue University-West Lafayett URL:www.purdue.edu	5. (0.727) Empty title field URL:www.auaa.org
6. (0.008) UC Berkeley home page URL:www.berkeley.edu	6. (0.009) UC Berkeley home page URL:www.berkeley.edu	6. (0.700) **Welcome to aadap.org. Here y** **URL:www.aadap.org**
7. (0.008) University of Michigan URL:www.umich.edu	7. (0.008) University of Michigan URL:www.umich.edu	7. (0.689) Adobe Acrobat Reader - Download URL:www.adobe.com/products/acrobat
8. (0.008) The University of Arizona URL:www.arizona.edu	8. (0.008) The University of Arizona URL:www.arizona.edu	8. (0.687) **CIR Home** **URL:www.cir-usa.org**
9. (0.008) The University of Iowa Homepage URL:www.uiowa.edu	9. (0.008) The University of Iowa Homepage URL:www.uiowa.edu	9. (0.661) *DiversityWeb - A Resource Hub for* *URL:www.diversityweb.org*
10. (0.008) Penn: University of Pennsylvani URL:www.upenn.edu	10. (0.008) Penn: University of Pennsylvani URL:www.upenn.edu	10. (0.657) **CAA** **URL:www.caasf.org**

Table C.2: Query "affirmative action"

Hits	PageRank	InDegree
1. (1.000) **NCADI: SAMHSA's The** **URL:www.health.org**	1. (1.000) FirstGov 8212; Your First Cli URL:www.firstgov.gov	1. (1.000) **NCADI: SAMHSA's The** **URL:www.health.org**
2. (0.904) **National Institute on Alcohol** **URL:www.niaaa.nih.gov**	2. (0.555) *United States Department of Healt* *URL:www.os.dhhs.gov*	2. (0.834) **National Institute on Alcohol** **URL:www.niaaa.nih.gov**
3. (0.794) *The Substance Abuse and Mental* *URL:www.samhsa.gov*	3. (0.432) Alice!gear: Go Ask Alice's Onli URL:www.alicegear.com	3. (0.811) *The Substance Abuse and Mental* *URL:www.samhsa.gov*
4. (0.703) *National Institute on Drug Abuse* *URL:www.nida.nih.gov*	4. (0.382) **National Institute on Alcohol** **URL:www.niaaa.nih.gov**	4. (0.612) *National Institute on Drug Abuse* *URL:www.nida.nih.gov*
5. (0.548) **Alcoholics Anonymous** **URL:www.alcoholics-anonymous.org**	5. (0.372) *Go Ask Alice! Home Page* *URL:www.goaskalice.columbia.edu*	5. (0.528) **Alcoholics Anonymous** **URL:www.alcoholics-anonymous.org**
6. (0.502) *Join Together Online - Take Actio* *URL:www.jointogether.org*	6. (0.295) National Institutes of Health (URL:www.nih.gov	6. (0.478) **Welcome to APOLNET** **URL:www.apolnet.org**
7. (0.475) **National Council on Alcoholis** **URL:www.ncadd.org**	7. (0.285) **NCADI: SAMHSA's The** **URL:www.health.org**	7. (0.432) Centers for Disease Control and URL:www.cdc.gov
8. (0.460) Welcome to the Office of Nation URL:www.whitehousedrugpolicy.gov	8. (0.268) Adobe Acrobat Reader - Download URL:www.adobe.com/products/acrobat	8. (0.426) **ETOH Home Page** **URL:etoh.niaaa.nih.gov**
9. (0.454) **Center on Alcohol Marketing a** **URL:camy.org**	9. (0.254) *The Substance Abuse and Mental* *URL:www.samhsa.gov*	9. (0.382) **Center on Alcohol Marketing a** **URL:camy.org**
10. (0.443) **Al-Anon/Alateen** **URL:www.al-anon.org**	10. (0.228) **Alcoholics Anonymous** **URL:www.alcoholics-anonymous.org**	10. (0.377) *United States Department of Healt* *URL:www.os.dhhs.gov*

HubAvg	Max	AT-med
1. (1.000) *The Substance Abuse and Mental* *URL:www.samhsa.gov*	1. (1.000) **NCADI: SAMHSA's The** **URL:www.health.org**	1. (1.000) **NCADI: SAMHSA's The** **URL:www.health.org**
2. (0.870) **NCADI: SAMHSA's The** **URL:www.health.org**	2. (0.739) **National Institute on Alcohol** **URL:www.niaaa.nih.gov**	2. (0.825) **National Institute on Alcohol** **URL:www.niaaa.nih.gov**
3. (0.693) **National Institute on Alcohol** **URL:www.niaaa.nih.gov**	3. (0.661) *The Substance Abuse and Mental* *URL:www.samhsa.gov*	3. (0.716) *The Substance Abuse and Mental* *URL:www.samhsa.gov*
4. (0.667) **ETOH Home Page** **URL:etoh.niaaa.nih.gov**	4. (0.526) *National Institute on Drug Abuse* *URL:www.nida.nih.gov*	4. (0.590) *National Institute on Drug Abuse* *URL:www.nida.nih.gov*
5. (0.595) *SAMHSA Web: Center for* *URL:www.samhsa.gov/centers/csap/cs*	5. (0.389) **Alcoholics Anonymous** **URL:www.alcoholics-anonymous.org**	5. (0.426) **Alcoholics Anonymous** **URL:www.alcoholics-anonymous.org**
6. (0.559) **Facility Locator** **URL:findtreatment.samhsa.gov/facil**	6. (0.313) *Join Together Online - Take Actio* *URL:www.jointogether.org*	6. (0.347) *Join Together Online - Take Actio* *URL:www.jointogether.org*
7. (0.544) **SAMHSA Web: Center for** **URL:www.samhsa.gov/centers/csat200**	7. (0.290) **National Council on Alcoholis** **URL:www.ncadd.org**	7. (0.326) **National Council on Alcoholis** **URL:www.ncadd.org**
8. (0.530) SAMHSA Web: Center for Mental URL:www.samhsa.gov/centers/cmhs/cm	8. (0.287) Centers for Disease Control and URL:www.cdc.gov	8. (0.318) Centers for Disease Control and URL:www.cdc.gov
9. (0.479) *National Institute on Drug Abuse* *URL:www.nida.nih.gov*	9. (0.283) Welcome to the Office of Nation URL:www.whitehousedrugpolicy.gov	9. (0.316) Welcome to the Office of Nation URL:www.whitehousedrugpolicy.gov
10. (0.320) **Alcoholics Anonymous** **URL:www.alcoholics-anonymous.org**	10. (0.275) **Higher Education Center for** **URL:www.edc.org/hec**	10. (0.306) **Higher Education Center for** **URL:www.edc.org/hec**

AT-avg	Norm	BFS
1. (1.000) **NCADI: SAMHSA's The** **URL:www.health.org**	1. (1.000) **NCADI: SAMHSA's The** **URL:www.health.org**	1. (1.000) **NCADI: SAMHSA's The** **URL:www.health.org**
2. (0.863) **National Institute on Alcohol** **URL:www.niaaa.nih.gov**	2. (0.831) **National Institute on Alcohol** **URL:www.niaaa.nih.gov**	2. (0.963) **National Institute on Alcohol** **URL:www.niaaa.nih.gov**
3. (0.794) *The Substance Abuse and Mental* *URL:www.samhsa.gov*	3. (0.748) *The Substance Abuse and Mental* *URL:www.samhsa.gov*	3. (0.894) *The Substance Abuse and Mental* *URL:www.samhsa.gov*
4. (0.644) *National Institute on Drug Abuse* *URL:www.nida.nih.gov*	4. (0.607) *National Institute on Drug Abuse* *URL:www.nida.nih.gov*	4. (0.863) *National Institute on Drug Abuse* *URL:www.nida.nih.gov*
5. (0.454) **Alcoholics Anonymous** **URL:www.alcoholics-anonymous.org**	5. (0.446) **Alcoholics Anonymous** **URL:www.alcoholics-anonymous.org**	5. (0.855) **Alcoholics Anonymous** **URL:www.alcoholics-anonymous.org**
6. (0.387) *Join Together Online - Take Actio* *URL:www.jointogether.org*	6. (0.374) *Join Together Online - Take Actio* *URL:www.jointogether.org*	6. (0.834) **Welcome to APOLNET** **URL:www.apolnet.org**
7. (0.363) **National Council on Alcoholis** **URL:www.ncadd.org**	7. (0.351) **National Council on Alcoholis** **URL:www.ncadd.org**	7. (0.796) Centers for Disease Control and URL:www.cdc.gov
8. (0.356) Welcome to the Office of Nation URL:www.whitehousedrugpolicy.gov	8. (0.342) Welcome to the Office of Nation URL:www.whitehousedrugpolicy.gov	8. (0.787) **National Council on Alcoholis** **URL:www.ncadd.org**
9. (0.343) Centers for Disease Control and URL:www.cdc.gov	9. (0.332) **Center on Alcohol Marketing a** **URL:camy.org**	9. (0.775) **Al-Anon/Alateen** **URL:www.al-anon.org**
10. (0.340) **Center on Alcohol Marketing** **URL:camy.org**	10. (0.330) Centers for Disease Control and URL:www.cdc.gov	10. (0.767) **Center on Alcohol Marketing** **URL:camy.org**

Table C.3: Query "alcohol"

Hits	PageRank	InDegree
1. (1.000) *Welcome to RENOLDI rides* *URL:www.renoldi.com*	1. (1.000) HONcode: Principles URL:www.hon.ch/HONcode/Conduct.htm	1. (1.000) HONcode: Principles URL:www.hon.ch/HONcode/Conduct.htm
2. (0.933) **IAAPA** **URL:www.iaapa.org**	2. (0.495) abc,Inflatable,moonwalk,moon bo URL:www.adventure-bounce.com	2. (0.645) **Empty title field** **URL:www.sixflags.com**
3. (0.834) **Knott's** **URL:www.knotts.com**	3. (0.335) Local Business Listings Beachco URL:business.beachcomberii.com	3. (0.589) **Busch Gardens Adventure** **URL:www.buschgardens.com**
4. (0.799) *Traditional Amusement parks of th* *URL:www.tradition.cjb.net*	4. (0.332) *NAARSO National Association* *URL:www.naarso.com*	4. (0.567) **IAAPA** **URL:www.iaapa.org**
5. (0.788) **HUSS** **URL:www.hussrides.com**	5. (0.332) AttorneyPages Helps You Find th URL:attorneypages.com	5. (0.560) Free Legal Advice in 100+ Law T URL:freeadvice.com
6. (0.779) *Empty title field* *URL:www.aimsintl.org*	6. (0.332) Do It Yourself Home Improvement URL:doityourself.com	6. (0.560) AttorneyPages Helps You Find th URL:attorneypages.com
7. (0.772) **Screamscape** **URL:www.screamscape.com**	7. (0.321) *Theme Parks Classifieds* *URL:adlistings.themeparks.about.co*	7. (0.560) Do It Yourself Home Improvement URL:doityourself.com
8. (0.770) **REVERCHON : HOME PAGE** **URL:www.reverchon.com**	8. (0.317) FreeFind Site Search URL:search.freefind.com/find.html?	8. (0.532) ExpertPages.com - Books, Tapes URL:expert-pages.com/books.htm
9. (0.769) *Empty title field* *URL:www.zierer.com*	9. (0.315) Free Legal Advice in 100+ Law T URL:freeadvice.com	9. (0.489) Empty title field URL:imgserv.adbutler.com/go2/;ID=1
10. (0.767) *DE* *URL:www.drewexpo.com*	10. (0.303) **IAAPA** **URL:www.iaapa.org**	10. (0.489) Empty title field URL:imgserv.adbutler.com/go2/;ID=1

HubAvg	Max	AT-med
1. (1.000) HONcode: Principles URL:www.hon.ch/HONcode/Conduct.htm	1. (1.000) HONcode: Principles URL:www.hon.ch/HONcode/Conduct.htm	1. (1.000) AttorneyPages Helps You Find th URL:attorneypages.com
2. (0.000) AttorneyPages Helps You Find th URL:attorneypages.com	2. (0.000) **Empty title field** **URL:www.sixflags.com**	2. (1.000) Do It Yourself Home Improvement URL:doityourself.com
3. (0.000) Do It Yourself Home Improvement URL:doityourself.com	3. (0.000) **Busch Gardens Adventure** **URL:www.buschgardens.com**	3. (0.987) Free Legal Advice in 100+ Law T URL:freeadvice.com
4. (0.000) Free Legal Advice in 100+ Law T URL:freeadvice.com	4. (0.000) **Cedar Point Amusement Park** **URL:www.cedarpoint.com**	4. (0.949) ExpertPages.com - Books, Tapes URL:expert-pages.com/books.htm
5. (0.000) ExpertPages.com - Books, Tapes URL:expert-pages.com/books.htm	5. (0.000) **IAAPA** **URL:www.iaapa.org**	5. (0.866) Accidents Happen - Why are Lawy URL:law.freeadvice.com/resources/c
6. (0.000) Accidents Happen - Why are Lawy URL:law.freeadvice.com/resources/c	6. (0.000) **Knott's** **URL:www.knotts.com**	6. (0.032) Expert Witness Directory — Fore URL:expertpages.com
7. (0.000) Empty title field URL:imgserv.adbutler.com/go2/;ID=1	7. (0.000) **Universal Studios** **URL:www.usf.com**	7. (0.017) Get Your Discount Card Today URL:www.usaphonetime.com
8. (0.000) Empty title field URL:imgserv.adbutler.com/go2/;ID=1	8. (0.000) *Welcome to RENOLDI rides* *URL:www.renoldi.com*	8. (0.016) MapQuest: Home URL:www.mapquest.com
9. (0.000) Site Meter - Counter and Statis URL:s10.sitemeter.com/stats.asp?si	9. (0.000) **Kennywood : America's Finest** **URL:www.kennywood.com**	9. (0.014) Adventure travel outdoor recr URL:www.outsidemag.com
10. (0.000) Empty title field URL:imgserv.adbutler.com/go2/;ID=1	10. (0.000) *Exhibits Collection – Amusement* *URL:www.learner.org/exhibits/parkp*	10. (0.013) **Disneyland Resort - The offi** **URL:www.disneyland.com**

AT-avg	Norm	BFS
1. (1.000) AttorneyPages Helps You Find th URL:attorneypages.com	1. (1.000) AttorneyPages Helps You Find th URL:attorneypages.com	1. (1.000) **Knott's** **URL:www.knotts.com**
2. (1.000) Do It Yourself Home Improvement URL:doityourself.com	2. (1.000) Do It Yourself Home Improvement URL:doityourself.com	2. (0.910) *Welcome to Gillette Shows* *URL:www.gilletteshows.biz*
3. (0.995) Free Legal Advice in 100+ Law T URL:freeadvice.com	3. (0.995) Free Legal Advice in 100+ Law T URL:freeadvice.com	3. (0.885) *Welcome to RENOLDI rides* *URL:www.renoldi.com*
4. (0.973) ExpertPages.com - Books, Tapes URL:expert-pages.com/books.htm	4. (0.962) ExpertPages.com - Books, Tapes URL:expert-pages.com/books.htm	4. (0.884) **IAAPA** **URL:www.iaapa.org**
5. (0.900) Accidents Happen - Why are Lawy URL:law.freeadvice.com/resources/c	5. (0.885) Accidents Happen - Why are Lawy URL:law.freeadvice.com/resources/c	5. (0.881) *e-musementparkstore.com* *URL:www.e-musementparkstore.com*
6. (0.016) Expert Witness Directory — Fore URL:expertpages.com	6. (0.025) Expert Witness Directory — Fore URL:expertpages.com	6. (0.879) *Great Adventure Source* *URL:greatadventure.8m.com*
7. (0.012) Get Your Discount Card Today URL:www.usaphonetime.com	7. (0.015) Get Your Discount Card Today URL:www.usaphonetime.com	7. (0.868) Web Page Under Construction URL:www.carousel.org
8. (0.008) The Expert Pages - About Advice URL:expertpages.com/about.htm	8. (0.011) MapQuest: Home URL:www.mapquest.com	8. (0.854) amutech URL:amutech.homestead.com
9. (0.008) Terms amp; Conditions at Exper URL:expertpages.com/conditions.htm	9. (0.010) The Expert Pages - About Advice URL:expertpages.com/about.htm	9. (0.850) **Joyrides - Amusement Park** **URL:www.joyrides.com**
10. (0.008) Expert Pages Privacy Policy URL:expertpages.com/privacy.htm	10. (0.010) Terms amp; Conditions at Exper URL:expertpages.com/conditions.htm	10. (0.849) **Pharaohs Lost Kingdom** **URL:www.pharaohslostkingdom.com**

Table C.4: Query "amusement parks"

Hits	PageRank	InDegree
1. (1.000) *All Conferences . Com* *URL:www.allconferences.com*	1. (1.000) Virginia Tech URL:www.vt.edu	1. (1.000) - - totemweb.com URL:www.totemweb.com
2. (0.996) Castles of the World Tours URL:www.castlesoftheworld.com	2. (0.808) University Libraries at Virgini URL:www.lib.vt.edu	2. (0.932) **Architecture Design Images Hi** **URL:www.greatbuildings.com**
3. (0.937) Affordable Globus Tours 11% Dis URL:www.affordableglobustours.com	3. (0.574) **Archeire - Irish Architecture** **URL:www.archeire.com**	3. (0.828) Google URL:www.google.com
4. (0.935) Trafalgar Tours: 12% Discount o URL:www.affordableTrafalgartours.c	4. (0.338) *Architecture Classifieds* *URL:adlistings.architecture.about.*	4. (0.818) **Empty title field** **URL:www.aia.org**
5. (0.935) Contiki Tours: 5% discount on C URL:www.affordableContikitours.com	5. (0.319) *AIFIA — Asilomar Institute for In* *URL:www.aifia.org*	5. (0.813) **ADAM, the Art, Design** **URL:adam.ac.uk**
6. (0.935) Ireland Tours: 5% Off CIE Tours URL:www.affordableIrelandtours.com	6. (0.315) **Architecture Ring** **URL:archring.hypermart.net**	6. (0.776) **Architecture.com** **URL:www.architecture.com**
7. (0.935) Collette Vacations: 5% discount URL:www.affordablecollettetours.co	7. (0.309) *Hirst Arts Fantasy Architecture.* *URL:www.hirstarts.com*	7. (0.766) **e-Architect** **URL:www.e-architect.com**
8. (0.931) Affordable Resorts - Discounts URL:www.affordableresorts.com	8. (0.307) *Welcome to Isfahan!* *URL:www.anglia.ac.uk/∼trochford/is*	8. (0.703) **Architecture Centre Bristol** **URL:www.arch-centre.demon.co.uk**
9. (0.931) Club Med Resorts- Discounted Cl URL:www.affordableclubmedresorts.c	9. (0.306) The Source for Java Technology URL:java.sun.com	9. (0.682) ReSources Home Page URL:www.resources.com
10. (0.931) Disney Vacation - Disney Vacati URL:www.affordabledisneyresorts.co	10. (0.305) **Medieval France Home Page** **URL:www.pitt.edu/∼medart/menufra**	10. (0.662) **The Institute of Classical A** **URL:www.classicist.org**

HubAvg	Max	AT-med
1. (1.000) *All Conferences . Com* *URL:www.allconferences.com*	1. (1.000) - - totemweb.com URL:www.totemweb.com	1. (1.000) - - totemweb.com URL:www.totemweb.com
2. (0.961) Castles of the World Tours URL:www.castlesoftheworld.com	2. (0.567) **Architecture Design Images Hi** **URL:www.greatbuildings.com**	2. (0.980) **Architecture Design Images Hi** **URL:www.greatbuildings.com**
3. (0.750) Crosses.org URL:www.crosses.org	3. (0.465) **Empty title field** **URL:www.aia.org**	3. (0.839) **ADAM, the Art, Design** **URL:adam.ac.uk**
4. (0.703) Castles Hotels URL:www.castles-hotels.com	4. (0.406) Google URL:www.google.com	4. (0.773) **Empty title field** **URL:www.aia.org**
5. (0.703) Castles For Sale URL:www.castles-for-sale.com	5. (0.384) **e-Architect** **URL:www.e-architect.com**	5. (0.770) **e-Architect** **URL:www.e-architect.com**
6. (0.648) Past Tours from Castles of the URL:www.castlesoftheworld.com/Past	6. (0.378) **ADAM, the Art, Design** **URL:adam.ac.uk**	6. (0.766) **Architecture.com** **URL:www.architecture.com**
7. (0.620) Affordable Globus Tours 11% Dis URL:www.affordableglobustours.com	7. (0.372) ReSources Home Page URL:www.resources.com	7. (0.622) *Fine Art - World Wide Arts Resour* *URL:wwar.com*
8. (0.608) Trafalgar Tours: 12% Discount o URL:www.affordableTrafalgartours.c	8. (0.365) **Architecture.com** **URL:www.architecture.com**	8. (0.558) **Architecture Web Resources** **URL:library.nevada.edu/arch/rsrce/**
9. (0.608) Contiki Tours: 5% discount on C URL:www.affordableContikitours.com	9. (0.351) What You Need to Know About84 URL:www.about.com	9. (0.549) ReSources Home Page URL:www.resources.com
10. (0.608) Ireland Tours: 5% Off CIE Tours URL:www.affordableIrelandtours.com	10. (0.318) **Architecture Centre Bristol** **URL:www.arch-centre.demon.co.uk**	10. (0.546) What You Need to Know About84 URL:www.about.com

AT-avg	Norm	BFS
1. (1.000) *All Conferences . Com* *URL:www.allconferences.com*	1. (1.000) *All Conferences . Com* *URL:www.allconferences.com*	1. (1.000) - - totemweb.com URL:www.totemweb.com
2. (0.940) Castles of the World Tours URL:www.castlesoftheworld.com	2. (0.969) Castles of the World Tours URL:www.castlesoftheworld.com	2. (0.956) ReSources Home Page URL:www.resources.com
3. (0.743) Crosses.org URL:www.crosses.org	3. (0.828) Affordable Globus Tours 11% Dis URL:www.affordableglobustours.com	3. (0.956) What You Need to Know About84 URL:www.about.com
4. (0.704) Affordable Globus Tours 11% Dis URL:www.affordableglobustours.com	4. (0.823) Trafalgar Tours: 12% Discount o URL:www.affordableTrafalgartours.c	4. (0.949) **Architecture Centre Bristol** **URL:www.arch-centre.demon.co.uk**
5. (0.695) Collette Vacations: 5% discount URL:www.affordablecollettetours.co	5. (0.823) Contiki Tours: 5% discount on C URL:www.affordableContikitours.com	5. (0.948) **ADAM, the Art, Design** **URL:adam.ac.uk**
6. (0.695) Trafalgar Tours: 12% Discount o URL:www.affordableTrafalgartours.c	6. (0.823) Ireland Tours: 5% Off CIE Tours URL:www.affordableIrelandtours.com	6. (0.948) **Empty title field** **URL:www.aia.org**
7. (0.695) Contiki Tours: 5% discount on C URL:www.affordableContikitours.com	7. (0.823) Collette Vacations: 5% discount URL:www.affordablecollettetours.co	7. (0.941) **e-Architect** **URL:www.e-architect.com**
8. (0.695) Ireland Tours: 5% Off CIE Tours URL:www.affordableIrelandtours.com	8. (0.817) Affordable Resorts - Discounts URL:www.affordableresorts.com	8. (0.936) **The Institute of Classical Ar** **URL:www.classicist.org**
9. (0.686) Affordable Tours - Discounts on URL:www.AffordableTours.com	9. (0.817) Club Med Resorts- Discounted Cl URL:www.affordableclubmedresorts.c	9. (0.931) *AIFIA — Asilomar Institute for In* *URL:www.aifia.org*
10. (0.686) Affordable Resorts - Discounts URL:www.affordableresorts.com	10. (0.817) Disney Vacation - Disney Vacati URL:www.affordabledisneyresorts.co	10. (0.929) SourceNom.com URL:www.p-pub.com

Table C.5: Query “architecture”

Hits	PageRank	InDegree
1. (1.000) Town Hall Book Service URL:www.thbookservice.com	1. (1.000) WETA TV 26/ 90.9 FM URL:www.weta.org	1. (1.000) Town Hall Book Service URL:www.thbookservice.com
2. (0.995) World Vision URL:etools.ncol.com/a/jgroup/bg_wo	2. (0.850) *PBS - JAZZ A Film By Ken Burns* *URL:www.pbs.org/jazz*	2. (0.970) World Vision URL:etools.ncol.com/a/jgroup/bg_wo
3. (0.995) Town Hall - Letter from David L URL:cf.heritage.org/rd.cfm?id=36	3. (0.662) *Armstrong International, Inc. - s* *URL:www.armstrong-intl.com*	3. (0.970) Town Hall - Letter from David L URL:cf.heritage.org/rd.cfm?id=36
4. (0.995) World Vision URL:etools.ncol.com/a/jgroup/bg_wo	4. (0.458) *Armstrong Atlantic State Univers* *URL:www.armstrong.edu*	4. (0.970) World Vision URL:etools.ncol.com/a/jgroup/bg_wo
5. (0.995) Patterson, Colonel Robert: Dere URL:www.thbookservice.com/BookPage	5. (0.458) GraphicSmiths - Innovative, Pro URL:www.graphicsmith.com/gs	5. (0.970) Patterson, Colonel Robert: Dere URL:www.thbookservice.com/BookPage
6. (0.995) Welcome to the Alexa Toolbar Do URL:cf.heritage.org/rd.cfm?id=286	6. (0.410) Town Hall Book Service URL:www.thbookservice.com	6. (0.970) Welcome to the Alexa Toolbar Do URL:cf.heritage.org/rd.cfm?id=286
7. (0.215) *Amazon.com: Books: Letters to* *URL:www.amazon.com/exec/obidos/tg/*	7. (0.405) *Armstrong International European* *URL:www.armstrong.be*	7. (0.958) FIETSEN TEGEN KANKER URL:www.fietsentegenkanker.org
8. (0.012) Amazon.com: Books: Letters to a URL:www.amazon.com/exec/obidos/ASI	8. (0.393) Board of Regents of the Univers URL:www.usg.edu	8. (0.928) *Herbert W. Armstrong Library and* *URL:www.herbertwarmstrong.org*
9. (0.012) *Amazon.com: Books: Beyond* *URL:www.amazon.com/exec/obidos/ASI*	9. (0.378) *Welcome To Timeless Records* *URL:www.timeless-records.com*	9. (0.737) Biblical Evidence for Catholici URL:www.biblicalcatholic.com
10. (0.001) Empty title field URL:www.townhall.com	10. (0.375) FIETSEN TEGEN KANKER URL:www.fietsentegenkanker.org	10. (0.719) *R.V. Armstrong amp Associates* *URL:www.rvarmstrong.com*

HubAvg	Max	AT-med
1. (1.000) Town Hall Book Service URL:www.thbookservice.com	1. (1.000) Town Hall Book Service URL:www.thbookservice.com	1. (1.000) Town Hall Book Service URL:www.thbookservice.com
2. (0.972) World Vision URL:etools.ncol.com/a/jgroup/bg_wo	2. (0.970) World Vision URL:etools.ncol.com/a/jgroup/bg_wo	2. (0.985) World Vision URL:etools.ncol.com/a/jgroup/bg_wo
3. (0.972) Town Hall - Letter from David L URL:cf.heritage.org/rd.cfm?id=36	3. (0.970) Town Hall - Letter from David L URL:cf.heritage.org/rd.cfm?id=36	3. (0.985) Town Hall - Letter from David L URL:cf.heritage.org/rd.cfm?id=36
4. (0.972) World Vision URL:etools.ncol.com/a/jgroup/bg_wo	4. (0.970) World Vision URL:etools.ncol.com/a/jgroup/bg_wo	4. (0.985) World Vision URL:etools.ncol.com/a/jgroup/bg_wo
5. (0.972) Patterson, Colonel Robert: Dere URL:www.thbookservice.com/BookPage	5. (0.970) Patterson, Colonel Robert: Dere URL:www.thbookservice.com/BookPage	5. (0.985) Patterson, Colonel Robert: Dere URL:www.thbookservice.com/BookPage
6. (0.972) Welcome to the Alexa Toolbar Do URL:cf.heritage.org/rd.cfm?id=286	6. (0.970) Welcome to the Alexa Toolbar Do URL:cf.heritage.org/rd.cfm?id=286	6. (0.985) Welcome to the Alexa Toolbar Do URL:cf.heritage.org/rd.cfm?id=286
7. (0.185) *Amazon.com: Books: Letters to* *URL:www.amazon.com/exec/obidos/tg/*	7. (0.204) *Amazon.com: Books: Letters to* *URL:www.amazon.com/exec/obidos/tg/*	7. (0.207) *Amazon.com: Books: Letters to* *URL:www.amazon.com/exec/obidos/tg/*
8. (0.010) Amazon.com: Books: Letters to a URL:www.amazon.com/exec/obidos/ASI	8. (0.012) Amazon.com: Books: Letters to a URL:www.amazon.com/exec/obidos/ASI	8. (0.012) Amazon.com: Books: Letters to a URL:www.amazon.com/exec/obidos/ASI
9. (0.010) *Amazon.com: Books: Beyond* *URL:www.amazon.com/exec/obidos/ASI*	9. (0.012) *Amazon.com: Books: Beyond* *URL:www.amazon.com/exec/obidos/ASI*	9. (0.012) *Amazon.com: Books: Beyond* *URL:www.amazon.com/exec/obidos/ASI*
10. (0.003) Empty title field URL:www.townhall.com	10. (0.006) Empty title field URL:www.townhall.com	10. (0.003) Empty title field URL:www.townhall.com

AT-avg	Norm	BFS
1. (1.000) Town Hall Book Service URL:www.thbookservice.com	1. (1.000) Town Hall Book Service URL:www.thbookservice.com	1. (1.000) FIETSEN TEGEN KANKER URL:www.fietsentegenkanker.org
2. (0.990) World Vision URL:etools.ncol.com/a/jgroup/bg_wo	2. (0.987) World Vision URL:etools.ncol.com/a/jgroup/bg_wo	2. (0.973) *Herbert W. Armstrong Library and* *URL:www.herbertwarmstrong.org*
3. (0.990) Town Hall - Letter from David L URL:cf.heritage.org/rd.cfm?id=36	3. (0.987) Town Hall - Letter from David L URL:cf.heritage.org/rd.cfm?id=36	3. (0.942) Biblical Evidence for Catholici URL:www.biblicalcatholic.com
4. (0.990) World Vision URL:etools.ncol.com/a/jgroup/bg_wo	4. (0.987) World Vision URL:etools.ncol.com/a/jgroup/bg_wo	4. (0.887) *R.V. Armstrong amp Associates IS* *URL:www.rvarmstrong.com*
5. (0.990) Patterson, Colonel Robert: Dere URL:www.thbookservice.com/BookPage	5. (0.987) Patterson, Colonel Robert: Dere URL:www.thbookservice.com/BookPage	5. (0.855) Geist: Canadian Ideas, Canadian URL:geist.com
6. (0.990) Welcome to the Alexa Toolbar Do URL:cf.heritage.org/rd.cfm?id=286	6. (0.987) Welcome to the Alexa Toolbar Do URL:cf.heritage.org/rd.cfm?id=286	6. (0.821) *CleanReg by Armstrong's Systems* *URL:www.cleanreg.com*
7. (0.208) *Amazon.com: Books: Letters to* *URL:www.amazon.com/exec/obidos/tg/*	7. (0.208) *Amazon.com: Books: Letters to* *URL:www.amazon.com/exec/obidos/tg/*	7. (0.791) *The Unofficial Lance Armstrong Fa* *URL:www.lancearmstrongfanclub.com*
8. (0.012) Amazon.com: Books: Letters to a URL:www.amazon.com/exec/obidos/ASI	8. (0.012) Amazon.com: Books: Letters to a URL:www.amazon.com/exec/obidos/ASI	8. (0.746) Visual Escapes - artist directo URL:surrealities.cjb.net
9. (0.012) *Amazon.com: Books: Beyond* *URL:www.amazon.com/exec/obidos/ASI*	9. (0.012) *Amazon.com: Books: Beyond* *URL:www.amazon.com/exec/obidos/ASI*	9. (0.734) brushstroke.tv URL:www.brushstroke.tv
10. (0.002) Empty title field URL:www.townhall.com	10. (0.003) Empty title field URL:www.townhall.com	10. (0.725) *Bienvenidos a la Municipalidad de* *URL:www.armstrong.gov.ar*

Table C.6: Query "armstrong"

Hits	PageRank	InDegree
1. (1.000) E Business Solutions,Website Pr URL:www.intermesh.net/advertis.htm	1. (1.000) Hoovers Online Job Postings URL:RecruitingCenter.net/clients/H	1. (1.000) Travel - India Travel,Tourism I URL:www.indiantravelportal.com
2. (0.979) *Empty title field* *URL:auto.indiamart.com*	2. (0.680) twp home page URL:www.worldpages.com/TWP/twpwebs	2. (0.873) Empty title field URL:www.indiantravelportal.com/taj
3. (0.978) Empty title field URL:www.indiamart.com	3. (0.585) Travel - India Travel,Tourism I URL:www.indiantravelportal.com	3. (0.618) Government of Canada Site — Sit URL:canada.gc.ca
4. (0.975) Empty title field URL:www.indiangiftsportal.com	4. (0.554) Government of Canada Site — Sit URL:canada.gc.ca	4. (0.578) twp home page URL:www.worldpages.com/TWP/twpwebs
5. (0.972) Jewelry Box, Jewelry Gift Box, URL:www.indiangiftsportal.com/indi	5. (0.431) The National Academies URL:www.nationalacademies.org	5. (0.510) *Empty title field* *URL:www.bombaymotor.com*
6. (0.972) Birthday Gifts,Birthday Gift Id URL:www.indiangiftsportal.com/indi	6. (0.425) Introduction au site Web offici URL:www.canada.gc.ca/main_f.html	6. (0.471) Empty title field URL:www.indiantravelportal.com/ind
7. (0.972) Anniversary Gifts,Wedding Anniv URL:www.indiangiftsportal.com/indi	7. (0.414) Empty title field URL:www.indiantravelportal.com/taj	7. (0.461) Mesurer et analyser l'audience URL:www.xiti.com/xiti.asp?s=27855
8. (0.972) Wedding Gifts,Wedding Anniversa URL:www.indiangiftsportal.com/indi	8. (0.359) National Academies Press URL:www.nap.edu	8. (0.461) HitBoxCentral - HitBox Central URL:rd1.hitbox.com/rd?acct=WQ50071
9. (0.972) Mixed Bag, Exclusives, Indian G URL:www.indiangiftsportal.com/indi	9. (0.318) **Empty title field** **URL:www.aiacanada.com**	9. (0.451) Weborama leader europen de la URL:www.weborama.com
10. (0.972) Business Solutions,Ecommerce Bu URL:www.intermesh.net	10. (0.311) Group Goldwire, Complete Intern URL:www.goldwire.com	10. (0.431) **Automotive Industries** **URL:www.ai-online.com**

HubAvg	Max	AT-med
1. (1.000) Travel - India Travel,Tourism I URL:www.indiantravelportal.com	1. (1.000) Travel - India Travel,Tourism I URL:www.indiantravelportal.com	1. (1.000) Travel - India Travel,Tourism I URL:www.indiantravelportal.com
2. (0.841) Empty title field URL:www.indiantravelportal.com/taj	2. (0.873) Empty title field URL:www.indiantravelportal.com/taj	2. (0.917) Empty title field URL:www.indiantravelportal.com/taj
3. (0.525) Empty title field URL:www.indiantravelportal.com/ind	3. (0.505) *Empty title field* *URL:www.bombaymotor.com*	3. (0.524) *Empty title field* *URL:www.bombaymotor.com*
4. (0.400) *Empty title field* *URL:www.bombaymotor.com*	4. (0.471) Empty title field URL:www.indiantravelportal.com/ind	4. (0.485) Empty title field URL:www.indiantravelportal.com/ind
5. (0.220) Adventure Tour Travel,India Adv URL:www.indiantravelportal.com/adv	5. (0.284) Adventure Tour Travel,India Adv URL:www.indiantravelportal.com/adv	5. (0.299) Adventure Tour Travel,India Adv URL:www.indiantravelportal.com/adv
6. (0.121) Himalayas,Himalaya,India Himala URL:www.indiantravelportal.com/him	6. (0.206) Himalayas,Himalaya,India Himala URL:www.indiantravelportal.com/him	6. (0.216) Himalayas,Himalaya,India Himala URL:www.indiantravelportal.com/him
7. (0.095) Empty title field URL:www.indiantravelportal.com/tre	7. (0.147) Empty title field URL:www.indiantravelportal.com/tre	7. (0.154) Empty title field URL:www.indiantravelportal.com/tre
8. (0.050) Empty title field URL:www.indiantravelportal.com/fai	8. (0.098) Empty title field URL:www.indiantravelportal.com/fai	8. (0.103) Empty title field URL:www.indiantravelportal.com/fai
9. (0.050) Empty title field URL:www.indiantravelportal.com/fes	9. (0.098) Empty title field URL:www.indiantravelportal.com/fes	9. (0.103) Empty title field URL:www.indiantravelportal.com/fes
10. (0.012) Tripura,Tripura India,Tourism i URL:www.indiantravelportal.com/tri	10. (0.029) Tripura,Tripura India,Tourism i URL:www.indiantravelportal.com/tri	10. (0.031) Tripura,Tripura India,Tourism i URL:www.indiantravelportal.com/tri

AT-avg	Norm	BFS
1. (1.000) Travel - India Travel,Tourism I URL:www.indiantravelportal.com	1. (1.000) Travel - India Travel,Tourism I URL:www.indiantravelportal.com	1. (1.000) The Ontario Neurotrauma Foundat URL:www.onf.org
2. (0.935) Empty title field URL:www.indiantravelportal.com/taj	2. (0.905) Empty title field URL:www.indiantravelportal.com/taj	2. (0.836) **Automotive Industries** **URL:www.ai-online.com**
3. (0.564) *Empty title field* *URL:www.bombaymotor.com*	3. (0.526) *Empty title field* *URL:www.bombaymotor.com*	3. (0.812) Ward's Dealer Business URL:wdb.wardsauto.com
4. (0.462) Empty title field URL:www.indiantravelportal.com/ind	4. (0.473) Empty title field URL:www.indiantravelportal.com/ind	4. (0.787) Travel - India Travel,Tourism I URL:www.indiantravelportal.com
5. (0.334) Adventure Tour Travel,India Adv URL:www.indiantravelportal.com/adv	5. (0.302) Adventure Tour Travel,India Adv URL:www.indiantravelportal.com/adv	5. (0.751) *Automobile Magazine* *URL:www.automobilemag.com*
6. (0.239) Himalayas,Himalaya,India Himala URL:www.indiantravelportal.com/him	6. (0.219) Himalayas,Himalaya,India Himala URL:www.indiantravelportal.com/him	6. (0.739) *Empty title field* *URL:www.neoliteppi.com*
7. (0.172) Empty title field URL:www.indiantravelportal.com/tre	7. (0.158) Empty title field URL:www.indiantravelportal.com/tre	7. (0.734) **DaimlerChrysler** **URL:www.daimlerchrysler.com**
8. (0.113) Empty title field URL:www.indiantravelportal.com/fai	8. (0.104) Empty title field URL:www.indiantravelportal.com/fai	8. (0.733) **Empty title field** **URL:www.auto.com**
9. (0.113) Empty title field URL:www.indiantravelportal.com/fes	9. (0.104) Empty title field URL:www.indiantravelportal.com/fes	9. (0.717) Empty title field URL:www.indiantravelportal.com/taj
10. (0.035) Tripura,Tripura India,Tourism i URL:www.indiantravelportal.com/tri	10. (0.032) Tripura,Tripura India,Tourism i URL:www.indiantravelportal.com/tri	10. (0.715) **Empty title field** **URL:www.ford.com**

Table C.7: Query "automobile industries"

Hits	PageRank	InDegree
1. (1.000) Welcome to MSN.com URL:g.msn.com/0nwenus0/AK/14	1. (1.000) Student Advantage Discount Card URL:www.studentadvantage.com	1. (1.000) **NBA.com** **URL:www.nba.com**
2. (0.994) Welcome to MSN.com URL:g.msn.com/0nwenus0/AK/07	2. (0.630) **NBA.com** **URL:www.nba.com**	2. (0.996) Student Advantage Discount Card URL:www.studentadvantage.com
3. (0.994) Welcome to MSN.com URL:g.msn.com/0nwenus0/AK/08	3. (0.362) Knight Ridder Corporate Web sit URL:www.knightridder.com	3. (0.425) **FIBA - International Basketba** **URL:www.fiba.com**
4. (0.994) Empty title field URL:g.msn.com/0nwenus0/AK/09	4. (0.329) **NCAA Online** **URL:www.ncaa.org**	4. (0.392) Welcome to MSN.com URL:g.msn.com/0nwenus0/AK/14
5. (0.994) MSN Search – More Useful Every URL:g.msn.com/0nwenus0/AK/10	5. (0.313) **FIBA - International Basketba** **URL:www.fiba.com**	5. (0.387) Welcome to MSN.com URL:g.msn.com/0nwenus0/AK/07
6. (0.994) Welcome to MSN Shopping URL:g.msn.com/0nwenus0/AK/11	6. (0.234) **National Basketball League** **URL:www.nbl.com.au**	6. (0.387) Welcome to MSN.com URL:g.msn.com/0nwenus0/AK/08
7. (0.994) MSN Money - More Useful Everyda URL:g.msn.com/0nwenus0/AK/12	7. (0.227) *National Association of Basketbal* *URL:www.nabc.com*	7. (0.387) Empty title field URL:g.msn.com/0nwenus0/AK/09
8. (0.994) MSN People and Chat - More Usef URL:g.msn.com/0nwenus0/AK/13	8. (0.223) Realcities.com URL:www.realcities.com	8. (0.387) MSN Search – More Useful Every URL:g.msn.com/0nwenus0/AK/10
9. (0.988) Welcome to MSN.com URL:g.msn.com/0nwenus0/AK/00	9. (0.214) **Index of /** **URL:www.internationalbasketball.co**	9. (0.387) Welcome to MSN Shopping URL:g.msn.com/0nwenus0/AK/11
10. (0.988) Welcome to MSN.com URL:g.msn.com/0nwenus0/AK/01	10. (0.204) **The Official Web Site of the** **URL:www.hoophall.com**	10. (0.387) MSN Money - More Useful URL:g.msn.com/0nwenus0/AK/12

HubAvg	Max	AT-med
1. (1.000) Student Advantage Discount Card URL:www.studentadvantage.com	1. (1.000) **NBA.com** **URL:www.nba.com**	1. (1.000) **NBA.com** **URL:www.nba.com**
2. (0.103) The University of North Carolin URL:www.unc.edu	2. (0.326) Welcome to MSN.com URL:g.msn.com/0nwenus0/AK/14	2. (0.532) Welcome to MSN.com URL:g.msn.com/0nwenus0/AK/14
3. (0.103) University of North Carolina - URL:www.mediateamlink.com/oas/unc	3. (0.322) Welcome to MSN.com URL:g.msn.com/0nwenus0/AK/07	3. (0.526) Welcome to MSN.com URL:g.msn.com/0nwenus0/AK/07
4. (0.095) UNC Rams Club URL:www.ramsclub.org/home/5805.asp	4. (0.322) Welcome to MSN.com URL:g.msn.com/0nwenus0/AK/08	4. (0.526) Welcome to MSN.com URL:g.msn.com/0nwenus0/AK/08
5. (0.076) Florida State University Varsit URL:www.fsuvarsityclub.org	5. (0.322) Empty title field URL:g.msn.com/0nwenus0/AK/09	5. (0.526) Empty title field URL:g.msn.com/0nwenus0/AK/09
6. (0.074) www.seminole-boosters.com URL:www.seminole-boosters.com	6. (0.322) MSN Search – More Useful Every URL:g.msn.com/0nwenus0/AK/10	6. (0.526) MSN Search – More Useful Every URL:g.msn.com/0nwenus0/AK/10
7. (0.074) Tallahassee Map URL:www.fsu.edu/Welcome/tallymaps/	7. (0.322) Welcome to MSN Shopping URL:g.msn.com/0nwenus0/AK/11	7. (0.526) Welcome to MSN Shopping URL:g.msn.com/0nwenus0/AK/11
8. (0.072) Welcome to Duke University Stor URL:www.dukestore.com	8. (0.322) MSN Money - More Useful Everyda URL:g.msn.com/0nwenus0/AK/12	8. (0.526) MSN Money - More Useful Everyda URL:g.msn.com/0nwenus0/AK/12
9. (0.071) Empty title field URL:netstile.evenue.net/evenue/se/	9. (0.322) MSN People and Chat - More Usef URL:g.msn.com/0nwenus0/AK/13	9. (0.526) MSN People and Chat - More Usef URL:g.msn.com/0nwenus0/AK/13
10. (0.067) University of Notre Dame URL:www.nd.edu	10. (0.319) Welcome to MSN.com URL:g.msn.com/0nwenus0/AK/00	10. (0.521) Welcome to MSN.com URL:g.msn.com/0nwenus0/AK/00

AT-avg	Norm	BFS
1. (1.000) **NBA.com** **URL:www.nba.com**	1. (1.000) **NBA.com** **URL:www.nba.com**	1. (1.000) **NBA.com** **URL:www.nba.com**
2. (0.783) Welcome to MSN.com URL:g.msn.com/0nwenus0/AK/14	2. (0.715) Welcome to MSN.com URL:g.msn.com/0nwenus0/AK/14	2. (0.842) **FIBA - International Basketba** **URL:www.fiba.com**
3. (0.775) Welcome to MSN.com URL:g.msn.com/0nwenus0/AK/07	3. (0.709) Welcome to MSN.com URL:g.msn.com/0nwenus0/AK/07	3. (0.788) **NCAA Online** **URL:www.ncaa.org**
4. (0.775) Welcome to MSN.com URL:g.msn.com/0nwenus0/AK/08	4. (0.709) Welcome to MSN.com URL:g.msn.com/0nwenus0/AK/08	4. (0.762) **The Official Site of USA Bask** **URL:www.usabasketball.com**
5. (0.775) Empty title field URL:g.msn.com/0nwenus0/AK/09	5. (0.709) Empty title field URL:g.msn.com/0nwenus0/AK/09	5. (0.753) **Men's Basketball - NCAA** **URL:www.finalfour.net**
6. (0.775) MSN Search – More Useful Every URL:g.msn.com/0nwenus0/AK/10	6. (0.709) MSN Search – More Useful Every URL:g.msn.com/0nwenus0/AK/10	6. (0.743) *ESPN.com* *URL:www.espn.com*
7. (0.775) Welcome to MSN Shopping URL:g.msn.com/0nwenus0/AK/11	7. (0.709) Welcome to MSN Shopping URL:g.msn.com/0nwenus0/AK/11	7. (0.723) *Winning Hoops Basketball* *URL:www.winninghoops.com*
8. (0.775) MSN Money - More Useful Everyda URL:g.msn.com/0nwenus0/AK/12	8. (0.709) MSN Money - More Useful Everyda URL:g.msn.com/0nwenus0/AK/12	8. (0.723) *Rivals.com* *URL:www.rivals.com*
9. (0.775) MSN People and Chat - More Usef URL:g.msn.com/0nwenus0/AK/13	9. (0.709) MSN People and Chat - More Usef URL:g.msn.com/0nwenus0/AK/13	9. (0.715) **The Official Web Site of the** **URL:www.hoophall.com**
10. (0.766) Welcome to MSN.com URL:g.msn.com/0nwenus0/AK/00	10. (0.703) Welcome to MSN.com URL:g.msn.com/0nwenus0/AK/00	10. (0.712) *Duke University Blue Devils - Off* *URL:www.goduke.com*

Table C.8: Query "basketball"

Hits	PageRank	InDegree
1. (1.000) **The Blues Foundation. Your** **URL:www.blues.org**	1. (1.000) Vivendi Universal URL:www.vivendiuniversal.com	1. (1.000) **The Blues Foundation. Your** **URL:www.blues.org**
2. (0.927) WebRing: addsite_login URL:l.webring.com/wrman?ring=blues	2. (0.876) *MP3.com: THE destination for digi* *URL:www.mp3.com*	2. (0.864) *Harry's Blues Lyrics Online, Home* *URL:blueslinks.tripod.com*
3. (0.890) **A Cyber Blues Society for mus** **URL:www.bluessociety.net**	3. (0.664) **The Blues Foundation. Your** **URL:www.blues.org**	3. (0.713) Delta Blues - DeltaBlues - delt URL:www.deltablues.com
4. (0.866) Google Search: URL:www.google.com/search	4. (0.566) Rob Hutten's Home Page URL:www.hutten.org/rob	4. (0.691) **The Blue Highway** **URL:www.thebluehighway.com**
5. (0.863) **BluesSociety.net - for musici** **URL:sitebuilder.bluessociety.net**	5. (0.504) **BluesNet home page** **URL:bluesnet.hub.org**	5. (0.662) *home* *URL:www.fargobluesfest.com*
6. (0.855) Your Mailinglist Provider URL:www.yourmailinglistprovider.co	6. (0.445) *Lily Sazz* *URL:www.lilysazz.com*	6. (0.656) stantonanderson.com URL:www.stantonanderson.com
7. (0.850) BluesSociety.net - Free Email S URL:mail.bluessociety.net	7. (0.430) *FestivalFinder: Music Festivals o* *URL:www.festivalfinder.com*	7. (0.634) Dan's Police Page URL:danspolice.8m.com
8. (0.847) **A Cyber Blues Society Blues L** **URL:www.bluessociety.net/links03.h**	8. (0.413) *Harry's Blues Lyrics Online, Home* *URL:blueslinks.tripod.com*	8. (0.520) **Blues On Stage, your complete** **URL:www.mnblues.com**
9. (0.847) **A Cyber Blues Society Blues L** **URL:www.bluessociety.net/links05.h**	9. (0.393) **YEAR OF THE BLUES 2003** **URL:www.yearoftheblues.org**	9. (0.502) **A Cyber Blues Society for mus** **URL:www.bluessociety.net**
10. (0.847) **Blues Biographies - Artist o** **URL:www.bluessociety.net/greats.**	10. (0.334) **Southwest Blues** **URL:www.southwestblues.com**	10. (0.495) Empty title field URL:www.letthegoodtimesroll.com

HubAvg	Max	AT-med
1. (1.000) **The Blues Foundation. Your** **URL:www.blues.org**	1. (1.000) **The Blues Foundation. Your** **URL:www.blues.org**	1. (1.000) **The Blues Foundation. Your** **URL:www.blues.org**
2. (0.687) *Harry's Blues Lyrics Online, Home* *URL:blueslinks.tripod.com*	2. (0.500) *Harry's Blues Lyrics Online, Home* *URL:blueslinks.tripod.com*	2. (0.703) *Harry's Blues Lyrics Online, Home* *URL:blueslinks.tripod.com*
3. (0.568) **The Blue Highway** **URL:www.thebluehighway.com**	3. (0.413) **The Blue Highway** **URL:www.thebluehighway.com**	3. (0.566) **The Blue Highway** **URL:www.thebluehighway.com**
4. (0.524) Delta Blues - DeltaBlues - delt URL:www.deltablues.com	4. (0.386) **A Cyber Blues Society for mus** **URL:www.bluessociety.net**	4. (0.452) Delta Blues - DeltaBlues - delt URL:www.deltablues.com
5. (0.445) *home* *URL:www.fargobluesfest.com*	5. (0.316) WebRing: addsite_login URL:l.webring.com/wrman?ring=blues	5. (0.429) **A Cyber Blues Society for mus** **URL:www.bluessociety.net**
6. (0.432) stantonanderson.com URL:www.stantonanderson.com	6. (0.310) **BluesSociety.net - for musici** **URL:sitebuilder.bluessociety.net**	6. (0.428) *home* *URL:www.fargobluesfest.com*
7. (0.400) Dan's Police Page URL:danspolice.8m.com	7. (0.302) Google Search: URL:www.google.com/search	7. (0.423) stantonanderson.com URL:www.stantonanderson.com
8. (0.319) **Blues On Stage, your complete** **URL:www.mnblues.com**	8. (0.292) Delta Blues - DeltaBlues - delt URL:www.deltablues.com	8. (0.400) Dan's Police Page URL:danspolice.8m.com
9. (0.315) **A Cyber Blues Society for mus** **URL:www.bluessociety.net**	9. (0.287) **Blues On Stage, your complete** **URL:www.mnblues.com**	9. (0.386) **Blues On Stage, your complete** **URL:www.mnblues.com**
10. (0.299) **BLUES WORLD** **URL:www.bluesworld.com**	10. (0.276) BluesSociety.net - Free Email S URL:mail.bluessociety.net	10. (0.342) **BluesSociety.net - for music** **URL:sitebuilder.bluessociety.net**

AT-avg	Norm	BFS
1. (1.000) **The Blues Foundation. Your** **URL:www.blues.org**	1. (1.000) **The Blues Foundation. Your** **URL:www.blues.org**	1. (1.000) *Harry's Blues Lyrics Online, Home* *URL:blueslinks.tripod.com*
2. (0.789) *Harry's Blues Lyrics Online, Home* *URL:blueslinks.tripod.com*	2. (0.594) **A Cyber Blues Society for mus** **URL:www.bluessociety.net**	2. (0.958) *home* *URL:www.fargobluesfest.com*
3. (0.608) **The Blue Highway** **URL:www.thebluehighway.com**	3. (0.533) WebRing: addsite_login URL:l.webring.com/wrman?ring=blues	3. (0.954) Delta Blues - DeltaBlues - delt URL:www.deltablues.com
4. (0.589) *home* *URL:www.fargobluesfest.com*	4. (0.515) **BluesSociety.net - for musici** **URL:sitebuilder.bluessociety.net**	4. (0.952) **The Blues Foundation. Your** **URL:www.blues.org**
5. (0.582) stantonanderson.com URL:www.stantonanderson.com	5. (0.502) Google Search: URL:www.google.com/search	5. (0.935) stantonanderson.com URL:www.stantonanderson.com
6. (0.566) Delta Blues - DeltaBlues - delt URL:www.deltablues.com	6. (0.476) BluesSociety.net - Free Email S URL:mail.bluessociety.net	6. (0.927) Dan's Police Page URL:danspolice.8m.com
7. (0.548) Dan's Police Page URL:danspolice.8m.com	7. (0.473) Your Mailinglist Provider URL:www.yourmailinglistprovider.co	7. (0.917) **The Blue Highway** **URL:www.thebluehighway.com**
8. (0.521) **A Cyber Blues Society for mus** **URL:www.bluessociety.net**	8. (0.466) **A Cyber Blues Society Blues L** **URL:www.bluessociety.net/links03.h**	8. (0.906) Empty title field URL:www.letthegoodtimesroll.com
9. (0.478) **Blues On Stage, your complete** **URL:www.mnblues.com**	9. (0.466) **A Cyber Blues Society Blues L** **URL:www.bluessociety.net/links05.h**	9. (0.898) *Dallas Blues Society Records* *URL:www.dallasbluessociety.com*
10. (0.438) **Natchel' Blues Network** **URL:www.natchelblues.org**	10. (0.466) **Blues Biographies - Artist o** **URL:www.bluessociety.net/greats.**	10. (0.883) Paul Pelletier - Book Publisher URL:www.brightguy.demon.co.uk

Table C.9: Query "blues"

Hits	PageRank	InDegree
1. (1.000) nbsp; nbsp; nbsp; nbsp; (ca URL:caffeinediary.blogspot.com	1. (1.000) nbsp; nbsp; nbsp; nbsp; (ca URL:caffeinediary.blogspot.com	1. (1.000) nbsp; nbsp; nbsp; nbsp; (ca URL:caffeinediary.blogspot.com
2. (0.874) wrongwaygoback : dynamic ribbon URL:www.wrongwaygoback.com	2. (0.993) iGourmet.com Customer Reviews URL:www.bizrate.com/merchant/repor	2. (0.761) *800cheesecake.com* *URL:www.800cheesecake.com*
3. (0.861) Boing Boing: A Directory of Won URL:www.boingboing.net	3. (0.977) Southern U.S. Cuisine Classifie URL:adlistings.southernfood.about.	3. (0.685) Coming Soon... URL:www.cheesegiftbasket.net
4. (0.847) movabletype.org URL:www.movabletype.org	4. (0.903) movabletype.org URL:www.movabletype.org	4. (0.674) *Cheese of the month club — cheese* *URL:www.cheeseexpress.com*
5. (0.846) Izzle! Izzle pfaff! URL:www.izzlepfaff.com	5. (0.888) *800cheesecake.com* *URL:www.800cheesecake.com*	5. (0.674) movabletype.org URL:www.movabletype.org
6. (0.843) Parasyte: Insanity of the Mind URL:parasyte.pitas.com	6. (0.842) Gawker URL:www.gawker.com	6. (0.614) Ethnic art : African dance and URL:www.ethnicarts.org
7. (0.823) harrumph! still crazy. URL:www.harrumph.com	7. (0.742) BBBOnLine Seal Verification URL:www.bbbonline.org/cks.asp?id=1	7. (0.582) Half Moon Bay Bed and Breakfast URL:www.millroseinn.com
8. (0.815) guestofbeth.diaryland.com URL:guestofbeth.diaryland.com	8. (0.726) GRILLEDCHEESE (.) C URL:www.grilledcheese.com	8. (0.582) Snowboard Boots for sale at Sno URL:www.snowboard-boots.com
9. (0.815) ::: wood s lot ::: "fictive th URL:www.ncf.ca/∼ek867/wood_s_lot.h	9. (0.697) Luminee : Northern California W URL:www.luminee.com	9. (0.560) *Empty title field* *URL:www.cheese-express.com*
10. (0.814) Caterina.net URL:caterina.net	10. (0.684) *Cheese of the month club — cheese* *URL:www.cheeseexpress.com*	10. (0.538) Say Cheese - SayCheese - sayche URL:www.saycheese.net

HubAvg	Max	AT-med
1. (1.000) Coming Soon... URL:www.cheesegiftbasket.net	1. (1.000) nbsp; nbsp; nbsp; nbsp; (ca URL:caffeinediary.blogspot.com	1. (1.000) nbsp; nbsp; nbsp; nbsp; (ca URL:caffeinediary.blogspot.com
2. (0.942) Ethnic art : African dance and URL:www.ethnicarts.org	2. (0.559) movabletype.org URL:www.movabletype.org	2. (0.595) movabletype.org URL:www.movabletype.org
3. (0.934) Snowboard Boots for sale at Sno URL:www.snowboard-boots.com	3. (0.419) wrongwaygoback : dynamic ribbon URL:www.wrongwaygoback.com	3. (0.457) wrongwaygoback : dynamic ribbon URL:www.wrongwaygoback.com
4. (0.931) Half Moon Bay Bed and Breakfast URL:www.millroseinn.com	4. (0.371) Boing Boing: A Directory of Won URL:www.boingboing.net	4. (0.403) Izzle! Izzle pfaff! URL:www.izzlepfaff.com
5. (0.911) *Empty title field* *URL:www.cheese-express.com*	5. (0.371) Izzle! Izzle pfaff! URL:www.izzlepfaff.com	5. (0.400) Boing Boing: A Directory of Won URL:www.boingboing.net
6. (0.236) Empty title field URL:www.santacruzwebdesign.com	6. (0.358) guestofbeth.diaryland.com URL:guestofbeth.diaryland.com	6. (0.389) guestofbeth.diaryland.com URL:guestofbeth.diaryland.com
7. (0.235) Small business merchant account URL:www.ikorb.com	7. (0.325) Parasyte: Insanity of the Mind URL:parasyte.pitas.com	7. (0.345) Parasyte: Insanity of the Mind URL:parasyte.pitas.com
8. (0.227) Cookie gifts : cookie delivery URL:www.pacificcookie.com	8. (0.272) harrumph! still crazy. URL:www.harrumph.com	8. (0.299) harrumph! still crazy. URL:www.harrumph.com
9. (0.216) Bushrods BBQ Equipment URL:www.bushrods.com	9. (0.270) kottke.org :: home of fine hype URL:kottke.org	9. (0.298) kottke.org :: home of fine hype URL:kottke.org
10. (0.212) b.firm Skin Care Products - Try URL:www.tobfirm.com	10. (0.267) anil dash - New York Still Love URL:dashes.com/anil	10. (0.295) anil dash - New York Still Love URL:dashes.com/anil

AT-avg	Norm	BFS
1. (1.000) Coming Soon... URL:www.cheesegiftbasket.net	1. (1.000) nbsp; nbsp; nbsp; nbsp; (ca URL:caffeinediary.blogspot.com	1. (1.000) *Cheese of the month club — cheese* *URL:www.cheeseexpress.com*
2. (0.985) Ethnic art : African dance and URL:www.ethnicarts.org	2. (0.664) movabletype.org URL:www.movabletype.org	2. (0.940) *800cheesecake.com* *URL:www.800cheesecake.com*
3. (0.968) Snowboard Boots for sale at Sno URL:www.snowboard-boots.com	3. (0.596) wrongwaygoback : dynamic ribbon URL:www.wrongwaygoback.com	3. (0.881) Say Cheese - SayCheese - sayche URL:www.saycheese.net
4. (0.961) Half Moon Bay Bed and Breakfast URL:www.millroseinn.com	4. (0.551) Boing Boing: A Directory of Won URL:www.boingboing.net	4. (0.858) floor4.org URL:floor4.org
5. (0.925) *Empty title field* *URL:www.cheese-express.com*	5. (0.543) Izzle! Izzle pfaff! URL:www.izzlepfaff.com	5. (0.843) nbsp; nbsp; nbsp; nbsp; (ca URL:caffeinediary.blogspot.com
6. (0.363) Empty title field URL:www.santacruzwebdesign.com	6. (0.519) guestofbeth.diaryland.com URL:guestofbeth.diaryland.com	6. (0.811) *littlebarnfarm.com* *URL:www.littlebarnfarm.com*
7. (0.359) Small business merchant account URL:www.ikorb.com	7. (0.508) Parasyte: Insanity of the Mind URL:parasyte.pitas.com	7. (0.793) **CHEESE.COM - All about** **URL:www.cheese.com**
8. (0.350) Cookie gifts : cookie delivery URL:www.pacificcookie.com	8. (0.466) harrumph! still crazy. URL:www.harrumph.com	8. (0.789) Cheese Racing URL:www.cheeseracing.org
9. (0.333) Bushrods BBQ Equipment URL:www.bushrods.com	9. (0.456) anil dash - New York Still Love URL:dashes.com/anil	9. (0.755) Ask Jesus - askjesus.8k.com URL:www.askjesus.8k.com
10. (0.329) b.firm Skin Care Products - Try URL:www.tobfirm.com	10. (0.451) Travelers Diagram...an apprecia URL:www.travelersdiagram.com	10. (0.753) *Entertaining* *URL:cheese.about.com*

Table C.10: Query "cheese"

Hits	PageRank	InDegree
1. (1.000) **earlyromanticguitar.com** **URL:www.earlyromanticguitar.com**	1. (1.000) Take Note Publishing Limited fo URL:www.takenote.co.uk	1. (1.000) Guitar Alive - GuitarAlive - gu URL:www.guitaralive.com
2. (0.927) **Empty title field** **URL:classicalguitar.freehosting.ne**	2. (0.724) *Guitar music, guitar books and so* *URL:www.booksforguitar.com*	2. (0.895) **earlyromanticguitar.com** **URL:www.earlyromanticguitar.com**
3. (0.889) *Adirondack Spruce.com* *URL:adirondackspruce.com*	3. (0.588) **Registry of Guitar Tutors** **URL:www.registryofguitartutors.com**	3. (0.889) **Empty title field** **URL:www.guitarfoundation.org**
4. (0.766) *The Classical Guitar Homepage of* *URL:www.ak-c.demon.nl*	4. (0.528) Guitar Alive - GuitarAlive - gu URL:www.guitaralive.com	4. (0.882) *Hitsquad.com - Musicians Web* *URL:www.hitsquad.com*
5. (0.732) Guitar Alive - GuitarAlive - gu URL:www.guitaralive.com	5. (0.416) *Hitsquad.com - Musicians Web* *URL:www.hitsquad.com*	5. (0.850) Hitsquad Privacy Policy URL:www.hitsquad.com/privacy.shtml
6. (0.681) **Empty title field** **URL:www.guitarfoundation.org**	6. (0.413) Hitsquad Privacy Policy URL:www.hitsquad.com/privacy.shtml	6. (0.850) Advertising on Hitsquad Music I URL:www.hitsquad.com/advertising.s
7. (0.676) *GUITAR REVIEW* *URL:www.guitarreview.com*	7. (0.413) Advertising on Hitsquad Music I URL:www.hitsquad.com/advertising.s	7. (0.784) *Adirondack Spruce.com* *URL:adirondackspruce.com*
8. (0.644) *Avi Afriat - Classical guitar hom* *URL:afriat.tripod.com*	8. (0.387) **Empty title field** **URL:www.guitarfoundation.org**	8. (0.778) **Empty title field** **URL:classicalguitar.freehosting.ne**
9. (0.605) **The Classical Guitar Home Pag** **URL:www.guitarist.com/cg/cg.htm**	9. (0.343) **Guitar Foundation of America:** **URL:64.78.54.231**	9. (0.765) *Empty title field* *URL:www.vicnet.net.au/∼easyjamn*
10. (0.586) *Empty title field* *URL:www.duolenz.com*	10. (0.322) Vivendi Universal URL:www.vivendiuniversal.com	10. (0.739) *The Classical Guitar Homepage of* *URL:www.ak-c.demon.nl*

HubAvg	Max	AT-med
1. (1.000) *Hitsquad.com - Musicians Web* *URL:www.hitsquad.com*	1. (1.000) Guitar Alive - GuitarAlive - gu URL:www.guitaralive.com	1. (1.000) *Hitsquad.com - Musicians Web* *URL:www.hitsquad.com*
2. (0.995) Hitsquad Privacy Policy URL:www.hitsquad.com/privacy.shtml	2. (0.619) **earlyromanticguitar.com** **URL:www.earlyromanticguitar.com**	2. (0.983) Hitsquad Privacy Policy URL:www.hitsquad.com/privacy.shtml
3. (0.995) Advertising on Hitsquad Music I URL:www.hitsquad.com/advertising.s	3. (0.506) **Empty title field** **URL:www.guitarfoundation.org**	3. (0.983) Advertising on Hitsquad Music I URL:www.hitsquad.com/advertising.s
4. (0.856) *Empty title field* *URL:www.vicnet.net.au/∼easyjamn*	4. (0.451) *Adirondack Spruce.com* *URL:adirondackspruce.com*	4. (0.880) *Empty title field* *URL:www.vicnet.net.au/∼easyjamn*
5. (0.135) *AMG All Music Guide* *URL:www.allmusic.com*	5. (0.441) **Empty title field** **URL:classicalguitar.freehosting.ne**	5. (0.205) *AMG All Music Guide* *URL:www.allmusic.com*
6. (0.132) Free Music Download, MP3 Music, URL:ubl.com	6. (0.378) *GUITAR REVIEW* *URL:www.guitarreview.com*	6. (0.193) Free Music Download, MP3 Music, URL:ubl.com
7. (0.130) *2000 Guitars Database* *URL:dargo.vicnet.net.au/guitar/lis*	7. (0.377) *The Classical Guitar Homepage of* *URL:www.ak-c.demon.nl*	7. (0.169) *2000 Guitars Database* *URL:dargo.vicnet.net.au/guitar/lis*
8. (0.115) Guitar Alive - GuitarAlive - gu URL:www.guitaralive.com	8. (0.371) **The Classical Guitar Home Pag** **URL:www.guitarist.com/cg/cg.htm**	8. (0.132) Guitar Alive - GuitarAlive - gu URL:www.guitaralive.com
9. (0.096) CDNOW URL:www.cdnow.com/from=sr-767167	9. (0.336) *Hitsquad.com - Musicians Web* *URL:www.hitsquad.com*	9. (0.129) CDNOW URL:www.cdnow.com/from=sr-767167
10. (0.056) *OLGA - The On-Line Guitar* *URL:www.olga.net*	10. (0.312) Hitsquad Privacy Policy URL:www.hitsquad.com/privacy.shtml	10. (0.082) *OLGA - The On-Line Guitar* *URL:www.olga.net*

AT-avg	Norm	BFS
1. (1.000) *Hitsquad.com - Musicians Web* *URL:www.hitsquad.com*	1. (1.000) *Hitsquad.com - Musicians Web* *URL:www.hitsquad.com*	1. (1.000) **Empty title field** **URL:classicalguitar.freehosting.ne**
2. (0.986) Hitsquad Privacy Policy URL:www.hitsquad.com/privacy.shtml	2. (0.979) Hitsquad Privacy Policy URL:www.hitsquad.com/privacy.shtml	2. (0.991) **earlyromanticguitar.com** **URL:www.earlyromanticguitar.com**
3. (0.986) Advertising on Hitsquad Music I URL:www.hitsquad.com/advertising.s	3. (0.979) Advertising on Hitsquad Music I URL:www.hitsquad.com/advertising.s	3. (0.974) *Adirondack Spruce.com* *URL:adirondackspruce.com*
4. (0.906) *Empty title field* *URL:www.vicnet.net.au/∼easyjamn*	4. (0.889) *Empty title field* *URL:www.vicnet.net.au/∼easyjamn*	4. (0.962) **Empty title field** **URL:www.guitarfoundation.org**
5. (0.210) *AMG All Music Guide* *URL:www.allmusic.com*	5. (0.218) *AMG All Music Guide* *URL:www.allmusic.com*	5. (0.945) Guitar Alive - GuitarAlive - gu URL:www.guitaralive.com
6. (0.199) Free Music Download, MP3 Music, URL:ubl.com	6. (0.202) Free Music Download, MP3 Music, URL:ubl.com	6. (0.933) *The Classical Guitar Homepage of* *URL:www.ak-c.demon.nl*
7. (0.179) *2000 Guitars Database* *URL:dargo.vicnet.net.au/guitar/lis*	7. (0.185) Guitar Alive - GuitarAlive - gu URL:www.guitaralive.com	7. (0.917) **The Classical Guitar Home Pag** **URL:www.guitarist.com/cg/cg.htm**
8. (0.135) CDNOW URL:www.cdnow.com/from=sr-767167	8. (0.172) *2000 Guitars Database* *URL:dargo.vicnet.net.au/guitar/lis*	8. (0.898) *Avi Afriat - Classical guitar hom* *URL:afriat.tripod.com*
9. (0.122) Guitar Alive - GuitarAlive - gu URL:www.guitaralive.com	9. (0.130) CDNOW URL:www.cdnow.com/from=sr-767167	9. (0.889) *GUITAR REVIEW* *URL:www.guitarreview.com*
10. (0.080) *OLGA - The On-Line Guitar* *URL:www.olga.net*	10. (0.118) **earlyromanticguitar.com** **URL:www.earlyromanticguitar.com**	10. (0.881) *Empty title field* *URL:www.duolenz.com*

Table C.11: Query “classical guitar”

Hits	PageRank	InDegree
1. (1.000) Ziff Davis Media — Home URL:www.ziffdavis.com	1. (1.000) Manchester Metropolitan Univers URL:www.mmu.ac.uk	1. (1.000) *SFI Home Page* *URL:www.santafe.edu*
2. (0.998) Ziff Davis Media — Privacy Poli URL:www.ziffdavis.com/terms/index.	2. (0.727) *SFI Home Page* *URL:www.santafe.edu*	2. (0.863) **ECCC - The Electronic** **URL:eccc.uni-trier.de/eccc**
3. (0.998) Ziff Davis Media — Privacy Poli URL:www.ziffdavis.com/terms/index.	3. (0.703) **Complexity Digest** **URL:www.comdig.org**	3. (0.601) Ziff Davis Media — Home URL:www.ziffdavis.com
4. (0.950) :::::::::::::::::::: Registrati URL:webevents.broadcast.com/ziffda	4. (0.685) **ECCC - The Electronic** **URL:eccc.uni-trier.de/eccc**	4. (0.595) *Artificial Life VIII The 8th Inte* *URL:alife8.alife.org*
5. (0.950) :::::::::::::::::::: Registrati URL:webevents.broadcast.com/ziffda	5. (0.653) Holistic Politics URL:www.holisticpolitics.com	5. (0.583) Ziff Davis Media — Privacy Poli URL:www.ziffdavis.com/terms/index.
6. (0.950) :::::::::::::::::::: Registrati URL:webevents.broadcast.com/ziffda	6. (0.639) *Artificial Life VIII The 8th Inte* *URL:alife8.alife.org*	6. (0.583) Ziff Davis Media — Privacy Poli URL:www.ziffdavis.com/terms/index.
7. (0.934) eWeek Research Library: Wireles URL:eweek.bitpipe.com/data/rlist?t	7. (0.565) Google URL:www.google.com	7. (0.577) Google URL:www.google.com
8. (0.934) eWeek Research Library: Wireles URL:eweek.bitpipe.com/data/detail?	8. (0.563) KMNetwork: World's most reputed URL:www.kmnetwork.com	8. (0.554) **Complexity Digest** **URL:www.comdig.org**
9. (0.934) eWeek Research Library: How to URL:eweek.bitpipe.com/data/detail?	9. (0.551) Business, Technology, and Knowl URL:www.kmnetwork.com/ken/jobs.htm	9. (0.470) *Empty title field* *URL:www.ams.org*
10. (0.934) eWeek Research Library: The CIO URL:eweek.bitpipe.com/data/detail?	10. (0.530) **Fractal 2004: International** **URL:www.kingston.ac.uk/fractal**	10. (0.440) :::::::::::::::::::: Registrati URL:webevents.broadcast.com/ziffda

HubAvg	Max	AT-med
1. (1.000) Ziff Davis Media — Home URL:www.ziffdavis.com	1. (1.000) *SFI Home Page* *URL:www.santafe.edu*	1. (1.000) *SFI Home Page* *URL:www.santafe.edu*
2. (0.981) Ziff Davis Media — Privacy Poli URL:www.ziffdavis.com/terms/index.	2. (0.325) **New England Complex** **URL:necsi.org**	2. (0.489) **ECCC - The Electronic** **URL:eccc.uni-trier.de/eccc**
3. (0.981) Ziff Davis Media — Privacy Poli URL:www.ziffdavis.com/terms/index.	3. (0.304) **ECCC - The Electronic** **URL:eccc.uni-trier.de/eccc**	3. (0.362) **New England Complex** **URL:necsi.org**
4. (0.894) :::::::::::::::::::: Registrati URL:webevents.broadcast.com/ziffda	4. (0.251) **Complexity Digest** **URL:www.comdig.org**	4. (0.338) **Complexity Digest** **URL:www.comdig.org**
5. (0.894) :::::::::::::::::::: Registrati URL:webevents.broadcast.com/ziffda	5. (0.234) *Artificial Life VIII The 8th Inte* *URL:alife8.alife.org*	5. (0.321) *Artificial Life VIII The 8th Inte* *URL:alife8.alife.org*
6. (0.894) :::::::::::::::::::: Registrati URL:webevents.broadcast.com/ziffda	6. (0.216) **The Complexity and Artificial** **URL:www.calresco.org**	6. (0.288) Google URL:www.google.com
7. (0.866) Ziff Davis Media — About URL:www.ziffdavis.com/about/index.	7. (0.205) Google URL:www.google.com	7. (0.248) **The Complexity and Artificial** **URL:www.calresco.org**
8. (0.835) eWeek Research Library: Wireles URL:eweek.bitpipe.com/data/rlist?t	8. (0.186) **Complexity, Self Adaptive** **URL:www.brint.com/Systems.htm**	8. (0.241) *The Collection of Computer Scienc* *URL:liinwww.ira.uka.de/bibliograph*
9. (0.835) eWeek Research Library: Wireles URL:eweek.bitpipe.com/data/detail?	9. (0.186) **CCSR Homepage** **URL:www.ccsr.uiuc.edu**	9. (0.222) *Empty title field* *URL:www.ams.org*
10. (0.835) eWeek Research Library: How to URL:eweek.bitpipe.com/data/detail?	10. (0.182) **Emergence - A Journal of** **URL:www.emergence.org**	10. (0.211) **Complexity, Self Adaptive** **URL:www.brint.com/Systems.htm**

AT-avg	Norm	BFS
1. (1.000) Ziff Davis Media — Home URL:www.ziffdavis.com	1. (1.000) Ziff Davis Media — Home URL:www.ziffdavis.com	1. (1.000) *SFI Home Page* *URL:www.santafe.edu*
2. (0.989) Ziff Davis Media — Privacy Poli URL:www.ziffdavis.com/terms/index.	2. (0.991) Ziff Davis Media — Privacy Poli URL:www.ziffdavis.com/terms/index.	2. (0.921) **ECCC - The Electronic** **URL:eccc.uni-trier.de/eccc**
3. (0.989) Ziff Davis Media — Privacy Poli URL:www.ziffdavis.com/terms/index.	3. (0.991) Ziff Davis Media — Privacy Poli URL:www.ziffdavis.com/terms/index.	3. (0.854) Google URL:www.google.com
4. (0.773) :::::::::::::::::::: Registrati URL:webevents.broadcast.com/ziffda	4. (0.872) :::::::::::::::::::: Registrati URL:webevents.broadcast.com/ziffda	4. (0.832) *The Collection of Computer Scienc* *URL:liinwww.ira.uka.de/bibliograph*
5. (0.773) :::::::::::::::::::: Registrati URL:webevents.broadcast.com/ziffda	5. (0.872) :::::::::::::::::::: Registrati URL:webevents.broadcast.com/ziffda	5. (0.812) **Complexity, Self Adaptive** **URL:www.brint.com/Systems.htm**
6. (0.773) :::::::::::::::::::: Registrati URL:webevents.broadcast.com/ziffda	6. (0.872) :::::::::::::::::::: Registrati URL:webevents.broadcast.com/ziffda	6. (0.805) **New England Complex** **URL:necsi.org**
7. (0.754) Ziff Davis Media — About URL:www.ziffdavis.com/about/index.	7. (0.848) Ziff Davis Media — About URL:www.ziffdavis.com/about/index.	7. (0.771) **The Complexity and Artificial** **URL:www.calresco.org**
8. (0.731) eWeek Research Library: Wireles URL:eweek.bitpipe.com/data/rlist?t	8. (0.843) eWeek Research Library: Wireles URL:eweek.bitpipe.com/data/rlist?t	8. (0.768) **Complexity International** **URL:www.csu.edu.au/ci**
9. (0.731) eWeek Research Library: Wireles URL:eweek.bitpipe.com/data/detail?	9. (0.843) eWeek Research Library: Wireles URL:eweek.bitpipe.com/data/detail?	9. (0.766) Computer Science Papers NEC Res URL:citeseer.nj.nec.com/cs
10. (0.731) eWeek Research Library: How to URL:eweek.bitpipe.com/data/detail?	10. (0.843) eWeek Research Library: How to URL:eweek.bitpipe.com/data/detail?	10. (0.762) *Artificial Life VIII The 8th Inte* *URL:alife8.alife.org*

Table C.12: Query "complexity"

Hits	PageRank	InDegree
1. (1.000) **ECCC - The Electronic** **URL:eccc.uni-trier.de/eccc**	1. (1.000) **ECCC - The Electronic** **URL:eccc.uni-trier.de/eccc**	1. (1.000) **ECCC - The Electronic** **URL:eccc.uni-trier.de/eccc**
2. (0.323) *ACM: Association for Computing* *URL:www.acm.org*	2. (0.985) *My Computational Complexity Web* *URL:www.fortnow.com/lance/complog*	2. (0.517) *My Computational Complexity Web* *URL:www.fortnow.com/lance/complog*
3. (0.288) *The Electronic Journal of Combina* *URL:www.combinatorics.org*	3. (0.877) **Computational Complexity** **URL:computationalcomplexity.org**	3. (0.415) *Springer Link - Publication* *URL:link.springer-ny.com/link/serv*
4. (0.287) *Center for Discrete Mathematics a* *URL:dimacs.rutgers.edu*	4. (0.635) *European Association for Theoreti* *URL:www.eatcs.org*	4. (0.305) *ACM: Association for Computing* *URL:www.acm.org*
5. (0.273) *Springer Link - Publication* *URL:link.springer-ny.com/link/serv*	5. (0.570) Welcome to Springer, springer-v URL:www.springer.de	5. (0.297) *The Electronic Journal of Combina* *URL:www.combinatorics.org*
6. (0.205) *IEEE Computer Society* *URL:computer.org*	6. (0.564) **Volume on Computational** **URL:www.c3.lanl.gov/∼percus/volume**	6. (0.271) **IEEE Conference on Comp** **URL:cs.utep.edu/longpre/complexity**
7. (0.202) *European Association for Theoreti* *URL:www.eatcs.org*	7. (0.542) *The Hyper Bulletin of the EATCS* *URL:www.liacs.nl/∼beatcs*	7. (0.263) *Center for Discrete Mathematics a* *URL:dimacs.rutgers.edu*
8. (0.197) **Complexity People** **URL:eccc.uni-trier.de/eccc/info/pe**	8. (0.480) *Springer Link - Publication* *URL:link.springer-ny.com/link/serv*	8. (0.229) Computer Science Papers NEC URL:citeseer.nj.nec.com/cs
9. (0.163) **IEEE Conference on Comp** **URL:cs.utep.edu/longpre/complexity**	9. (0.474) gillespiefox web design URL:www.gillespiefox.com	9. (0.229) *IEEE Computer Society* *URL:computer.org*
10. (0.157) Computer Science Papers NEC URL:citeseer.nj.nec.com/cs	10. (0.455) University of Illinois URL:www.uiuc.edu	10. (0.220) **Computational Complexity** **URL:computationalcomplexity.org**

HubAvg	Max	AT-med
1. (1.000) **ECCC - The Electronic** **URL:eccc.uni-trier.de/eccc**	1. (1.000) **ECCC - The Electronic** **URL:eccc.uni-trier.de/eccc**	1. (1.000) **ECCC - The Electronic** **URL:eccc.uni-trier.de/eccc**
2. (0.771) *My Computational Complexity Web* *URL:www.fortnow.com/lance/complog*	2. (0.225) *Springer Link - Publication* *URL:link.springer-ny.com/link/serv*	2. (0.249) *Springer Link - Publication* *URL:link.springer-ny.com/link/serv*
3. (0.196) *Springer Link - Publication* *URL:link.springer-ny.com/link/serv*	3. (0.219) *ACM: Association for Computing* *URL:www.acm.org*	3. (0.244) *ACM: Association for Computing* *URL:www.acm.org*
4. (0.149) *The Electronic Journal of Combina* *URL:www.combinatorics.org*	4. (0.210) *The Electronic Journal of Combina* *URL:www.combinatorics.org*	4. (0.238) *The Electronic Journal of Combina* *URL:www.combinatorics.org*
5. (0.131) *ACM: Association for Computing* *URL:www.acm.org*	5. (0.195) *Center for Discrete Mathematics a* *URL:dimacs.rutgers.edu*	5. (0.217) *Center for Discrete Mathematics a* *URL:dimacs.rutgers.edu*
6. (0.128) *Center for Discrete Mathematics a* *URL:dimacs.rutgers.edu*	6. (0.142) **Complexity People** **URL:eccc.uni-trier.de/eccc/info/pe**	6. (0.160) *IEEE Computer Society* *URL:computer.org*
7. (0.113) **CC Published by Birkhauml;us** **URL:www.birkhauser.ch/journals/370**	7. (0.138) *IEEE Computer Society* *URL:computer.org*	7. (0.154) **Complexity People** **URL:eccc.uni-trier.de/eccc/info/pe**
8. (0.108) **IEEE Conference on Comp** **URL:cs.utep.edu/longpre/complexity**	8. (0.135) *European Association for Theoreti* *URL:www.eatcs.org*	8. (0.145) Computer Science Papers NEC URL:citeseer.nj.nec.com/cs
9. (0.103) Computer Science Papers NEC URL:citeseer.nj.nec.com/cs	9. (0.134) Computer Science Papers NEC URL:citeseer.nj.nec.com/cs	9. (0.144) *European Association for Theoreti* *URL:www.eatcs.org*
10. (0.085) **Complexity People** **URL:eccc.uni-trier.de/eccc/info/**	10. (0.125) **IEEE Conference on Comp** **URL:cs.utep.edu/longpre/complexi**	10. (0.136) **IEEE Conference on Comp** **URL:cs.utep.edu/longpre/complexi**

AT-avg	Norm	BFS
1. (1.000) **ECCC - The Electronic** **URL:eccc.uni-trier.de/eccc**	1. (1.000) **ECCC - The Electronic** **URL:eccc.uni-trier.de/eccc**	1. (1.000) **ECCC - The Electronic** **URL:eccc.uni-trier.de/eccc**
2. (0.265) *ACM: Association for Computing* *URL:www.acm.org*	2. (0.235) *Springer Link - Publication* *URL:link.springer-ny.com/link/serv*	2. (0.715) *Lance Fortnow* *URL:www.neci.nj.nec.com/homepages/*
3. (0.262) *Springer Link - Publication* *URL:link.springer-ny.com/link/serv*	3. (0.232) *ACM: Association for Computing* *URL:www.acm.org*	3. (0.698) *Springer Link - Publication* *URL:link.springer-ny.com/link/serv*
4. (0.255) *The Electronic Journal of Combina* *URL:www.combinatorics.org*	4. (0.223) *The Electronic Journal of Combina* *URL:www.combinatorics.org*	4. (0.648) Computer Science Papers NEC URL:citeseer.nj.nec.com/cs
5. (0.236) *Center for Discrete Mathematics a* *URL:dimacs.rutgers.edu*	5. (0.208) *Center for Discrete Mathematics a* *URL:dimacs.rutgers.edu*	5. (0.642) *The Electronic Journal of Combina* *URL:www.combinatorics.org*
6. (0.175) *IEEE Computer Society* *URL:computer.org*	6. (0.149) *IEEE Computer Society* *URL:computer.org*	6. (0.639) *Jiri Sgall* *URL:www.math.cas.cz/∼sgall*
7. (0.164) **Complexity People** **URL:eccc.uni-trier.de/eccc/info/pe**	7. (0.149) **Complexity People** **URL:eccc.uni-trier.de/eccc/info/pe**	7. (0.639) *ACM: Association for Computing* *URL:www.acm.org*
8. (0.156) *European Association for Theoreti* *URL:www.eatcs.org*	8. (0.142) *European Association for Theoreti* *URL:www.eatcs.org*	8. (0.638) *My Computational Complexity Web* *URL:www.fortnow.com/lance/complog*
9. (0.150) Computer Science Papers NEC URL:citeseer.nj.nec.com/cs	9. (0.139) Computer Science Papers NEC URL:citeseer.nj.nec.com/cs	9. (0.636) *Paul Beame* *URL:www.cs.washington.edu/homes/be*
10. (0.142) **IEEE Conference on Comp** **URL:cs.utep.edu/longpre/complexi**	10. (0.132) **IEEE Conference on Comp** **URL:cs.utep.edu/longpre/complexi**	10. (0.632) **Complexity People** **URL:eccc.uni-trier.de/eccc/info/**

Table C.13: Query “computational complexity”

Hits	PageRank	InDegree
1. (1.000) *Geometry in Action* *URL:www.ics.uci.edu/∼eppstein/geom*	1. (1.000) University of California, Irvin URL:www.uci.edu	1. (1.000) *Geometry in Action* *URL:www.ics.uci.edu/∼eppstein/geom*
2. (0.745) **The former CGAL home page** **URL:www.cs.uu.nl/CGAL**	2. (0.719) BertelsmannSpringer Science+Bus URL:www.bertelsmannspringer.de	2. (0.681) **The former CGAL home page** **URL:www.cs.uu.nl/CGAL**
3. (0.665) **The Geometry Junkyard** **URL:www.ics.uci.edu/∼eppstein/junk**	3. (0.702) **The CGAL Home Page** **URL:www.cgal.org**	3. (0.632) *ACM: Association for Computing* *URL:www.acm.org*
4. (0.618) *David Eppstein* *URL:www.ics.uci.edu/∼eppstein*	4. (0.663) *ACM: Association for Computing* *URL:www.acm.org*	4. (0.618) WebCT.com URL:www.webct.com
5. (0.608) **Computational Geometry** **URL:www.scs.carleton.ca/∼csgs/reso**	5. (0.537) Validation Results URL:validator.w3.org/check/referer	5. (0.590) DREXEL UNIVERSITY URL:www.drexel.edu
6. (0.556) *Joseph O'Rourke* *URL:cs.smith.edu/∼orourke*	6. (0.517) Google URL:www.google.com	6. (0.590) A Virtual Math Community URL:www.drexel.edu/ia/mathforum
7. (0.529) *Springer Link - Publication* *URL:link.springer.de/link/service/*	7. (0.472) *Geometry in Action* *URL:www.ics.uci.edu/∼eppstein/geom*	7. (0.535) *Springer Link - Publication* *URL:link.springer.de/link/service/*
8. (0.516) *LEDA moved to Algorithmic Sol* *URL:www.mpi-sb.mpg.de/LEDA/leda.ht*	8. (0.460) **ScienceDirect - Computational** **URL:www.sciencedirect.com/science/**	8. (0.472) **Computational Geometry** **URL:www.scs.carleton.ca/∼csgs/reso**
9. (0.452) **The compgeom mailing lists** **URL:netlib.bell-labs.com/netlib/co**	9. (0.429) DREXEL UNIVERSITY URL:www.drexel.edu	9. (0.458) **Computational Geometry** **URL:www.elsevier.nl/locate/comgeo**
10. (0.446) *Gnter M. Ziegler* *URL:www.math.tu-berlin.de/∼ziegler*	10. (0.429) A Virtual Math Community URL:www.drexel.edu/ia/mathforum	10. (0.458) **The Geometry Junkyard** **URL:www.ics.uci.edu/∼eppstein/ju**

HubAvg	Max	AT-med
1. (1.000) WebCT.com URL:www.webct.com	1. (1.000) *Geometry in Action* *URL:www.ics.uci.edu/∼eppstein/geom*	1. (1.000) WebCT.com URL:www.webct.com
2. (0.944) DREXEL UNIVERSITY URL:www.drexel.edu	2. (0.459) **The former CGAL home page** **URL:www.cs.uu.nl/CGAL**	2. (0.986) DREXEL UNIVERSITY URL:www.drexel.edu
3. (0.944) A Virtual Math Community URL:www.drexel.edu/ia/mathforum	3. (0.377) **The Geometry Junkyard** **URL:www.ics.uci.edu/∼eppstein/junk**	3. (0.986) A Virtual Math Community URL:www.drexel.edu/ia/mathforum
4. (0.129) *Geometry in Action* *URL:www.ics.uci.edu/∼eppstein/geom*	4. (0.346) **Computational Geometr** **URL:www.scs.carleton.ca/∼csgs/reso**	4. (0.467) *Geometry in Action* *URL:www.ics.uci.edu/∼eppstein/geom*
5. (0.095) **Computational Geometry** **URL:www.scs.carleton.ca/∼csgs/reso**	5. (0.283) **Computational Geometry** **URL:compgeom.cs.uiuc.edu/∼jeffe/co**	5. (0.257) **Computational Geometry** **URL:www.scs.carleton.ca/∼csgs/reso**
6. (0.094) **The CGAL Home Page** **URL:www.cgal.org**	6. (0.220) *David Eppstein* *URL:www.ics.uci.edu/∼eppstein*	6. (0.256) **The Geometry Junkyard** **URL:www.ics.uci.edu/∼eppstein/junk**
7. (0.083) *Joseph O'Rourke* *URL:cs.smith.edu/∼orourke*	7. (0.210) *ACM: Association for Computing* *URL:www.acm.org*	7. (0.220) **The former CGAL home page** **URL:www.cs.uu.nl/CGAL**
8. (0.081) *Fast Robust Predicates for Comput* *URL:www.cs.cmu.edu/∼quake/robust.h*	8. (0.205) *Springer Link - Publication* *URL:link.springer.de/link/service/*	8. (0.205) *Joseph O'Rourke* *URL:cs.smith.edu/∼orourke*
9. (0.080) **Computational Geometry** **URL:www.uiuc.edu/ph/www/jeffe/comp**	9. (0.194) *The Stony Brook Algorithm Reposit* *URL:www.cs.sunysb.edu/∼algorith*	9. (0.165) **Computational Geometry** **URL:compgeom.cs.uiuc.edu/∼jeffe/co**
10. (0.074) **The Geometry Junkyard** **URL:www.ics.uci.edu/∼eppstein/ju**	10. (0.191) *MathWorld* *URL:mathworld.wolfram.com*	10. (0.150) **The CGAL Home Page** **URL:www.cgal.org**

AT-avg	Norm	BFS
1. (1.000) *Geometry in Action* *URL:www.ics.uci.edu/∼eppstein/geom*	1. (1.000) *Geometry in Action* *URL:www.ics.uci.edu/∼eppstein/geom*	1. (1.000) *Geometry in Action* *URL:www.ics.uci.edu/∼eppstein/geom*
2. (0.899) WebCT.com URL:www.webct.com	2. (0.507) **The former CGAL home page** **URL:www.cs.uu.nl/CGAL**	2. (0.898) **The former CGAL home page** **URL:www.cs.uu.nl/CGAL**
3. (0.885) DREXEL UNIVERSITY URL:www.drexel.edu	3. (0.420) **The Geometry Junkyard** **URL:www.ics.uci.edu/∼eppstein/junk**	3. (0.841) *Springer Link - Publication* *URL:link.springer.de/link/service/*
4. (0.885) A Virtual Math Community URL:www.drexel.edu/ia/mathforum	4. (0.393) **Computational Geometry** **URL:www.scs.carleton.ca/∼csgs/reso**	4. (0.836) **The Geometry Junkyard** **URL:www.ics.uci.edu/∼eppstein/junk**
5. (0.533) **The former CGAL home page** **URL:www.cs.uu.nl/CGAL**	5. (0.339) WebCT.com URL:www.webct.com	5. (0.815) *LEDA moved to Algorithmic Sol* *URL:www.mpi-sb.mpg.de/LEDA/leda.ht*
6. (0.502) **The Geometry Junkyard** **URL:www.ics.uci.edu/∼eppstein/junk**	6. (0.327) DREXEL UNIVERSITY URL:www.drexel.edu	6. (0.814) *David Eppstein* *URL:www.ics.uci.edu/∼eppstein*
7. (0.486) **Computational Geometry** **URL:www.scs.carleton.ca/∼csgs/reso**	7. (0.327) A Virtual Math Community URL:www.drexel.edu/ia/mathforum	7. (0.808) *Joseph O'Rourke* *URL:cs.smith.edu/∼orourke*
8. (0.356) **Computational Geometry** **URL:compgeom.cs.uiuc.edu/∼jeffe/co**	8. (0.313) **Computational Geometry** **URL:compgeom.cs.uiuc.edu/∼jeffe/co**	8. (0.802) **Computational Geometry** **URL:www.scs.carleton.ca/∼csgs/reso**
9. (0.316) *Joseph O'Rourke* *URL:cs.smith.edu/∼orourke*	9. (0.250) *Springer Link - Publication* *URL:link.springer.de/link/service/*	9. (0.752) *The Stony Brook Algorithm Reposit* *URL:www.cs.sunysb.edu/∼algorith*
10. (0.284) *Springer Link - Publication* *URL:link.springer.de/link/service/*	10. (0.250) *ACM: Association for Computing* *URL:www.acm.org*	10. (0.734) **The compgeom mailing lists** **URL:netlib.bell-labs.com/netlib/**

Table C.14: Query "computational geometry"

Hits	PageRank	InDegree
1. (1.000) **Death Penalty Information** **URL:www.deathpenaltyinfo.org**	1. (1.000) *Moratorium Campaign – Current* *URL:www.capwiz.com/moratorium/issu*	1. (1.000) **Death Penalty Information** **URL:www.deathpenaltyinfo.org**
2. (0.968) **NCADP - National Coalition** **URL:www.ncadp.org**	2. (0.954) CBS.SportsLine.com URL:cbs.sportsline.com	2. (0.807) **NCADP - National Coalition** **URL:www.ncadp.org**
3. (0.904) **CUADP: For Alternatives to** **URL:www.cuadp.org**	3. (0.912) **Empty title field** **URL:www.ncadp2.org**	3. (0.522) **Pro-death penalty.com** **URL:www.prodeathpenalty.com**
4. (0.896) *Death Penalty : Sister Helen Pr* *URL:www.moratorium2000.org*	4. (0.746) Office of Justice Programs Home URL:www.ojp.usdoj.gov	4. (0.454) CBS.SportsLine.com URL:cbs.sportsline.com
5. (0.893) **Death Penalty Information (fr** **URL:sun.soci.niu.edu/∼critcrim/dp/**	5. (0.710) **NCADP - National Coalition** **URL:www.ncadp.org**	5. (0.424) **CUADP: For Alternatives to** **URL:www.cuadp.org**
6. (0.890) **Death Penalty Focus** **URL:www.deathpenalty.org**	6. (0.623) Office of Juvenile Justice and URL:www.ojjdp.ncjrs.org	6. (0.397) **Death Penalty Information (fr** **URL:sun.soci.niu.edu/∼critcrim/dp/**
7. (0.887) **Campaign To End The Death** **URL:www.nodeathpenalty.org**	7. (0.430) *Juvenile Campaign* *URL:www.ncadp.org/html/juvenile_ca*	7. (0.383) *American Civil Liberties Union* *URL:www.aclu.org*
8. (0.878) **Death Penalty Links** **URL:www.derechos.org/dp**	8. (0.429) **Death Penalty Information** **URL:www.deathpenaltyinfo.org**	8. (0.380) **Campaign To End The Death** **URL:www.nodeathpenalty.org**
9. (0.865) *Virginians for Alternatives to* *URL:www.vadp.org*	9. (0.418) *Pro Death Penalty.com Discussio* *URL:prodp.proboards18.com*	9. (0.373) *Death Penalty : Sister Helen Pr* *URL:www.moratorium2000.org*
10. (0.863) *Ohioans To Stop Executions* *URL:www.otse.org*	10. (0.407) *Human Rights Watch* *URL:store.yahoo.com/hrwpubs*	10. (0.369) **Death Penalty Links** **URL:www.derechos.org/dp**

HubAvg	Max	AT-med
1. (1.000) CBS.SportsLine.com URL:cbs.sportsline.com	1. (1.000) **Death Penalty Information** **URL:www.deathpenaltyinfo.org**	1. (1.000) **Death Penalty Information** **URL:www.deathpenaltyinfo.org**
2. (0.008) *TDCJ - Statistics - Home Page* *URL:www.tdcj.state.tx.us/statistic*	2. (0.661) **NCADP - National Coalition** **URL:www.ncadp.org**	2. (0.774) **NCADP - National Coalition** **URL:www.ncadp.org**
3. (0.004) CBSNews.com URL:www.cbsnews.com	3. (0.388) **Pro-death penalty.com** **URL:www.prodeathpenalty.com**	3. (0.444) **Pro-death penalty.com** **URL:www.prodeathpenalty.com**
4. (0.002) **Death Penalty Information** **URL:www.deathpenaltyinfo.org**	4. (0.324) **CUADP: For Alternatives to** **URL:www.cuadp.org**	4. (0.401) **CUADP: For Alternatives to** **URL:www.cuadp.org**
5. (0.002) **Pro-death penalty.com** **URL:www.prodeathpenalty.com**	5. (0.277) *Death Penalty : Sister Helen Pr* *URL:www.moratorium2000.org*	5. (0.348) *Death Penalty : Sister Helen Pr* *URL:www.moratorium2000.org*
6. (0.002) **Campaign To End The Death** **URL:www.nodeathpenalty.org**	6. (0.268) **Death Penalty Links** **URL:www.derechos.org/dp**	6. (0.329) **Death Penalty Links** **URL:www.derechos.org/dp**
7. (0.001) **American Civil Liberties Unio** **URL:www.aclu.org/death-penalty**	7. (0.259) **Campaign To End The Death** **URL:www.nodeathpenalty.org**	7. (0.323) **Campaign To End The Death** **URL:www.nodeathpenalty.org**
8. (0.001) *Death Penalty : Sister Helen Pr* *URL:www.moratorium2000.org*	8. (0.244) **Death Penalty Information (fr** **URL:sun.soci.niu.edu/∼critcrim/dp/**	8. (0.306) **Death Penalty Information (fr** **URL:sun.soci.niu.edu/∼critcrim/dp/**
9. (0.001) **LII: Law about...the Death Pe** **URL:www.law.cornell.edu/topics/dea**	9. (0.227) *Murder Victims Families for Rec* *URL:www.mvfr.org*	9. (0.278) *Murder Victims Families for Rec* *URL:www.mvfr.org*
10. (0.001) ABCNEWS.com: Home URL:www.abcnews.com	10. (0.222) **American Civil Liberties Uni** **URL:www.aclu.org/death-penalty**	10. (0.277) **American Civil Liberties Uni** **URL:www.aclu.org/death-penalty**

AT-avg	Norm	BFS
1. (1.000) **Death Penalty Information** **URL:www.deathpenaltyinfo.org**	1. (1.000) **Death Penalty Information** **URL:www.deathpenaltyinfo.org**	1. (1.000) **Death Penalty Information** **URL:www.deathpenaltyinfo.org**
2. (0.830) **NCADP - National Coalition** **URL:www.ncadp.org**	2. (0.768) **NCADP - National Coalition** **URL:www.ncadp.org**	2. (0.899) **NCADP - National Coalition** **URL:www.ncadp.org**
3. (0.472) **CUADP: For Alternatives to** **URL:www.cuadp.org**	3. (0.436) **Pro-death penalty.com** **URL:www.prodeathpenalty.com**	3. (0.867) **Pro-death penalty.com** **URL:www.prodeathpenalty.com**
4. (0.448) **Pro-death penalty.com** **URL:www.prodeathpenalty.com**	4. (0.412) **CUADP: For Alternatives to** **URL:www.cuadp.org**	4. (0.843) *Death Penalty : Sister Helen Pr* *URL:www.moratorium2000.org*
5. (0.418) *Death Penalty : Sister Helen Pr* *URL:www.moratorium2000.org*	5. (0.364) *Death Penalty : Sister Helen Pr* *URL:www.moratorium2000.org*	5. (0.838) **Death Penalty Information (fr** **URL:sun.soci.niu.edu/∼critcrim/dp/**
6. (0.386) **Campaign To End The Death** **URL:www.nodeathpenalty.org**	6. (0.344) **Campaign To End The Death** **URL:www.nodeathpenalty.org**	6. (0.824) **CUADP: For Alternatives to** **URL:www.cuadp.org**
7. (0.384) **Death Penalty Links** **URL:www.derechos.org/dp**	7. (0.343) **Death Penalty Links** **URL:www.derechos.org/dp**	7. (0.816) **Death Penalty Links** **URL:www.derechos.org/dp**
8. (0.380) **Death Penalty Information (fr** **URL:sun.soci.niu.edu/∼critcrim/dp/**	8. (0.333) **Death Penalty Information (fr** **URL:sun.soci.niu.edu/∼critcrim/dp/**	8. (0.811) **Campaign To End The Death** **URL:www.nodeathpenalty.org**
9. (0.332) **Death Penalty Focus** **URL:www.deathpenalty.org**	9. (0.294) **Death Penalty Focus** **URL:www.deathpenalty.org**	9. (0.801) **Amnesty International** **URL:www.web.amnesty.org/rmp/dplibr**
10. (0.325) *Murder Victims Families for Rec* *URL:www.mvfr.org*	10. (0.290) *Murder Victims Families for Rec* *URL:www.mvfr.org*	10. (0.798) *Murder Victims Families for Rec* *URL:www.mvfr.org*

Table C.15: Query “death penalty”

Hits	PageRank	InDegree
1. (1.000) **NCBI HomePage** **URL:www.ncbi.nlm.nih.gov**	1. (1.000) *National Institutes of Health (* *URL:www.nih.gov*	1. (1.000) **NCBI HomePage** **URL:www.ncbi.nlm.nih.gov**
2. (0.656) **The Genome Database** **URL:www.gdb.org**	2. (0.958) *National Institute of General M* *URL:www.nigms.nih.gov*	2. (0.640) *National Institutes of Health (* *URL:www.nih.gov*
3. (0.644) *The Wellcome Trust Sanger Insti* *URL:www.sanger.ac.uk*	3. (0.565) All Conferences . Com URL:www.allconferences.net	3. (0.541) **OMIM Home Page – Online** **URL:www3.ncbi.nlm.nih.gov/Omim**
4. (0.565) **The Institute for Genomic Res** **URL:www.tigr.org**	4. (0.555) Castles of the World URL:www.castles.org	4. (0.497) **www.genome.gov** **URL:www.nhgri.nih.gov**
5. (0.545) **OMIM Home Page – Online** **URL:www3.ncbi.nlm.nih.gov/Omim**	5. (0.549) *The Jackson Laboratory- Advanci* *URL:www.jax.org*	5. (0.451) **The Genome Database** **URL:www.gdb.org**
6. (0.537) **Whitehead Institute/MIT** **URL:www-genome.wi.mit.edu**	6. (0.547) **MGI 2.96nbsp;-nbsp;Mouse Ge** **URL:www.informatics.jax.org**	6. (0.409) **Genetic Alliance, Inc.** **URL:www.geneticalliance.org**
7. (0.502) **www.genome.gov** **URL:www.nhgri.nih.gov**	7. (0.484) Crosses.org URL:www.crosses.org	7. (0.396) *The Wellcome Trust Sanger Insti* *URL:www.sanger.ac.uk*
8. (0.491) *National Institutes of Health (* *URL:www.nih.gov*	8. (0.476) *U.S. National Library of Medici* *URL:www.nlm.nih.gov*	8. (0.390) *U.S. National Library of Medici* *URL:www.nlm.nih.gov*
9. (0.482) **European Bioinformatics Insti** **URL:www.ebi.ac.uk**	9. (0.386) **NCBI HomePage** **URL:www.ncbi.nlm.nih.gov**	9. (0.376) **The Genetic Algorithms** **URL:www.aic.nrl.navy.mil/galist**
10. (0.478) **UK MRC HGMP-RC** **URL:www.hgmp.mrc.ac.uk**	10. (0.337) *Office of Science* *URL:www.er.doe.gov*	10. (0.352) **The Institute for Genomic Re** **URL:www.tigr.org**

HubAvg	Max	AT-med
1. (1.000) **NCBI HomePage** **URL:www.ncbi.nlm.nih.gov**	1. (1.000) **NCBI HomePage** **URL:www.ncbi.nlm.nih.gov**	1. (1.000) **NCBI HomePage** **URL:www.ncbi.nlm.nih.gov**
2. (0.767) *National Institutes of Health (* *URL:www.nih.gov*	2. (0.361) *National Institutes of Health (* *URL:www.nih.gov*	2. (0.450) *National Institutes of Health (* *URL:www.nih.gov*
3. (0.432) *U.S. National Library of Medici* *URL:www.nlm.nih.gov*	3. (0.338) **The Genome Database** **URL:www.gdb.org**	3. (0.391) **The Genome Database** **URL:www.gdb.org**
4. (0.368) **OMIM Home Page – Online** **URL:www3.ncbi.nlm.nih.gov/Omim**	4. (0.316) **OMIM Home Page – Online** **URL:www3.ncbi.nlm.nih.gov/Omim**	4. (0.384) **OMIM Home Page – Online** **URL:www3.ncbi.nlm.nih.gov/Omim**
5. (0.339) **www.genome.gov** **URL:www.nhgri.nih.gov**	5. (0.284) **www.genome.gov** **URL:www.nhgri.nih.gov**	5. (0.348) **www.genome.gov** **URL:www.nhgri.nih.gov**
6. (0.302) **The Genome Database** **URL:www.gdb.org**	6. (0.277) *The Wellcome Trust Sanger Insti* *URL:www.sanger.ac.uk*	6. (0.320) *The Wellcome Trust Sanger Insti* *URL:www.sanger.ac.uk*
7. (0.261) **Genetic Alliance, Inc.** **URL:www.geneticalliance.org**	7. (0.256) **Whitehead Institute/MIT** **URL:www-genome.wi.mit.edu**	7. (0.286) **The Institute for Genomic Res** **URL:www.tigr.org**
8. (0.218) *The Wellcome Trust Sanger Insti* *URL:www.sanger.ac.uk*	8. (0.251) **The Institute for Genomic Res** **URL:www.tigr.org**	8. (0.280) **Whitehead Institute/MIT** **URL:www-genome.wi.mit.edu**
9. (0.209) *Entrez-PubMed* *URL:www4.ncbi.nlm.nih.gov/PubMed*	9. (0.224) **European Bioinformatics Insti** **URL:www.ebi.ac.uk**	9. (0.269) *U.S. National Library of Medici* *URL:www.nlm.nih.gov*
10. (0.208) **Whitehead Institute/MIT** **URL:www-genome.wi.mit.edu**	10. (0.210) *U.S. National Library of Medici* *URL:www.nlm.nih.gov*	10. (0.245) **European Bioinformatics Inst** **URL:www.ebi.ac.uk**

AT-avg	Norm	BFS
1. (1.000) **NCBI HomePage** **URL:www.ncbi.nlm.nih.gov**	1. (1.000) **NCBI HomePage** **URL:www.ncbi.nlm.nih.gov**	1. (1.000) **NCBI HomePage** **URL:www.ncbi.nlm.nih.gov**
2. (0.467) *National Institutes of Health (* *URL:www.nih.gov*	2. (0.431) *National Institutes of Health (* *URL:www.nih.gov*	2. (0.901) **OMIM Home Page – Online** **URL:www3.ncbi.nlm.nih.gov/Omim**
3. (0.454) **The Genome Database** **URL:www.gdb.org**	3. (0.412) **The Genome Database** **URL:www.gdb.org**	3. (0.866) *National Institutes of Health (* *URL:www.nih.gov*
4. (0.438) **OMIM Home Page – Online** **URL:www3.ncbi.nlm.nih.gov/Omim**	4. (0.395) **OMIM Home Page – Online** **URL:www3.ncbi.nlm.nih.gov/Omim**	4. (0.841) **www.genome.gov** **URL:www.nhgri.nih.gov**
5. (0.395) **www.genome.gov** **URL:www.nhgri.nih.gov**	5. (0.357) **www.genome.gov** **URL:www.nhgri.nih.gov**	5. (0.821) **GeneTests Home Page** **URL:www.geneclinics.org**
6. (0.379) *The Wellcome Trust Sanger Insti* *URL:www.sanger.ac.uk*	6. (0.349) *The Wellcome Trust Sanger Insti* *URL:www.sanger.ac.uk*	6. (0.819) *U.S. National Library of Medici* *URL:www.nlm.nih.gov*
7. (0.331) **The Institute for Genomic Res** **URL:www.tigr.org**	7. (0.309) **The Institute for Genomic Res** **URL:www.tigr.org**	7. (0.813) **The Genome Database** **URL:www.gdb.org**
8. (0.317) **Whitehead Institute/MIT** **URL:www-genome.wi.mit.edu**	8. (0.302) **Whitehead Institute/MIT** **URL:www-genome.wi.mit.edu**	8. (0.810) Darren Fisher nbsp; Computer A URL:www.dazzy-d.demon.co.uk
9. (0.284) *U.S. National Library of Medici* *URL:www.nlm.nih.gov*	9. (0.265) **European Bioinformatics Insti** **URL:www.ebi.ac.uk**	9. (0.809) **The Institute for Genomic Res** **URL:www.tigr.org**
10. (0.278) **European Bioinformatics Inst** **URL:www.ebi.ac.uk**	10. (0.257) *U.S. National Library of Medici* *URL:www.nlm.nih.gov*	10. (0.806) *The Wellcome Trust Sanger Insti* *URL:www.sanger.ac.uk*

Table C.16: Query “genetic”

Hits	PageRank	InDegree
1. (1.000) **The Geometry Center** **URL:freeabel.geom.umn.edu**	1. (1.000) WebCT.com URL:www.webct.com	1. (1.000) **The Geometry Center** **URL:freeabel.geom.umn.edu**
2. (0.628) **Geometry in Action** **URL:www.ics.uci.edu/∼eppstein/geom**	2. (0.931) University of California, Irvin URL:www.uci.edu	2. (0.876) WebCT.com URL:www.webct.com
3. (0.610) *The Geometry Junkyard* *URL:www.ics.uci.edu/∼eppstein/junk*	3. (0.791) Site Meter - Counter and Statis URL:sm2.sitemeter.com/stats.asp?si	3. (0.618) **Geometry in Action** **URL:www.ics.uci.edu/∼eppstein/geom**
4. (0.446) **MathWorld** **URL:mathworld.wolfram.com**	4. (0.644) Medieval Art, History and Archi URL:www.newyorkcarver.com	4. (0.560) **MathWorld** **URL:mathworld.wolfram.com**
5. (0.404) **Euclid's Elements, Introducti** **URL:aleph0.clarku.edu/∼djoyce/java**	5. (0.618) WebEQ has moved URL:www.webeq.com	5. (0.537) *The Geometry Junkyard* *URL:www.ics.uci.edu/∼eppstein/junk*
6. (0.373) **A Gallery of Interactive On-L** **URL:www.geom.umn.edu/apps/gallery.**	6. (0.593) **The Geometry Center** **URL:freeabel.geom.umn.edu**	6. (0.452) **Geometry and Topology** **URL:www.maths.warwick.ac.uk/gt**
7. (0.336) *The Math Forum Home Page* *URL:mathforum.org*	7. (0.583) Design Science - How Science Co URL:www.dessci.com	7. (0.452) **Euclid's Elements, Introducti** **URL:aleph0.clarku.edu/∼djoyce/java**
8. (0.300) **Geometry Formulas and Facts** **URL:www.geom.umn.edu/docs/referenc**	8. (0.438) *The Interactive Geometry Softwa* *URL:www.cinderella.de*	8. (0.436) *SpringerLink - Publication* *URL:link.springer.de/link/service/*
9. (0.262) Wolfram Research, Inc. URL:www.wri.com	9. (0.436) National Science Foundation (NS URL:www.nsf.gov	9. (0.417) *The Math Forum Home Page* *URL:mathforum.org*
10. (0.258) *Directory of Computational Geom* *URL:www.geom.umn.edu/software/cgli*	10. (0.415) Knowledge Management Software URL:math.askme.com/op	10. (0.386) *Empty title field* *URL:www.ams.org/ecgd*

HubAvg	Max	AT-med
1. (1.000) WebCT.com URL:www.webct.com	1. (1.000) **The Geometry Center** **URL:freeabel.geom.umn.edu**	1. (1.000) **The Geometry Center** **URL:freeabel.geom.umn.edu**
2. (0.069) **The Geometry Center** **URL:freeabel.geom.umn.edu**	2. (0.349) **Geometry in Action** **URL:www.ics.uci.edu/∼eppstein/geom**	2. (0.466) **Geometry in Action** **URL:www.ics.uci.edu/∼eppstein/geom**
3. (0.056) *The Geometry Junkyard* *URL:www.ics.uci.edu/∼eppstein/junk*	3. (0.329) *The Geometry Junkyard* *URL:www.ics.uci.edu/∼eppstein/junk*	3. (0.439) *The Geometry Junkyard* *URL:www.ics.uci.edu/∼eppstein/junk*
4. (0.042) *Connected Geometry Home Page* *URL:www.edc.org/LTT/ConnGeo*	4. (0.237) **MathWorld** **URL:mathworld.wolfram.com**	4. (0.333) **MathWorld** **URL:mathworld.wolfram.com**
5. (0.038) *Cynthia Lanius' Lessons: Geomet* *URL:math.rice.edu/∼lanius/Geom*	5. (0.231) **A Gallery of Interactive On-L** **URL:www.geom.umn.edu/apps/gallery.**	5. (0.285) **Euclid's Elements, Introducti** **URL:aleph0.clarku.edu/∼djoyce/java**
6. (0.032) **Dynamic Geometry Home** **URL:www.edc.org/LTT/DG**	6. (0.227) WebCT.com URL:www.webct.com	6. (0.278) **A Gallery of Interactive On-L** **URL:www.geom.umn.edu/apps/gallery.**
7. (0.032) *C.a.R.* *URL:mathsrv.ku-eichstaett.de/MGF/h*	7. (0.226) **Euclid's Elements, Introducti** **URL:aleph0.clarku.edu/∼djoyce/java**	7. (0.277) WebCT.com URL:www.webct.com
8. (0.031) **Geometry in Action** **URL:www.ics.uci.edu/∼eppstein/geom**	8. (0.225) *The Math Forum Home Page* *URL:mathforum.org*	8. (0.275) *The Math Forum Home Page* *URL:mathforum.org*
9. (0.031) **Geometry Step by Step from** **URL:agutie.homestead.com**	9. (0.175) **GANG — Geometry Analysis** **URL:www.gang.umass.edu**	9. (0.198) **Geometry Formulas and Facts** **URL:www.geom.umn.edu/docs/referenc**
10. (0.031) *The Interactive Geometry Softwa* *URL:www.cinderella.de*	10. (0.161) **Geometry Formulas and Facts** **URL:www.geom.umn.edu/docs/refere**	10. (0.197) **GANG — Geometry Analysis** **URL:www.gang.umass.edu**

AT-avg	Norm	BFS
1. (1.000) **The Geometry Center** **URL:freeabel.geom.umn.edu**	1. (1.000) **The Geometry Center** **URL:freeabel.geom.umn.edu**	1. (1.000) **The Geometry Center** **URL:freeabel.geom.umn.edu**
2. (0.542) **Geometry in Action** **URL:www.ics.uci.edu/∼eppstein/geom**	2. (0.429) **Geometry in Action** **URL:www.ics.uci.edu/∼eppstein/geom**	2. (0.898) **Geometry in Action** **URL:www.ics.uci.edu/∼eppstein/geom**
3. (0.509) *The Geometry Junkyard* *URL:www.ics.uci.edu/∼eppstein/junk*	3. (0.405) *The Geometry Junkyard* *URL:www.ics.uci.edu/∼eppstein/junk*	3. (0.891) *The Geometry Junkyard* *URL:www.ics.uci.edu/∼eppstein/junk*
4. (0.389) **MathWorld** **URL:mathworld.wolfram.com**	4. (0.307) **MathWorld** **URL:mathworld.wolfram.com**	4. (0.838) **MathWorld** **URL:mathworld.wolfram.com**
5. (0.336) **Euclid's Elements, Introducti** **URL:aleph0.clarku.edu/∼djoyce/java**	5. (0.275) **Euclid's Elements, Introducti** **URL:aleph0.clarku.edu/∼djoyce/java**	5. (0.816) **A Gallery of Interactive On-L** **URL:www.geom.umn.edu/apps/gallery.**
6. (0.323) **A Gallery of Interactive On-L** **URL:www.geom.umn.edu/apps/gallery.**	6. (0.271) WebCT.com URL:www.webct.com	6. (0.812) **Geometry Formulas and Facts** **URL:www.geom.umn.edu/docs/referenc**
7. (0.319) *The Math Forum Home Page* *URL:mathforum.org*	7. (0.266) **A Gallery of Interactive On-L** **URL:www.geom.umn.edu/apps/gallery.**	7. (0.801) **Euclid's Elements, Introducti** **URL:aleph0.clarku.edu/∼djoyce/java**
8. (0.252) WebCT.com URL:www.webct.com	8. (0.263) *The Math Forum Home Page* *URL:mathforum.org*	8. (0.798) *Directory of Computational Geom* *URL:www.geom.umn.edu/software/cgli*
9. (0.229) **Geometry Formulas and Facts** **URL:www.geom.umn.edu/docs/referenc**	9. (0.192) **GANG — Geometry Analysis** **URL:www.gang.umass.edu**	9. (0.789) WebCT.com URL:www.webct.com
10. (0.215) National Council of Teachers of URL:www.nctm.org	10. (0.188) **Geometry Formulas and Facts** **URL:www.geom.umn.edu/docs/refere**	10. (0.788) *Native American Geometry* *URL:www.earthmeasure.com*

Table C.17: Query "geometry"

Hits	PageRank	InDegree
1. (1.000) *INDYMEDIA TIJUANA :: centro* *URL:www.tijuanaimc.org*	1. (1.000) Welcome to Harvard University URL:www.harvard.edu	1. (1.000) *Independent Media Center -* *URL:www.indymedia.org*
2. (0.999) *Baltimore Independent Media Cen* *URL:baltimoreimc.org*	2. (0.602) Harvard Business School URL:www.hbs.edu	2. (0.793) *indymedia uk* *URL:www.indymedia.org.uk*
3. (0.994) *Empty title field* *URL:sdimc.org*	3. (0.465) *Center for International Develo* *URL:www.cid.harvard.edu*	3. (0.756) **International Forum on Global** **URL:www.ifg.org**
4. (0.991) *Melbourne Independent Media Cen* *URL:www.melbourne.indymedia.org*	4. (0.235) **OneWorld.net -** **URL:www.oneworld.net**	4. (0.695) **WTO — Welcome to the WTO** **URL:www.wto.org**
5. (0.989) *Urbana-Champaign Independent* *URL:www.ucimc.org*	5. (0.215) **Globalization Issues Classifi** **URL:adlistings.globalization.about**	5. (0.690) *INDYMEDIA TIJUANA :: centro* *URL:www.tijuanaimc.org*
6. (0.988) *Danbury, CT Independent Media* *URL:www.madhattersimc.org*	6. (0.197) *Foreign Affairs - Home* *URL:www.foreignaffairs.org*	6. (0.681) The Institute for Deep Ecology: URL:www.deep-ecology.org
7. (0.987) *IndyMedia Center -* *URL:indymedia.org.il*	7. (0.193) **The Globalization Website** **URL:www.emory.edu/SOC/globalizatio**	7. (0.653) *The World Bank Group* *URL:www.worldbank.org*
8. (0.984) *Indymedia - news - Aotearoa Ind* *URL:www.indymedia.org.nz*	8. (0.192) *Council on Foreign Relations* *URL:www.cfr.org*	8. (0.638) *Baltimore Independent Media Cen* *URL:baltimoreimc.org*
9. (0.983) *Vaikuttava Tietotoimisto (VAI)* *URL:www.vaikuttava.net*	9. (0.190) *CFR* *URL:www.cfr.org/about/mission.php*	9. (0.629) *Indymedia - news - Aotearoa Ind* *URL:www.indymedia.org.nz*
10. (0.981) *Adelaide indymedia - webcast ne* *URL:adelaide.indymedia.org.au*	10. (0.190) IDG.net — The Global IT Network URL:www.idg.net	10. (0.629) *Melbourne Independent Media Cen* *URL:www.melbourne.indymedia.org*

HubAvg	Max	AT-med
1. (1.000) *INDYMEDIA TIJUANA :: centro* *URL:www.tijuanaimc.org*	1. (1.000) *Independent Media Center -* *URL:www.indymedia.org*	1. (1.000) *Independent Media Center -* *URL:www.indymedia.org*
2. (0.998) *Independent Media Center -* *URL:www.indymedia.org*	2. (0.723) *indymedia uk* *URL:www.indymedia.org.uk*	2. (0.788) *indymedia uk* *URL:www.indymedia.org.uk*
3. (0.985) *Baltimore Independent Media Cen* *URL:baltimoreimc.org*	3. (0.618) *INDYMEDIA TIJUANA :: centro* *URL:www.tijuanaimc.org*	3. (0.698) *INDYMEDIA TIJUANA :: centro* *URL:www.tijuanaimc.org*
4. (0.982) *Melbourne Independent Media Cen* *URL:www.melbourne.indymedia.org*	4. (0.585) *Baltimore Independent Media Cen* *URL:baltimoreimc.org*	4. (0.668) *Baltimore Independent Media Cen* *URL:baltimoreimc.org*
5. (0.980) *Urbana-Champaign Independent* *URL:www.ucimc.org*	5. (0.575) *Melbourne Independent Media Cen* *URL:www.melbourne.indymedia.org*	5. (0.657) *Melbourne Independent Media Cen* *URL:www.melbourne.indymedia.org*
6. (0.979) *Indymedia - news - Aotearoa Ind* *URL:www.indymedia.org.nz*	6. (0.573) *Indymedia - news - Aotearoa Ind* *URL:www.indymedia.org.nz*	6. (0.653) *Indymedia - news - Aotearoa Ind* *URL:www.indymedia.org.nz*
7. (0.973) *Empty title field* *URL:sdimc.org*	7. (0.568) *Urbana-Champaign Independent* *URL:www.ucimc.org*	7. (0.650) *Urbana-Champaign Independent* *URL:www.ucimc.org*
8. (0.973) *Danbury, CT Independent Media* *URL:www.madhattersimc.org*	8. (0.564) *IndyMedia Center -* *URL:indymedia.org.il*	8. (0.645) *IndyMedia Center -* *URL:indymedia.org.il*
9. (0.968) *IndyMedia Center -* *URL:indymedia.org.il*	9. (0.564) *Empty title field* *URL:sdimc.org*	9. (0.645) *Empty title field* *URL:sdimc.org*
10. (0.966) *Adelaide indymedia - webcast ne* *URL:adelaide.indymedia.org.au*	10. (0.563) *Danbury, CT Independent Media* *URL:www.madhattersimc.org*	10. (0.644) *Danbury, CT Independent Media* *URL:www.madhattersimc.org*

AT-avg	Norm	BFS
1. (1.000) *Independent Media Center -* *URL:www.indymedia.org*	1. (1.000) *Independent Media Center -* *URL:www.indymedia.org*	1. (1.000) *Independent Media Center -* *URL:www.indymedia.org*
2. (0.930) *INDYMEDIA TIJUANA :: centro* *URL:www.tijuanaimc.org*	2. (0.956) *INDYMEDIA TIJUANA :: centro* *URL:www.tijuanaimc.org*	2. (0.917) **WTO — Welcome to the WTO** **URL:www.wto.org**
3. (0.916) *Baltimore Independent Media Cen* *URL:baltimoreimc.org*	3. (0.943) *Baltimore Independent Media Cen* *URL:baltimoreimc.org*	3. (0.906) The Institute for Deep Ecology: URL:www.deep-ecology.org
4. (0.906) *Melbourne Independent Media Cen* *URL:www.melbourne.indymedia.org*	4. (0.934) *Melbourne Independent Media Cen* *URL:www.melbourne.indymedia.org*	4. (0.892) **International Forum on Global** **URL:www.ifg.org**
5. (0.899) *Indymedia - news - Aotearoa Ind* *URL:www.indymedia.org.nz*	5. (0.929) *Empty title field* *URL:sdimc.org*	5. (0.847) **IMF – International Monetary** **URL:www.imf.org**
6. (0.895) *Urbana-Champaign Independent* *URL:www.ucimc.org*	6. (0.929) *Urbana-Champaign Independent* *URL:www.ucimc.org*	6. (0.838) *The World Bank Group* *URL:www.worldbank.org*
7. (0.892) *Empty title field* *URL:sdimc.org*	7. (0.929) *Indymedia - news - Aotearoa Ind* *URL:www.indymedia.org.nz*	7. (0.823) *indymedia uk* *URL:www.indymedia.org.uk*
8. (0.892) *Danbury, CT Independent Media* *URL:www.madhattersimc.org*	8. (0.926) *Danbury, CT Independent Media* *URL:www.madhattersimc.org*	8. (0.819) *Bretton Woods Project* *URL:www.brettonwoodsproject.org*
9. (0.890) *IndyMedia Center -* *URL:indymedia.org.il*	9. (0.925) *IndyMedia Center -* *URL:indymedia.org.il*	9. (0.815) *Welcome to the UN. It's your wo* *URL:www.un.org*
10. (0.885) *Adelaide indymedia - webcast ne* *URL:adelaide.indymedia.org.au*	10. (0.919) *Adelaide indymedia - webcast ne* *URL:adelaide.indymedia.org.au*	10. (0.789) *Landless Workers' Movement* *URL:www.mstbrazil.org*

Table C.18: Query “globalization”

HITS	PAGERANK	INDEGREE
1. (1.000) Coffee Club URL:www.batavia-rof.com	1. (1.000) Yahoo! Groups : Educate-Yoursel URL:groups.yahoo.com/group/Educate	1. (1.000) **National Rifle Association -** **URL:www.nra.org**
2. (0.982) Hotel and Travel URL:www.bwdriftwood.com	2. (0.811) Empty title field URL:educate-yourself.org	2. (0.782) **The Brady Campaign to** **URL:www.handguncontrol.org**
3. (0.935) Basement Writers URL:www.basement-writers.com	3. (0.562) Empty title field URL:www.cafeshops.com/cp/store.asp	3. (0.628) *Gun Owners of America* *URL:www.gunowners.org*
4. (0.935) Before Today URL:www.beforetoday.com	4. (0.541) **Legislative Action Center - G** **URL:capwiz.com/jointogether**	4. (0.504) **The Violence Policy Center** **URL:www.vpc.org**
5. (0.935) Bennett Boxing URL:www.bennettboxing.com	5. (0.472) **Gun Violence Home Page** **URL:www.jointogether.org/gv**	5. (0.474) *Jews for the Preservation of Fi* *URL:www.jpfo.org*
6. (0.935) Boeing Mail URL:www.boeingmail.com	6. (0.427) Empty title field URL:www.cafepress.com/esrgun	6. (0.457) **Coalition to Stop Gun Violenc** **URL:www.gunfree.org**
7. (0.935) Burdan USA URL:www.burdanusa.com	7. (0.392) **National Rifle Association -** **URL:www.nra.org**	7. (0.457) *CCRKBA Home Page* *URL:www.ccrkba.org*
8. (0.935) British Jokes URL:www.callusforfun.com	8. (0.389) **The Brady Campaign to** **URL:www.handguncontrol.org**	8. (0.385) **Women Against Gun Control** **URL:www.wagc.com**
9. (0.917) Religious Happenings URL:www.bellbrook-umc.com	9. (0.271) **Keep and Bear Armsnbsp;-nbs** **URL:www.keepandbeararms.com**	9. (0.355) **GunTruths: The truth about** **URL:www.guntruths.com**
10. (0.917) Blade Liners URL:www.bladeliners.com	10. (0.267) NewsMax.com - America's News URL:www.newsmax.com	10. (0.346) **GunCite: gun control and Sec** **URL:www.guncite.com**

HUBAVG	MAX	AT-MED
1. (1.000) **National Rifle Association -** **URL:www.nra.org**	1. (1.000) **National Rifle Association -** **URL:www.nra.org**	1. (1.000) **National Rifle Association -** **URL:www.nra.org**
2. (0.771) **The Brady Campaign to** **URL:www.handguncontrol.org**	2. (0.651) **The Brady Campaign to** **URL:www.handguncontrol.org**	2. (0.703) **The Brady Campaign to** **URL:www.handguncontrol.org**
3. (0.360) *Gun Owners of America* *URL:www.gunowners.org*	3. (0.537) *Gun Owners of America* *URL:www.gunowners.org*	3. (0.620) *Gun Owners of America* *URL:www.gunowners.org*
4. (0.328) **The Violence Policy Center** **URL:www.vpc.org**	4. (0.419) **The Violence Policy Center** **URL:www.vpc.org**	4. (0.488) **The Violence Policy Center** **URL:www.vpc.org**
5. (0.297) *CCRKBA Home Page* *URL:www.ccrkba.org*	5. (0.415) *CCRKBA Home Page* *URL:www.ccrkba.org*	5. (0.486) *CCRKBA Home Page* *URL:www.ccrkba.org*
6. (0.289) **Coalition to Stop Gun Violenc** **URL:www.gunfree.org**	6. (0.387) *Jews for the Preservation of Fi* *URL:www.jpfo.org*	6. (0.444) *Jews for the Preservation of Fi* *URL:www.jpfo.org*
7. (0.270) *Jews for the Preservation of Fi* *URL:www.jpfo.org*	7. (0.334) **Coalition to Stop Gun Violenc** **URL:www.gunfree.org**	7. (0.388) **Coalition to Stop Gun Violenc** **URL:www.gunfree.org**
8. (0.199) **Women Against Gun Control** **URL:www.wagc.com**	8. (0.298) **Women Against Gun Control** **URL:www.wagc.com**	8. (0.346) **Women Against Gun Control** **URL:www.wagc.com**
9. (0.178) **GunCite: gun control and Seco** **URL:www.guncite.com**	9. (0.266) *Second Amendment Sisters', Inc.* *URL:www.sas-aim.org*	9. (0.308) *Second Amendment Sisters', Inc.* *URL:www.sas-aim.org*
10. (0.171) *Second Amendment Sisters', Inc.* *URL:www.sas-aim.org*	10. (0.245) **GunCite: gun control and Sec** **URL:www.guncite.com**	10. (0.276) **GunCite: gun control and Sec** **URL:www.guncite.com**

AT-AVG	NORM	BFS
1. (1.000) **National Rifle Association -** **URL:www.nra.org**	1. (1.000) **National Rifle Association -** **URL:www.nra.org**	1. (1.000) **National Rifle Association -** **URL:www.nra.org**
2. (0.703) *Gun Owners of America* *URL:www.gunowners.org*	2. (0.685) **The Brady Campaign to** **URL:www.handguncontrol.org**	2. (0.922) **The Brady Campaign to** **URL:www.handguncontrol.org**
3. (0.671) **The Brady Campaign to** **URL:www.handguncontrol.org**	3. (0.606) *Gun Owners of America* *URL:www.gunowners.org*	3. (0.888) *Gun Owners of America* *URL:www.gunowners.org*
4. (0.567) *CCRKBA Home Page* *URL:www.ccrkba.org*	4. (0.478) *CCRKBA Home Page* *URL:www.ccrkba.org*	4. (0.826) **The Violence Policy Center** **URL:www.vpc.org**
5. (0.509) *Jews for the Preservation of Fi* *URL:www.jpfo.org*	5. (0.469) **The Violence Policy Center** **URL:www.vpc.org**	5. (0.811) *Jews for the Preservation of Fi* *URL:www.jpfo.org*
6. (0.509) **The Violence Policy Center** **URL:www.vpc.org**	6. (0.439) *Jews for the Preservation of Fi* *URL:www.jpfo.org*	6. (0.798) *CCRKBA Home Page* *URL:www.ccrkba.org*
7. (0.404) **Coalition to Stop Gun Violenc** **URL:www.gunfree.org**	7. (0.374) **Coalition to Stop Gun Violenc** **URL:www.gunfree.org**	7. (0.757) **GunCite: gun control and Seco** **URL:www.guncite.com**
8. (0.391) **Women Against Gun Control** **URL:www.wagc.com**	8. (0.339) **Women Against Gun Control** **URL:www.wagc.com**	8. (0.745) **Coalition to Stop Gun Violenc** **URL:www.gunfree.org**
9. (0.355) *Second Amendment Sisters', Inc.* *URL:www.sas-aim.org*	9. (0.302) *Second Amendment Sisters', Inc.* *URL:www.sas-aim.org*	9. (0.735) **Women Against Gun Control** **URL:www.wagc.com**
10. (0.311) **Keep and Bear Armsnbsp;-nb** **URL:www.keepandbeararms.com**	10. (0.275) **GunCite: gun control and Sec** **URL:www.guncite.com**	10. (0.729) *Second Amendment Sisters', Inc.* *URL:www.sas-aim.org*

Table C.19: Query "gun control"

Hits	PageRank	InDegree
1. (1.000) Google Search: URL:www.google.com/search	1. (1.000) CBS.SportsLine.com URL:cbs.sportsline.com	1. (1.000) Google Search: URL:www.google.com/search
2. (0.979) Moreover Technologies - Welcome URL:www.moreover.com	2. (0.855) Support UNICEF: A Project of th URL:www.supportunicef.org	2. (0.950) Moreover Technologies - Welcome URL:www.moreover.com
3. (0.977) ThePaperboy.com — Online URL:www.thepaperboy.com	3. (0.849) Welcome to the White House URL:www.whitehouse.gov	3. (0.899) ThePaperboy.com — Online URL:www.thepaperboy.com
4. (0.974) NewsLink URL:www.newslink.org/news.html	4. (0.815) Gannett Company, Inc. URL:www.gannett.com	4. (0.893) NewsLink URL:www.newslink.org/news.html
5. (0.960) Kidon Media-Link URL:www.kidon.com/media-link	5. (0.765) *USATODAY.com - News Info* *URL:www.usatoday.com*	5. (0.868) **United for Peace** **URL:www.unitedforpeace.org**
6. (0.955) Welcome - Roam International URL:www.roamintl.com	6. (0.711) **UNICEF - Iraq** **URL:www.unicef.org/noteworthy/iraq**	6. (0.868) Kidon Media-Link URL:www.kidon.com/media-link
7. (0.184) *Abu Dhabi News - current events* *URL:www.abudhabi.com*	7. (0.681) The New York Times: Theater Dir URL:www.nytbroadway.com/?thtrtx	7. (0.862) Welcome - Roam International URL:www.roamintl.com
8. (0.181) *Where is Raed ?* *URL:dear_raed.blogspot.com*	8. (0.560) *AlterNet: Top Stories* *URL:www.alternet.org*	8. (0.837) Welcome to the White House URL:www.whitehouse.gov
9. (0.158) *UNMOVIC* *URL:www.un.org/Depts/unmovic*	9. (0.472) CBC/Radio-Canada URL:cbc.radio-canada.ca	9. (0.742) *International A.N.S.W.E.R.* *URL:www.internationalanswer.org*
10. (0.156) *Iraq Liberated - U.S. Departmen* *URL:usinfo.state.gov/regional/nea/*	10. (0.472) Radio-Canada.ca URL:www.radio-canada.ca	10. (0.685) *Where is Raed ?* *URL:dear_raed.blogspot.com*

HubAvg	Max	AT-med
1. (1.000) Google Search: URL:www.google.com/search	1. (1.000) Google Search: URL:www.google.com/search	1. (1.000) Google Search: URL:www.google.com/search
2. (0.952) Moreover Technologies - Welcome URL:www.moreover.com	2. (0.945) Moreover Technologies - Welcome URL:www.moreover.com	2. (0.960) ThePaperboy.com — Online URL:www.thepaperboy.com
3. (0.930) ThePaperboy.com — Online URL:www.thepaperboy.com	3. (0.899) ThePaperboy.com — Online URL:www.thepaperboy.com	3. (0.958) Moreover Technologies - Welcome URL:www.moreover.com
4. (0.930) NewsLink URL:www.newslink.org/news.html	4. (0.893) NewsLink URL:www.newslink.org/news.html	4. (0.957) NewsLink URL:www.newslink.org/news.html
5. (0.910) Kidon Media-Link URL:www.kidon.com/media-link	5. (0.868) Kidon Media-Link URL:www.kidon.com/media-link	5. (0.930) Kidon Media-Link URL:www.kidon.com/media-link
6. (0.903) Welcome - Roam International URL:www.roamintl.com	6. (0.862) Welcome - Roam International URL:www.roamintl.com	6. (0.924) Welcome - Roam International URL:www.roamintl.com
7. (0.042) *Top Breaking News Headlines Fro* *URL:www.1stheadlines.com*	7. (0.085) **United for Peace** **URL:www.unitedforpeace.org**	7. (0.067) Google News URL:news.google.com
8. (0.038) Google News URL:news.google.com	8. (0.080) Google News URL:news.google.com	8. (0.061) Yahoo! UK Ireland News URL:uk.news.yahoo.com
9. (0.038) Yahoo! UK Ireland News URL:uk.news.yahoo.com	9. (0.074) Welcome to the White House URL:www.whitehouse.gov	9. (0.061) *Top Breaking News Headlines Fro* *URL:www.1stheadlines.com*
10. (0.021) Yahoo! UK Ireland URL:uk.yahoo.com	10. (0.063) *Where is Raed ?* *URL:dear_raed.blogspot.com*	10. (0.038) *Yahoo! News - Front Page* *URL:news.yahoo.com*

AT-avg	Norm	BFS
1. (1.000) Google Search: URL:www.google.com/search	1. (1.000) Google Search: URL:www.google.com/search	1. (1.000) *Where is Raed ?* *URL:dear_raed.blogspot.com*
2. (0.979) ThePaperboy.com — Online URL:www.thepaperboy.com	2. (0.965) Moreover Technologies - Welcome URL:www.moreover.com	2. (0.968) **United for Peace** **URL:www.unitedforpeace.org**
3. (0.978) NewsLink URL:www.newslink.org/news.html	3. (0.952) ThePaperboy.com — Online URL:www.thepaperboy.com	3. (0.954) *International A.N.S.W.E.R.* *URL:www.internationalanswer.org*
4. (0.977) Moreover Technologies - Welcome URL:www.moreover.com	4. (0.950) NewsLink URL:www.newslink.org/news.html	4. (0.941) **Iraq Body Count** **URL:www.iraqbodycount.net**
5. (0.963) Kidon Media-Link URL:www.kidon.com/media-link	5. (0.930) Kidon Media-Link URL:www.kidon.com/media-link	5. (0.907) **Antiwar.com** **URL:www.antiwar.com**
6. (0.959) Welcome - Roam International URL:www.roamintl.com	6. (0.925) Welcome - Roam International URL:www.roamintl.com	6. (0.903) **BBC NEWS — In Depth** **URL:news.bbc.co.uk/2/hi/in_depth/m**
7. (0.063) *Top Breaking News Headlines Fro* *URL:www.1stheadlines.com*	7. (0.062) Google News URL:news.google.com	7. (0.896) *The Nation* *URL:www.thenation.com/directory/vi*
8. (0.053) Yahoo! UK Ireland News URL:uk.news.yahoo.com	8. (0.061) *Top Breaking News Headlines Fro* *URL:www.1stheadlines.com*	8. (0.894) *Abu Dhabi News - current events* *URL:www.abudhabi.com*
9. (0.053) Google News URL:news.google.com	9. (0.056) Yahoo! UK Ireland News URL:uk.news.yahoo.com	9. (0.892) *DefenseLINK - Official Web Site* *URL:www.defenselink.mil*
10. (0.036) Venezuela URL:venezuela.newstrove.com	10. (0.038) *Yahoo! News - Front Page* *URL:news.yahoo.com*	10. (0.890) Welcome to the White House URL:www.whitehouse.gov

Table C.20: Query “iraq war”

Hits	PageRank	InDegree
1. (1.000) Apple iPod Updater 1.3 - Versio URL:www.VersionTracker.com/dyn/mor	1. (1.000) Apple .Mac Welcome URL:www.mac.com	1. (1.000) Apple .Mac Welcome URL:www.mac.com
2. (1.000) Apple iTunes 4.0 - VersionTrack URL:www.VersionTracker.com/dyn/mor	2. (0.563) Apple - Mac OS X URL:www.apple.com/macosx	2. (0.901) Griffman's OS X Collection URL:homepage.mac.com/rgriff
3. (1.000) VueScan 7.6.34 - VersionTracker URL:www.VersionTracker.com/dyn/mor	3. (0.561) Apple URL:www.apple.com	3. (0.744) Apple URL:www.apple.com
4. (1.000) VueScan 7.6.34 - VersionTracker URL:www.VersionTracker.com/dyn/mor	4. (0.509) AO Sunglasses for Military Pilo URL:aosunglasses.com	4. (0.719) Apple - Mac OS X URL:www.apple.com/macosx
5. (1.000) Apple iPod Updater 1.3 - Versio URL:www.VersionTracker.com/dyn/mor	5. (0.461) Bolle Coyote Serengeti Sunglass URL:123SUNGLASSES.COM	5. (0.686) Apple URL:www.apple.com/legal
6. (1.000) PHP 4.3.2RC2 - VersionTracker URL:www.VersionTracker.com/dyn/mor	6. (0.266) Griffman's OS X Collection URL:homepage.mac.com/rgriff	6. (0.686) Empty title field URL:www.gamesarchiv.com/Layout/?id
7. (1.000) Palm Desktop 4.1 - VersionTrack URL:www.VersionTracker.com/dyn/mor	7. (0.221) **Save the Jaguar** **URL:www.savethejaguar.com**	7. (0.669) ThinkGeek :: O'Reilly Store URL:www.thinkgeek.com/oreilly
8. (1.000) Apple - Games - Trailers URL:www.apple.com/games/trailers	8. (0.202) *Empty title field* *URL:www.cafepress.com/wcsjaguar*	8. (0.661) Apple iPod Updater 1.3 - Versio URL:www.VersionTracker.com/dyn/mor
9. (0.976) Dantz Retrospect 5.0 Driver Upd URL:www.VersionTracker.com/dyn/mor	9. (0.155) Amazon Honor System URL:s1.amazon.com/exec/varzea/pay/	9. (0.661) Apple iTunes 4.0 - VersionTrack URL:www.VersionTracker.com/dyn/mor
10. (0.976) iView MediaPro 1.5.7 - VersionT URL:www.VersionTracker.com/dyn/mor	10. (0.154) **Jaguar World Monthly Online** **URL:www.jagweb.com/jagworld**	10. (0.661) VueScan 7.6.34 - VersionTracker URL:www.VersionTracker.com/dyn/mor

HubAvg	Max	AT-med
1. (1.000) Griffman's OS X Collection URL:homepage.mac.com/rgriff	1. (1.000) Apple .Mac Welcome URL:www.mac.com	1. (1.000) Apple .Mac Welcome URL:www.mac.com
2. (0.558) Amazon Honor System URL:s1.amazon.com/exec/varzea/pay/	2. (0.603) Apple URL:www.apple.com	2. (0.859) Apple URL:www.apple.com
3. (0.018) Apple .Mac Welcome URL:www.mac.com	3. (0.584) Apple - Mac OS X URL:www.apple.com/macosx	3. (0.849) Apple - Mac OS X URL:www.apple.com/macosx
4. (0.011) Fink - Home URL:fink.sourceforge.net	4. (0.548) Apple URL:www.apple.com/legal	4. (0.753) Apple URL:www.apple.com/legal
5. (0.009) Apple - Mac OS X URL:www.apple.com/macosx	5. (0.393) Apple - Apple Customer Privacy URL:www.apple.com/legal/privacy	5. (0.549) Apple - Apple Customer Privacy URL:www.apple.com/legal/privacy
6. (0.006) Apple URL:www.apple.com	6. (0.169) The Apple Store (Japan) URL:www.apple.com/japanstore	6. (0.330) Apple iPod Updater 1.3 - Versio URL:www.VersionTracker.com/dyn/mor
7. (0.006) Apple - Discussions - Welcome URL:discussions.info.apple.com	7. (0.114) Fink - Home URL:fink.sourceforge.net	7. (0.330) Apple iTunes 4.0 - VersionTrack URL:www.VersionTracker.com/dyn/mor
8. (0.006) LinuxPrinting.org URL:www.linuxprinting.org	8. (0.085) Griffman's OS X Collection URL:homepage.mac.com/rgriff	8. (0.330) VueScan 7.6.34 - VersionTracker URL:www.VersionTracker.com/dyn/mor
9. (0.006) Jaguar Gimp-Print URL:www.allosx.com/1030154694/inde	9. (0.085) macosxhints - Get the most from URL:www.macosxhints.com	9. (0.330) VueScan 7.6.34 - VersionTracker URL:www.VersionTracker.com/dyn/mor
10. (0.006) Xamba URL:xamba.sourceforge.net/ssp	10. (0.081) Apple - fxbp URL:developer.apple.com/ja	10. (0.330) Apple iPod Updater 1.3 - Versio URL:www.VersionTracker.com/dyn/mor

AT-avg	Norm	BFS
1. (1.000) Apple iPod Updater 1.3 - Versio URL:www.VersionTracker.com/dyn/mor	1. (1.000) Apple iPod Updater 1.3 - Versio URL:www.VersionTracker.com/dyn/mor	1. (1.000) Team Franglais Home-Page (Anglo URL:franglais.8k.com
2. (1.000) Apple iTunes 4.0 - VersionTrack URL:www.VersionTracker.com/dyn/mor	2. (1.000) Apple iTunes 4.0 - VersionTrack URL:www.VersionTracker.com/dyn/mor	2. (0.983) The Atari Files URL:atarifiles.tripod.com
3. (1.000) VueScan 7.6.34 - VersionTracker URL:www.VersionTracker.com/dyn/mor	3. (1.000) VueScan 7.6.34 - VersionTracker URL:www.VersionTracker.com/dyn/mor	3. (0.977) **OSJI: ORIGINAL SPEC** **URL:www.osjimic.com**
4. (1.000) VueScan 7.6.34 - VersionTracker URL:www.VersionTracker.com/dyn/mor	4. (1.000) VueScan 7.6.34 - VersionTracker URL:www.VersionTracker.com/dyn/mor	4. (0.959) The Atarian Atmosphere URL:atmosphere.atariansun.com
5. (1.000) Apple iPod Updater 1.3 - Versio URL:www.VersionTracker.com/dyn/mor	5. (1.000) Apple iPod Updater 1.3 - Versio URL:www.VersionTracker.com/dyn/mor	5. (0.925) Fink - Home URL:fink.sourceforge.net
6. (1.000) PHP 4.3.2RC2 - VersionTracker URL:www.VersionTracker.com/dyn/mor	6. (1.000) PHP 4.3.2RC2 - VersionTracker URL:www.VersionTracker.com/dyn/mor	6. (0.921) Apple URL:www.apple.com
7. (1.000) Palm Desktop 4.1 - VersionTrack URL:www.VersionTracker.com/dyn/mor	7. (1.000) Palm Desktop 4.1 - VersionTrack URL:www.VersionTracker.com/dyn/mor	7. (0.913) Dreamweaver Templates - Fast - URL:www.dreamweaver-templates.net
8. (1.000) Apple - Games - Trailers URL:www.apple.com/games/trailers	8. (1.000) Apple - Games - Trailers URL:www.apple.com/games/trailers	8. (0.904) Emulation 4ever: The Emulation URL:emulation4ever.cjb.net
9. (0.975) Dantz Retrospect 5.0 Driver Upd URL:www.VersionTracker.com/dyn/mor	9. (0.976) Dantz Retrospect 5.0 Driver Upd URL:www.VersionTracker.com/dyn/mor	9. (0.857) Apple - Mac OS X URL:www.apple.com/macosx
10. (0.975) iView MediaPro 1.5.7 - VersionT URL:www.VersionTracker.com/dyn/mor	10. (0.976) iView MediaPro 1.5.7 - VersionT URL:www.VersionTracker.com/dyn/mor	10. (0.833) NoName Scriptware151;AppleScr URL:www.nonamescriptware.com

Table C.21: Query "jaguar"

Hits	PageRank	InDegree
1. (1.000) Welcome to MSN.com URL:g.msn.com/0nwenus0/AK/00	1. (1.000) Jordan Rudess : Feeding The Web URL:www.jordanrudess.com	1. (1.000) **Jordan Tourism Board** **URL:www.see-jordan.com**
2. (1.000) Welcome to MSN.com URL:g.msn.com/0nwenus0/AK/01	2. (0.965) *Jordan Tourism Board North* *URL:www.seejordan.org*	2. (0.707) Multitasking - multitasking.co URL:www.multitasking.com
3. (1.000) Empty title field URL:g.msn.com/0nwenus0/AK/02	3. (0.882) **Jordan Tourism Board** **URL:www.see-jordan.com**	3. (0.697) rowaq.com hosted at HostSave, t URL:www.rowaq.com
4. (1.000) MSN Search – More Useful Every URL:g.msn.com/0nwenus0/AK/03	4. (0.814) Site Meter - Counter and Statis URL:sm4.sitemeter.com/stats.asp?si	4. (0.678) *The Royal Autombile Club of Jor* *URL:www.racj.com*
5. (1.000) Welcome to MSN Shopping URL:g.msn.com/0nwenus0/AK/04	5. (0.787) Site Ask - What's your question URL:s12.sitemeter.com/stats.asp?si	5. (0.606) **National Information Center** **URL:www.nic.gov.jo**
6. (1.000) MSN Money - More Useful Everyda URL:g.msn.com/0nwenus0/AK/05	6. (0.585) Yahoo! URL:www.yahoo.com	6. (0.596) jordanzed.com URL:www.jordanzed.com
7. (1.000) MSN People and Chat - More Usef URL:g.msn.com/0nwenus0/AK/06	7. (0.485) TheCounter.com: The Full-Featur URL:www.TheCounter.com	7. (0.572) Home Page URL:www.lawtownmusic.8k.com
8. (1.000) Welcome to MSN.com URL:g.msn.com/0nwenus0/AK/14	8. (0.465) *Jordan Export Development* *URL:www.jedco.gov.jo*	8. (0.558) SheilaJordanJazz.com URL:www.sheilajordanjazz.com
9. (0.974) Welcome to MSN.com URL:g.msn.com/0nwenus0/AK/07	9. (0.454) Empty title field URL:www.bigbearvalleygallery.com	9. (0.558) Empty title field URL:www.bigbearvalleygallery.com
10. (0.974) Welcome to MSN.com URL:g.msn.com/0nwenus0/AK/08	10. (0.452) Multitasking - multitasking.co URL:www.multitasking.com	10. (0.505) stamps-by-year URL:stamps-of-jordan.tripod.com

HubAvg	Max	AT-med
1. (1.000) Yahoo! Directory URL:us.rd.yahoo.com/dir/yahoo/*htt	1. (1.000) **Jordan Tourism Board** **URL:www.see-jordan.com**	1. (1.000) **Jordan Tourism Board** **URL:www.see-jordan.com**
2. (1.000) Yahoo! URL:us.rd.yahoo.com/dir/yahoo/*htt	2. (0.468) *The Royal Autombile Club of Jor* *URL:www.racj.com*	2. (0.566) *The Royal Autombile Club of Jor* *URL:www.racj.com*
3. (1.000) Yahoo! Help - nbsp; URL:us.rd.yahoo.com/dir/help/*http	3. (0.429) **National Information Center** **URL:www.nic.gov.jo**	3. (0.522) **National Information Center** **URL:www.nic.gov.jo**
4. (0.032) Yahoo! Advanced Directory Searc URL:search.yahoo.com/dir/advanced	4. (0.262) **Jordan Embassy - U.S.A.** **URL:www.jordanembassyus.org**	4. (0.313) **Jordan Embassy - U.S.A.** **URL:www.jordanembassyus.org**
5. (0.006) Yahoo! Suggest a Site URL:us.rd.yahoo.com/dir/suggest/*h	5. (0.242) *Central Bank of Jordan Home Pag* *URL:www.cbj.gov.jo*	5. (0.292) *Central Bank of Jordan Home Pag* *URL:www.cbj.gov.jo*
6. (0.006) *Empty title field* *URL:us.rd.yahoo.com/dir/email/*htt*	6. (0.223) *Welcome to HIS ROYAL* *URL:www.princehassan.gov.jo*	6. (0.272) *Welcome to HIS ROYAL* *URL:www.princehassan.gov.jo*
7. (0.000) **Jordan Tourism Board** **URL:www.see-jordan.com**	7. (0.200) *RJ HOME* *URL:www.rja.com.jo*	7. (0.234) **Department Of Statistics** **URL:www.dos.gov.jo**
8. (0.000) *The Royal Autombile Club of Jor* *URL:www.racj.com*	8. (0.193) **Department Of Statistics** **URL:www.dos.gov.jo**	8. (0.221) *RJ HOME* *URL:www.rja.com.jo*
9. (0.000) **National Information Center** **URL:www.nic.gov.jo**	9. (0.189) *The University of Jordan's home* *URL:www.ju.edu.jo*	9. (0.219) *nbsp; – Jordan C* *URL:www.customs.gov.jo*
10. (0.000) rowaq.com hosted at HostSave, t URL:www.rowaq.com	10. (0.180) *nbsp; – Jordan C* *URL:www.customs.gov.jo*	10. (0.211) *The University of Jordan's home* *URL:www.ju.edu.jo*

AT-avg	Norm	BFS
1. (1.000) **Jordan Tourism Board** **URL:www.see-jordan.com**	1. (1.000) Welcome to MSN.com URL:g.msn.com/0nwenus0/AK/00	1. (1.000) Home Page URL:www.lawtownmusic.8k.com
2. (0.619) *The Royal Autombile Club of Jor* *URL:www.racj.com*	2. (1.000) Welcome to MSN.com URL:g.msn.com/0nwenus0/AK/01	2. (0.974) Multitasking - multitasking.co URL:www.multitasking.com
3. (0.583) **National Information Center** **URL:www.nic.gov.jo**	3. (1.000) Empty title field URL:g.msn.com/0nwenus0/AK/02	3. (0.956) **Jordan Tourism Board** **URL:www.see-jordan.com**
4. (0.355) **Jordan Embassy - U.S.A.** **URL:www.jordanembassyus.org**	4. (1.000) MSN Search – More Useful Every URL:g.msn.com/0nwenus0/AK/03	4. (0.938) rowaq.com hosted at HostSave, t URL:www.rowaq.com
5. (0.330) *Central Bank of Jordan Home Pag* *URL:www.cbj.gov.jo*	5. (1.000) Welcome to MSN Shopping URL:g.msn.com/0nwenus0/AK/04	5. (0.892) Empty title field URL:www.bigbearvalleygallery.com
6. (0.311) *Welcome to HIS ROYAL* *URL:www.princehassan.gov.jo*	6. (1.000) MSN Money - More Useful Everyda URL:g.msn.com/0nwenus0/AK/05	6. (0.889) *The Royal Autombile Club of Jor* *URL:www.racj.com*
7. (0.265) **Department Of Statistics** **URL:www.dos.gov.jo**	7. (1.000) MSN People and Chat - More Usef URL:g.msn.com/0nwenus0/AK/06	7. (0.885) jordanzed.com URL:www.jordanzed.com
8. (0.247) *nbsp; – Jordan C* *URL:www.customs.gov.jo*	8. (1.000) Welcome to MSN.com URL:g.msn.com/0nwenus0/AK/14	8. (0.878) SheilaJordanJazz.com URL:www.sheilajordanjazz.com
9. (0.235) *The University of Jordan's home* *URL:www.ju.edu.jo*	9. (0.965) Welcome to MSN.com URL:g.msn.com/0nwenus0/AK/07	9. (0.864) **National Information Center** **URL:www.nic.gov.jo**
10. (0.234) *RJ HOME* *URL:www.rja.com.jo*	10. (0.965) Welcome to MSN.com URL:g.msn.com/0nwenus0/AK/08	10. (0.836) **Jordan Embassy - U.S.A.** **URL:www.jordanembassyus.org**

Table C.22: Query "jordan"

Hits	PageRank	InDegree
1. (1.000) Long Distance Rate Finder .com URL:www.longdistanceratefinder.com	1. (1.000) Long Distance Rate Finder .com URL:www.longdistanceratefinder.com	1. (1.000) Long Distance Rate Finder .com URL:www.longdistanceratefinder.com
2. (0.893) Cognigen: Worldwide Telecommuni URL:longdist.net/?apl	2. (0.849) Cognigen: Worldwide Telecommuni URL:longdist.net/?apl	2. (0.803) *Empty title field* *URL:www.nasa.gov*
3. (0.892) BILLZilla - The Best Long Dista URL:www.billzilla.com/apl	3. (0.501) *Empty title field* *URL:www.nasa.gov*	3. (0.631) Real Estate Australia - Propert URL:www.realestate.com.au
4. (0.892) Talk America Local And Long Dis URL:cognigen.net/talkamerica/?apl	4. (0.226) Bushwhacked: Inside Stories of URL:www.conspiracydigest.com/urisb	4. (0.549) **Moon Hoax Index** **URL:www.redzero.demon.co.uk/moonho**
5. (0.892) OneStar Communications - Long D URL:cognigen.net/onestar/?apl	5. (0.220) Gannett Company, Inc. URL:www.gannett.com	5. (0.533) Cognigen: Worldwide Telecommuni URL:longdist.net/?apl
6. (0.892) CogniDial Discount Internationa URL:www.cognidial.com/dial-around/	6. (0.203) Real Estate Australia - Propert URL:www.realestate.com.au	6. (0.525) BILLZilla - The Best Long Dista URL:www.billzilla.com/apl
7. (0.892) Speakeasy High Speed Internet S URL:www.cognigen.net/speakeasy/?ap	7. (0.164) Yahoo! Privacy URL:privacy.yahoo.com/privacy/aa	7. (0.525) Talk America Local And Long Dis URL:cognigen.net/talkamerica/?apl
8. (0.892) DISH Network e-Store URL:cognigen.net/dish/?apl	8. (0.145) 1662;1585;1583;1607; URL:pardeh.blogspot.com	8. (0.525) OneStar Communications - Long D URL:cognigen.net/onestar/?apl
9. (0.892) Cognigen: Worldwide Telecommuni URL:ld.net/?apl	9. (0.142) **Phil Plait's Bad Astronomy: B** **URL:www.badastronomy.com/bad/tv/fo**	9. (0.525) CogniDial Discount Internationa URL:www.cognidial.com/dial-around/
10. (0.126) Exchange-it - Free Banner Excha URL:www.exchange-it.com/link.go?b1	10. (0.141) **Moon Hoax Index** **URL:www.redzero.demon.co.uk/moon**	10. (0.525) Speakeasy High Speed Internet S URL:www.cognigen.net/speakeasy/?ap

HubAvg	Max	AT-med
1. (1.000) Long Distance Rate Finder .com URL:www.longdistanceratefinder.com	1. (1.000) Long Distance Rate Finder .com URL:www.longdistanceratefinder.com	1. (1.000) Long Distance Rate Finder .com URL:www.longdistanceratefinder.com
2. (0.416) Cognigen: Worldwide Telecommuni URL:longdist.net/?apl	2. (0.529) Cognigen: Worldwide Telecommuni URL:longdist.net/?apl	2. (0.590) Cognigen: Worldwide Telecommuni URL:longdist.net/?apl
3. (0.410) BILLZilla - The Best Long Dista URL:www.billzilla.com/apl	3. (0.525) BILLZilla - The Best Long Dista URL:www.billzilla.com/apl	3. (0.586) BILLZilla - The Best Long Dista URL:www.billzilla.com/apl
4. (0.410) Talk America Local And Long Dis URL:cognigen.net/talkamerica/?apl	4. (0.525) Talk America Local And Long Dis URL:cognigen.net/talkamerica/?apl	4. (0.586) Talk America Local And Long Dis URL:cognigen.net/talkamerica/?apl
5. (0.410) OneStar Communications - Long D URL:cognigen.net/onestar/?apl	5. (0.525) OneStar Communications - Long D URL:cognigen.net/onestar/?apl	5. (0.586) OneStar Communications - Long D URL:cognigen.net/onestar/?apl
6. (0.410) CogniDial Discount Internationa URL:www.cognidial.com/dial-around/	6. (0.525) CogniDial Discount Internationa URL:www.cognidial.com/dial-around/	6. (0.586) CogniDial Discount Internationa URL:www.cognidial.com/dial-around/
7. (0.410) Speakeasy High Speed Internet S URL:www.cognigen.net/speakeasy/?ap	7. (0.525) Speakeasy High Speed Internet S URL:www.cognigen.net/speakeasy/?ap	7. (0.586) Speakeasy High Speed Internet S URL:www.cognigen.net/speakeasy/?ap
8. (0.410) DISH Network e-Store URL:cognigen.net/dish/?apl	8. (0.525) DISH Network e-Store URL:cognigen.net/dish/?apl	8. (0.586) DISH Network e-Store URL:cognigen.net/dish/?apl
9. (0.410) Cognigen: Worldwide Telecommuni URL:ld.net/?apl	9. (0.525) Cognigen: Worldwide Telecommuni URL:ld.net/?apl	9. (0.586) Cognigen: Worldwide Telecommuni URL:ld.net/?apl
10. (0.375) Name a Star - International Sta URL:click.linksynergy.com/fs-bin/s	10. (0.355) Name a Star - International Sta URL:click.linksynergy.com/fs-bin/s	10. (0.338) Name a Star - International Sta URL:click.linksynergy.com/fs-bin/s

AT-avg	Norm	BFS
1. (1.000) Long Distance Rate Finder .com URL:www.longdistanceratefinder.com	1. (1.000) Long Distance Rate Finder .com URL:www.longdistanceratefinder.com	1. (1.000) *Empty title field* *URL:www.nasa.gov*
2. (0.758) Cognigen: Worldwide Telecommuni URL:longdist.net/?apl	2. (0.711) Cognigen: Worldwide Telecommuni URL:longdist.net/?apl	2. (0.862) **Moon Hoax Index** **URL:www.redzero.demon.co.uk/moonho**
3. (0.756) BILLZilla - The Best Long Dista URL:www.billzilla.com/apl	3. (0.708) BILLZilla - The Best Long Dista URL:www.billzilla.com/apl	3. (0.831) **Phil Plait's Bad Astronomy: H** **URL:www.badastronomy.com**
4. (0.756) Talk America Local And Long Dis URL:cognigen.net/talkamerica/?apl	4. (0.708) Talk America Local And Long Dis URL:cognigen.net/talkamerica/?apl	4. (0.828) **Phil Plait's Bad Astronomy: B** **URL:www.badastronomy.com/bad/tv/fo**
5. (0.756) OneStar Communications - Long D URL:cognigen.net/onestar/?apl	5. (0.708) OneStar Communications - Long D URL:cognigen.net/onestar/?apl	5. (0.799) **The Moon Shots Were Faked** **URL:batesmotel.8m.com**
6. (0.756) CogniDial Discount Internationa URL:www.cognidial.com/dial-around/	6. (0.708) CogniDial Discount Internationa URL:www.cognidial.com/dial-around/	6. (0.723) **NASA Apollo 11 30th** **URL:www.hq.nasa.gov/office/pao/His**
7. (0.756) Speakeasy High Speed Internet S URL:www.cognigen.net/speakeasy/?ap	7. (0.708) Speakeasy High Speed Internet S URL:www.cognigen.net/speakeasy/?ap	7. (0.716) **The Great Moon Hoax** **URL:science.nasa.gov/headlines/y20**
8. (0.756) DISH Network e-Store URL:cognigen.net/dish/?apl	8. (0.708) DISH Network e-Store URL:cognigen.net/dish/?apl	8. (0.705) **Faked Moon Landings?** **URL:www.apollo-hoax.co.uk**
9. (0.756) Cognigen: Worldwide Telecommuni URL:ld.net/?apl	9. (0.708) Cognigen: Worldwide Telecommuni URL:ld.net/?apl	9. (0.705) *Funny Thing Happened on the Way* *URL:www.moonmovie.com*
10. (0.214) Name a Star - International Sta URL:click.linksynergy.com/fs-bin/s	10. (0.238) Name a Star - International Sta URL:click.linksynergy.com/fs-bin/s	10. (0.700) **The Moon Landings Were** **URL:pirlwww.lpl.arizona.edu/∼jsc**

Table C.23: Query "moon landing"

Hits	PageRank	InDegree
1. (1.000) **The Internet Movie Database** **URL:www.imdb.com**	1. (1.000) *Amazon.com–Earth's Biggest Sel* *URL:www.amazon.com/exec/obidos/red*	1. (1.000) **The Internet Movie Database** **URL:www.imdb.com**
2. (0.883) DHTML Lab: HierMenus URL:www.hiermenuscentral.com	2. (0.326) **The Internet Movie Database** **URL:www.imdb.com**	2. (0.565) Google URL:www.google.com
3. (0.796) internet.com: the Internet and URL:www.internet.com	3. (0.261) Knight Ridder Corporate Web sit URL:www.knightridder.com	3. (0.534) Signs on DVD URL:www.signs.movies.com
4. (0.779) WebReference.com - The Webmaste URL:webreference.com	4. (0.255) AllPosters.com Affiliates Home URL:affiliates.allposters.com/link	4. (0.471) *Amazon.com–Earth's Biggest Sel* *URL:www.amazon.com/exec/obidos/red*
5. (0.767) Welcome to internet.com's Devel URL:www.internet.com/sections/webd	5. (0.224) Google URL:www.google.com	5. (0.365) Get Wild - GetWild - getwild.co URL:www.getwild.com
6. (0.757) Jupitermedia Corporation Web Si URL:www.internet.com/corporate/leg	6. (0.217) Gannett Company, Inc. URL:www.gannett.com	6. (0.321) Gannett Company, Inc. URL:www.gannett.com
7. (0.757) Jupitermedia Privacy Policy URL:www.internet.com/corporate/pri	7. (0.214) CapeWeek: Arts Entertainment URL:www.capeweek.com	7. (0.308) **Empty title field** **URL:www.film.com**
8. (0.757) internet.com Commerce Partners URL:www.internet.com/partners	8. (0.205) Yahoo! Terms of Service URL:docs.yahoo.com/info/terms	8. (0.296) **Hollywood.com - Your entertai** **URL:www.hollywood.com**
9. (0.757) Search Internet.com URL:search.internet.com	9. (0.200) EarthWeb.com: The IT Industry P URL:www.earthweb.com	9. (0.275) Knight Ridder Corporate Web sit URL:www.knightridder.com
10. (0.754) internet.com Media Kit URL:www.internet.com/mediakit	10. (0.165) Apple .Mac Welcome URL:www.mac.com	10. (0.272) *A Beautiful Mind* *URL:abeautifulmind.com*

HubAvg	Max	AT-med
1. (1.000) *Amazon.com–Earth's Biggest Sel* *URL:www.amazon.com/exec/obidos/red*	1. (1.000) **The Internet Movie Database** **URL:www.imdb.com**	1. (1.000) **The Internet Movie Database** **URL:www.imdb.com**
2. (0.000) **The Internet Movie Database** **URL:www.imdb.com**	2. (0.296) Signs on DVD URL:www.signs.movies.com	2. (0.366) Signs on DVD URL:www.signs.movies.com
3. (0.000) Google URL:www.google.com	3. (0.253) Google URL:www.google.com	3. (0.309) Google URL:www.google.com
4. (0.000) CNI Newspapers: News Front Page URL:www.cninewsonline.com	4. (0.219) **Hollywood.com - Your entertai** **URL:www.hollywood.com**	4. (0.248) **Hollywood.com - Your entertai** **URL:www.hollywood.com**
5. (0.000) JS Online: General Information URL:graphics.jsonline.com/adsectio	5. (0.211) **Empty title field** **URL:www.film.com**	5. (0.241) **Empty title field** **URL:www.film.com**
6. (0.000) Signs on DVD URL:www.signs.movies.com	6. (0.174) Get Wild - GetWild - getwild.co URL:www.getwild.com	6. (0.208) Get Wild - GetWild - getwild.co URL:www.getwild.com
7. (0.000) **Hollywood.com - Your entertai** **URL:www.hollywood.com**	7. (0.161) **All Movie Guide** **URL:www.allmovie.com**	7. (0.170) **All Movie Guide** **URL:www.allmovie.com**
8. (0.000) **Empty title field** **URL:www.film.com**	8. (0.159) **Movie Review Query Engine** **URL:www.mrqe.com**	8. (0.168) **ROTTEN TOMATOES: Movie** **URL:www.rottentomatoes.com**
9. (0.000) **ROTTEN TOMATOES: Movie** **URL:www.rottentomatoes.com**	9. (0.159) **ROTTEN TOMATOES: Movie** **URL:www.rottentomatoes.com**	9. (0.168) **Movie Review Query Engine** **URL:www.mrqe.com**
10. (0.000) Get Wild - GetWild - getwild.co URL:www.getwild.com	10. (0.142) **Greatest Films** **URL:www.filmsite.org**	10. (0.154) **Greatest Films** **URL:www.filmsite.org**

AT-avg	Norm	BFS
1. (1.000) **The Internet Movie Database** **URL:www.imdb.com**	1. (1.000) **The Internet Movie Database** **URL:www.imdb.com**	1. (1.000) **The Internet Movie Database** **URL:www.imdb.com**
2. (0.426) Signs on DVD URL:www.signs.movies.com	2. (0.343) Signs on DVD URL:www.signs.movies.com	2. (0.901) Signs on DVD URL:www.signs.movies.com
3. (0.335) Google URL:www.google.com	3. (0.283) Google URL:www.google.com	3. (0.832) *A Beautiful Mind* *URL:abeautifulmind.com*
4. (0.288) **Hollywood.com - Your entertai** **URL:www.hollywood.com**	4. (0.244) **Hollywood.com - Your entertai** **URL:www.hollywood.com**	4. (0.821) Get Wild - GetWild - getwild.co URL:www.getwild.com
5. (0.279) **Empty title field** **URL:www.film.com**	5. (0.234) **Empty title field** **URL:www.film.com**	5. (0.814) Google URL:www.google.com
6. (0.243) Get Wild - GetWild - getwild.co URL:www.getwild.com	6. (0.202) Get Wild - GetWild - getwild.co URL:www.getwild.com	6. (0.776) **New Line Cinema** **URL:www.newline.com**
7. (0.186) **All Movie Guide** **URL:www.allmovie.com**	7. (0.170) **All Movie Guide** **URL:www.allmovie.com**	7. (0.746) *Spider-Man Movie From Columbia* *URL:www.spiderman.sonypictures.com*
8. (0.181) **ROTTEN TOMATOES: Movie** **URL:www.rottentomatoes.com**	8. (0.167) **ROTTEN TOMATOES: Movie** **URL:www.rottentomatoes.com**	8. (0.731) Poor Roger's Home Page - Nothin URL:www.poorroger.com
9. (0.180) **Movie Review Query Engine** **URL:www.mrqe.com**	9. (0.167) **Movie Review Query Engine** **URL:www.mrqe.com**	9. (0.730) **Hollywood.com - Your entertai** **URL:www.hollywood.com**
10. (0.173) **Paramount Pictures** **URL:www.paramount.com**	10. (0.153) **Greatest Films** **URL:www.filmsite.org**	10. (0.730) **Empty title field** **URL:www.film.com**

Table C.24: Query "movies"

Hits	PageRank	InDegree
1. (1.000) E Business Solutions,Website Pr URL:www.intermesh.net/advertis.htm	1. (1.000) **National Park Service - Exper** **URL:www.nps.gov**	1. (1.000) **National Park Service - Exper** **URL:www.nps.gov**
2. (0.998) Empty title field URL:www.indiangiftsportal.com	2. (0.560) FirstGov 8212; Your First Cli URL:www.firstgov.gov	2. (0.358) E Business Solutions,Website Pr URL:www.intermesh.net/advertis.htm
3. (0.998) Business Solutions,Ecommerce Bu URL:www.intermesh.net	3. (0.375) Welcome to the White House URL:www.whitehouse.gov	3. (0.358) Empty title field URL:news.indiamart.com
4. (0.998) Empty title field URL:news.indiamart.com	4. (0.271) egov – The Official Web Site o URL:egov.gov	4. (0.356) Empty title field URL:www.indiamart.com
5. (0.997) Empty title field URL:www.indiamart.com	5. (0.243) **NatureNet: The National Park** **URL:www.nature.nps.gov**	5. (0.356) Empty title field URL:apparel.indiamart.com
6. (0.997) Empty title field URL:apparel.indiamart.com	6. (0.219) **National Recreation and Park** **URL:www.nrpa.org**	6. (0.356) Empty title field URL:www.indiangiftsportal.com
7. (0.997) Empty title field URL:handicraft.indiamart.com	7. (0.213) Take Pride In America - Home URL:www.takepride.gov	7. (0.356) Empty title field URL:handicraft.indiamart.com
8. (0.997) India Finance and Investment Gu URL:finance.indiamart.com	8. (0.206) **Western National Parks Assoca** **URL:www.wnpa.org**	8. (0.356) India Finance and Investment Gu URL:finance.indiamart.com
9. (0.997) Empty title field URL:health.indiamart.com	9. (0.199) CoolWorks.com Summer Jobs URL:www.coolworks.com	9. (0.356) Empty title field URL:health.indiamart.com
10. (0.997) Empty title field URL:auto.indiamart.com	10. (0.196) **Links to the Past: National** **URL:www.cr.nps.gov**	10. (0.356) Empty title field URL:auto.indiamart.com

HubAvg	Max	AT-med
1. (1.000) **National Park Service - Exper** **URL:www.nps.gov**	1. (1.000) **National Park Service - Exper** **URL:www.nps.gov**	1. (1.000) **National Park Service - Exper** **URL:www.nps.gov**
2. (0.138) Privacy Statement, National Par URL:www.nps.gov/privacy.htm	2. (0.165) Privacy Statement, National Par URL:www.nps.gov/privacy.htm	2. (0.183) Privacy Statement, National Par URL:www.nps.gov/privacy.htm
3. (0.065) *Park Geology Tour - Geologic Fe* *URL:www.nature.nps.gov/grd/tour*	3. (0.101) *GORP.com-adventure travel-hikin* *URL:www.gorp.com*	3. (0.110) *GORP.com-adventure travel-hikin* *URL:www.gorp.com*
4. (0.058) USGS Western Earth Surface Proc URL:wrgis.wr.usgs.gov/wgmt	4. (0.101) *Park Geology Tour - Geologic Fe* *URL:www.nature.nps.gov/grd/tour*	4. (0.105) *Park Geology Tour - Geologic Fe* *URL:www.nature.nps.gov/grd/tour*
5. (0.054) **NPS Search Portal** **URL:www.nps.gov/search.htm**	5. (0.088) USGS Western Earth Surface Proc URL:wrgis.wr.usgs.gov/wgmt	5. (0.093) **NPS Search Portal** **URL:www.nps.gov/search.htm**
6. (0.051) **NatureNet: The National Park** **URL:www.nature.nps.gov**	6. (0.084) **NPS Search Portal** **URL:www.nps.gov/search.htm**	6. (0.092) USGS Western Earth Surface Proc URL:wrgis.wr.usgs.gov/wgmt
7. (0.042) *GORP.com-adventure travel-hikin* *URL:www.gorp.com*	7. (0.071) **National Park Guide** **URL:www.nps.gov/parks.html**	7. (0.078) **National Park Guide** **URL:www.nps.gov/parks.html**
8. (0.025) **National Park Guide** **URL:www.nps.gov/parks.html**	8. (0.061) **NatureNet: The National Park** **URL:www.nature.nps.gov**	8. (0.061) **NatureNet: The National Park** **URL:www.nature.nps.gov**
9. (0.021) **National Parks Conservation A** **URL:www.npca.org**	9. (0.055) **National Parks Conservation A** **URL:www.npca.org**	9. (0.058) **National Parks Conservation A** **URL:www.npca.org**
10. (0.016) **L.L.Bean - Park Search** **URL:www.llbean.com/parksearch**	10. (0.052) **L.L.Bean - Park Search** **URL:www.llbean.com/parksearch**	10. (0.054) **L.L.Bean - Park Search** **URL:www.llbean.com/parksearch**

AT-avg	Norm	BFS
1. (1.000) E Business Solutions,Website Pr URL:www.intermesh.net/advertis.htm	1. (1.000) E Business Solutions,Website Pr URL:www.intermesh.net/advertis.htm	1. (1.000) **National Park Service - Exper** **URL:www.nps.gov**
2. (0.996) Empty title field URL:news.indiamart.com	2. (0.997) Empty title field URL:news.indiamart.com	2. (0.733) Sw Parks - SwParks - swparks.co URL:www.swparks.com
3. (0.994) Empty title field URL:www.indiamart.com	3. (0.996) Empty title field URL:www.indiangiftsportal.com	3. (0.665) **U.S. National Parks - Welcome** **URL:www.us-national-parks.net**
4. (0.994) Empty title field URL:apparel.indiamart.com	4. (0.996) Business Solutions,Ecommerce Bu URL:www.intermesh.net	4. (0.664) **National Parks Conservation A** **URL:www.npca.org**
5. (0.994) Empty title field URL:handicraft.indiamart.com	5. (0.996) Empty title field URL:www.indiamart.com	5. (0.659) *The EnviroLink Network* *URL:www.envirolink.org*
6. (0.994) India Finance and Investment Gu URL:finance.indiamart.com	6. (0.996) Empty title field URL:apparel.indiamart.com	6. (0.659) Yahoo! URL:www.yahoo.com
7. (0.994) Empty title field URL:health.indiamart.com	7. (0.996) Empty title field URL:handicraft.indiamart.com	7. (0.644) *GORP.com-adventure travel-hikin* *URL:www.gorp.com*
8. (0.994) Empty title field URL:auto.indiamart.com	8. (0.996) India Finance and Investment Gu URL:finance.indiamart.com	8. (0.642) **National Park Guide** **URL:www.nps.gov/parks.html**
9. (0.994) Empty title field URL:www.indiangiftsportal.com	9. (0.996) Empty title field URL:health.indiamart.com	9. (0.640) One stop shopping for residenti URL:www.jasperrealestate.ab.ca
10. (0.994) Business Solutions,Ecommerce Bu URL:www.intermesh.net	10. (0.996) Empty title field URL:auto.indiamart.com	10. (0.628) **Australian Alps national par** **URL:www.australianalps.environme**

Table C.25: Query “national parks”

Hits	PageRank	InDegree
1. (1.000) movabletype.org URL:www.movabletype.org	1. (1.000) ArticleCentral - Content and Ar URL:articlecentral.com	1. (1.000) **EFF: Homepage** **URL:www.eff.org**
2. (0.952) Boing Boing: A Directory of Won URL:boingboing.net	2. (0.867) Newspapers OnLine URL:adsearch.chron.com	2. (0.656) **Internet Free Expression Alli** **URL:www.ifea.net**
3. (0.947) Metafilter — Community Weblog URL:www.metafilter.com	3. (0.726) **EFF: Homepage** **URL:www.eff.org**	3. (0.632) **American Civil Liberties Unio** **URL:www.aclu.org**
4. (0.931) Wired News URL:www.wired.com	4. (0.708) HoustonChronicle.com - News URL:www.houstonchronicle.com	4. (0.620) **The Center for Democracy and** **URL:www.cdt.org**
5. (0.930) The Doc Searls Weblog : Sunday, URL:doc.weblogs.com	5. (0.536) H2K2 URL:www.h2k2.net	5. (0.571) **P E A C E F I R E** **URL:www.peacefire.org**
6. (0.924) what's in rebecca's pocket? URL:www.rebeccablood.net	6. (0.533) *Index on Censorship: Latest New* *URL:www.indexonline.org*	6. (0.564) Vtw Directory Page URL:www.vtw.org
7. (0.911) InstaPundit.Com URL:www.instapundit.com	7. (0.502) movabletype.org URL:www.movabletype.org	7. (0.546) movabletype.org URL:www.movabletype.org
8. (0.910) blogdex - the weblog diffusion URL:blogdex.media.mit.edu	8. (0.472) Free-Market.Net ... Information URL:www.free-market.net	8. (0.448) **libertus.net: about censorshi** **URL:libertus.net**
9. (0.910) kottke.org :: home of fine hype URL:www.kottke.org	9. (0.468) **EFF Blue Ribbon Campaign** **URL:www.eff.org/blueribbon.html**	9. (0.429) **EFF Blue Ribbon Campaign** **URL:www.eff.org/blueribbon.html**
10. (0.907) kuro5hin.org —— technology and URL:www.kuro5hin.org	10. (0.454) H2K2 - HOPE 2002 URL:store.2600.com/h2k2hope2002.ht	10. (0.380) **Global Internet Liberty Camp** **URL:www.gilc.org**

HubAvg	Max	AT-med
1. (1.000) ArticleCentral - Content and Ar URL:articlecentral.com	1. (1.000) **EFF: Homepage** **URL:www.eff.org**	1. (1.000) **EFF: Homepage** **URL:www.eff.org**
2. (0.016) Microsoft Corporation URL:www.microsoft.com	2. (0.541) **Internet Free Expression Alli** **URL:www.ifea.net**	2. (0.631) **Internet Free Expression Alli** **URL:www.ifea.net**
3. (0.009) **EFF: Homepage** **URL:www.eff.org**	3. (0.517) **The Center for Democracy and** **URL:www.cdt.org**	3. (0.606) **The Center for Democracy and** **URL:www.cdt.org**
4. (0.007) NabaviOnline URL:www.nabavionline.com	4. (0.517) **American Civil Liberties Unio** **URL:www.aclu.org**	4. (0.600) **American Civil Liberties Unio** **URL:www.aclu.org**
5. (0.004) Vtw Directory Page URL:www.vtw.org	5. (0.386) Vtw Directory Page URL:www.vtw.org	5. (0.421) Vtw Directory Page URL:www.vtw.org
6. (0.003) **Internet Free Expression Alli** **URL:www.ifea.net**	6. (0.357) **P E A C E F I R E** **URL:www.peacefire.org**	6. (0.388) **P E A C E F I R E** **URL:www.peacefire.org**
7. (0.003) **The Center for Democracy and** **URL:www.cdt.org**	7. (0.277) **Global Internet Liberty Campa** **URL:www.gilc.org**	7. (0.313) **Global Internet Liberty Campa** **URL:www.gilc.org**
8. (0.003) **American Civil Liberties Unio** **URL:www.aclu.org**	8. (0.254) **libertus.net: about censorshi** **URL:libertus.net**	8. (0.276) **libertus.net: about censorshi** **URL:libertus.net**
9. (0.002) **EFF Blue Ribbon Campaign** **URL:www.eff.org/blueribbon.html**	9. (0.196) **EFF Blue Ribbon Campaign** **URL:www.eff.org/blueribbon.html**	9. (0.212) **EFF Blue Ribbon Campaign** **URL:www.eff.org/blueribbon.html**
10. (0.002) **P E A C E F I R E** **URL:www.peacefire.org**	10. (0.144) *The Freedom Forum* *URL:www.freedomforum.org*	10. (0.165) *The Freedom Forum* *URL:www.freedomforum.org*

AT-avg	Norm	BFS
1. (1.000) **EFF: Homepage** **URL:www.eff.org**	1. (1.000) **EFF: Homepage** **URL:www.eff.org**	1. (1.000) **EFF: Homepage** **URL:www.eff.org**
2. (0.716) **Internet Free Expression Alli** **URL:www.ifea.net**	2. (0.623) **Internet Free Expression Alli** **URL:www.ifea.net**	2. (0.945) **American Civil Liberties Unio** **URL:www.aclu.org**
3. (0.712) **The Center for Democracy and** **URL:www.cdt.org**	3. (0.607) **The Center for Democracy and** **URL:www.cdt.org**	3. (0.910) **Internet Free Expression Alli** **URL:www.ifea.net**
4. (0.676) **American Civil Liberties Unio** **URL:www.aclu.org**	4. (0.594) **American Civil Liberties Unio** **URL:www.aclu.org**	4. (0.869) **The Center for Democracy and** **URL:www.cdt.org**
5. (0.457) Vtw Directory Page URL:www.vtw.org	5. (0.419) Vtw Directory Page URL:www.vtw.org	5. (0.846) **P E A C E F I R E** **URL:www.peacefire.org**
6. (0.414) **P E A C E F I R E** **URL:www.peacefire.org**	6. (0.388) **P E A C E F I R E** **URL:www.peacefire.org**	6. (0.801) Vtw Directory Page URL:www.vtw.org
7. (0.359) **Global Internet Liberty Campa** **URL:www.gilc.org**	7. (0.312) **Global Internet Liberty Campa** **URL:www.gilc.org**	7. (0.796) **libertus.net: about censorshi** **URL:libertus.net**
8. (0.308) **libertus.net: about censorshi** **URL:libertus.net**	8. (0.281) **libertus.net: about censorshi** **URL:libertus.net**	8. (0.791) **Global Internet Liberty Campa** **URL:www.gilc.org**
9. (0.214) **EFF Blue Ribbon Campaign** **URL:www.eff.org/blueribbon.html**	9. (0.208) **EFF Blue Ribbon Campaign** **URL:www.eff.org/blueribbon.html**	9. (0.730) **EFF Blue Ribbon Campaign** **URL:www.eff.org/blueribbon.html**
10. (0.198) *The Freedom Forum* *URL:www.freedomforum.org*	10. (0.169) *The Freedom Forum* *URL:www.freedomforum.org*	10. (0.677) Google URL:www.google.com

Table C.26: Query "net censorship"

Hits	PageRank	InDegree
1. (1.000) *SpringerLink: Lecture Notes in* *URL:link.springer.de/link/service/*	1. (1.000) *MFCS'98 home page* *URL:www.fi.muni.cz/mfcs98*	1. (1.000) *Algorithms Courses on the WWW* *URL:www.cs.pitt.edu/∼kirk/algorith*
2. (1.000) *SpringerLink: Lecture Notes in* *URL:link.springer.de/link/service/*	2. (0.988) *Computational Geometry, Algorit* *URL:www.cs.uu.nl/geobook*	2. (1.000) *Computational Geometry, Algorit* *URL:www.cs.uu.nl/geobook*
3. (1.000) SpringerLink: Lecture Notes in URL:link.springer.de/link/service/	3. (0.884) The Digital Object Identifier URL:www.doi.org	3. (0.878) *MFCS'98 home page* *URL:www.fi.muni.cz/mfcs98*
4. (0.989) *ALGO 2002* *URL:www.dis.uniroma1.it/∼algo02*	4. (0.738) *Algorithms Courses on the WWW* *URL:www.cs.pitt.edu/∼kirk/algorith*	4. (0.780) **HTML redirection** **URL:cui.unige.ch/tcs/random-approx**
5. (0.301) Welcome to Springer, springer-v URL:www.springer.de	5. (0.735) **RAND-APX Thematic Net** **URL:www.maths.ox.ac.uk/rand-apx**	5. (0.537) *MHHE: INTRODUCTION TO* *URL:www.mhhe.com/catalogs/00701315*
6. (0.148) Mark Overmars Homepage URL:www.cs.uu.nl/people/markov	6. (0.704) **Home Page for RAND-APX** **URL:www.cs.lth.se/home/Andrzej_Lin**	6. (0.537) The Digital Object Identifier URL:www.doi.org
7. (0.135) *Pankaj K. Agarwal's Home Page* *URL:www.cs.duke.edu/∼pankaj*	7. (0.653) *MHHE: INTRODUCTION TO* *URL:www.mhhe.com/catalogs/00701315*	7. (0.512) Masaryk University Brno URL:www.muni.cz
8. (0.132) *Thomas H. Cormen* *URL:www.cs.dartmouth.edu/∼thc*	8. (0.643) **HTML redirection** **URL:cui.unige.ch/tcs/random-approx**	8. (0.488) *ALGO 2002* *URL:www.dis.uniroma1.it/∼algo02*
9. (0.086) *Algorithms Courses on the WWW* *URL:www.cs.pitt.edu/∼kirk/algorith*	9. (0.614) **APPROX 2001 + RANDOM** **URL:www.cs.princeton.edu/random-ap**	9. (0.463) *SpringerLink: Lecture Notes in* *URL:link.springer.de/link/service/*
10. (0.070) *WAE '98* *URL:www.mpi-sb.mpg.de/∼wae98*	10. (0.610) ERCIM, The European Research URL:www.ercim.org	10. (0.463) *SpringerLink: Lecture Notes in* *URL:link.springer.de/link/service/*

HubAvg	Max	AT-med
1. (1.000) *Computational Geometry, Algorit* *URL:www.cs.uu.nl/geobook*	1. (1.000) *Algorithms Courses on the WWW* *URL:www.cs.pitt.edu/∼kirk/algorith*	1. (1.000) *Algorithms Courses on the WWW* *URL:www.cs.pitt.edu/∼kirk/algorith*
2. (0.117) Directory of Computational Geom URL:www.geom.umn.edu/software/cgli	2. (1.000) *Computational Geometry, Algorit* *URL:www.cs.uu.nl/geobook*	2. (1.000) *Computational Geometry, Algorit* *URL:www.cs.uu.nl/geobook*
3. (0.068) The former CGAL home page URL:www.cs.uu.nl/CGAL	3. (0.270) Directory of Computational Geom URL:www.geom.umn.edu/software/cgli	3. (0.270) Directory of Computational Geom URL:www.geom.umn.edu/software/cgli
4. (0.046) Welcome to Springer, springer-v URL:www.springer.de	4. (0.258) LEDA moved to Algorithmic Solut URL:www.mpi-sb.mpg.de/LEDA/leda.ht	4. (0.258) LEDA moved to Algorithmic Solut URL:www.mpi-sb.mpg.de/LEDA/leda.ht
5. (0.039) LEDA moved to Algorithmic Solut URL:www.mpi-sb.mpg.de/LEDA/leda.ht	5. (0.257) *ANALYSIS of ALGORITHMS* *URL:pauillac.inria.fr/algo/AofA*	5. (0.257) *ANALYSIS of ALGORITHMS* *URL:pauillac.inria.fr/algo/AofA*
6. (0.038) CMSC 754 - Comp Geom URL:www.cs.umd.edu/∼mount/754	6. (0.237) IEEE Computer Society URL:computer.org	6. (0.237) IEEE Computer Society URL:computer.org
7. (0.031) Mark Overmars Homepage URL:www.cs.uu.nl/people/markov	7. (0.205) *Center for Discrete Mathematics* *URL:dimacs.rutgers.edu*	7. (0.205) *Center for Discrete Mathematics* *URL:dimacs.rutgers.edu*
8. (0.024) *Cormen/Leiserson/Rivest/Stein:* *URL:theory.lcs.mit.edu/∼clr*	8. (0.183) *MFCS'98 home page* *URL:www.fi.muni.cz/mfcs98*	8. (0.183) *MFCS'98 home page* *URL:www.fi.muni.cz/mfcs98*
9. (0.024) *Algorithms Courses on the WWW* *URL:www.cs.pitt.edu/∼kirk/algorith*	9. (0.182) Computer Science Papers NEC Res URL:citeseer.nj.nec.com/cs	9. (0.182) Computer Science Papers NEC Res URL:citeseer.nj.nec.com/cs
10. (0.017) *Computational Geometry, Algorit* *URL:www.cs.uu.nl/geobook/geom.html*	10. (0.178) Welcome to Springer, springer-v URL:www.springer.de	10. (0.178) Welcome to Springer, springer-v URL:www.springer.de

AT-avg	Norm	BFS
1. (1.000) *Algorithms Courses on the WWW* *URL:www.cs.pitt.edu/∼kirk/algorith*	1. (1.000) *Algorithms Courses on the WWW* *URL:www.cs.pitt.edu/∼kirk/algorith*	1. (1.000) *Algorithms Courses on the WWW* *URL:www.cs.pitt.edu/∼kirk/algorith*
2. (0.417) *Computational Geometry, Algorit* *URL:www.cs.uu.nl/geobook*	2. (0.444) *Computational Geometry, Algorit* *URL:www.cs.uu.nl/geobook*	2. (0.956) *ANALYSIS of ALGORITHMS* *URL:pauillac.inria.fr/algo/AofA*
3. (0.306) *ANALYSIS of ALGORITHMS* *URL:pauillac.inria.fr/algo/AofA*	3. (0.287) *ANALYSIS of ALGORITHMS* *URL:pauillac.inria.fr/algo/AofA*	3. (0.907) Computer Science Papers NEC Res URL:citeseer.nj.nec.com/cs
4. (0.280) LEDA moved to Algorithmic Solut URL:www.mpi-sb.mpg.de/LEDA/leda.ht	4. (0.261) LEDA moved to Algorithmic Solut URL:www.mpi-sb.mpg.de/LEDA/leda.ht	4. (0.901) *Center for Discrete Mathematics* *URL:dimacs.rutgers.edu*
5. (0.267) IEEE Computer Society URL:computer.org	5. (0.255) IEEE Computer Society URL:computer.org	5. (0.865) **HTML redirection** **URL:cui.unige.ch/tcs/random-approx**
6. (0.248) *Center for Discrete Mathematics* *URL:dimacs.rutgers.edu*	6. (0.233) *Center for Discrete Mathematics* *URL:dimacs.rutgers.edu*	6. (0.849) Welcome to Springer, springer-v URL:www.springer.de
7. (0.218) *MFCS'98 home page* *URL:www.fi.muni.cz/mfcs98*	7. (0.230) *MFCS'98 home page* *URL:www.fi.muni.cz/mfcs98*	7. (0.848) *MFCS'98 home page* *URL:www.fi.muni.cz/mfcs98*
8. (0.215) Directory of Computational Geom URL:www.geom.umn.edu/software/cgli	8. (0.193) Directory of Computational Geom URL:www.geom.umn.edu/software/cgli	8. (0.831) LEDA moved to Algorithmic Solut URL:www.mpi-sb.mpg.de/LEDA/leda.ht
9. (0.198) Computer Science Papers NEC Res URL:citeseer.nj.nec.com/cs	9. (0.186) Computer Science Papers NEC Res URL:citeseer.nj.nec.com/cs	9. (0.747) CiteSeer: The NEC Research Inst URL:citeseer.org
10. (0.175) CiteSeer: The NEC Research Inst URL:citeseer.org	10. (0.168) CiteSeer: The NEC Research Inst URL:citeseer.org	10. (0.739) *Computational Geometry, Algorit* *URL:www.cs.uu.nl/geobook*

Table C.27: Query "randomized algorithms"

Hits	PageRank	InDegree
1. (1.000) HonoluluAdvertiser.com gt; Haw URL:www.hawaiisclassifieds.com	1. (1.000) **Low-Fat Recipes - Home** **URL:www.low-fat-recipes.com**	1. (1.000) **EPICURIOUS: WORLD'S** **URL:www.epicurious.com**
2. (0.999) Gannett Company, Inc. URL:www.gannett.com	2. (0.627) Cookingaffiliates.com Home URL:www.cookingaffiliates.com	2. (0.886) *Food Network* *URL:www.foodtv.com*
3. (0.998) AP MoneyWire URL:apmoneywire.mm.ap.org	3. (0.613) Cutting-edge natural health tre URL:www.youngagain.com	3. (0.847) **All Recipes — Recipes** **URL:www.allrecipes.com**
4. (0.990) e.thePeople : Honolulu Advertis URL:www.e-thepeople.com/affiliates	4. (0.557) Shopping at Cooking.com: Find s URL:www.cooking.com	4. (0.792) **RecipeSource: Your Source for** **URL:www.recipesource.com**
5. (0.989) News From The Associated Press URL:customwire.ap.org/dynamic/fron	5. (0.535) The Honolulu Advertiser - Hawai URL:www.honoluluadvertiser.com	5. (0.699) What You Need to Know About84 URL:www.about.com
6. (0.987) Honolulu Traffic Cameras, City URL:www.co.honolulu.hi.us/cameras/	6. (0.507) National Honey Board@-I URL:www.nhb.jp	6. (0.682) **VegWeb - Vegan/Vegetarian** **URL:www.vegweb.com**
7. (0.987) News From The Associated Press URL:customwire.ap.org/dynamic/fron	7. (0.484) *Food Network* *URL:www.foodtv.com*	7. (0.669) **FATFREE: The Low Fat** **URL:www.fatfree.com**
8. (0.987) News From The Associated Press URL:customwire.ap.org/dynamic/fron	8. (0.470) Natural Progesterone Cream, 2 o URL:www.youngagain.com/progcream10	8. (0.669) Cutting-edge natural health tre URL:www.youngagain.com
9. (0.987) News From The Associated Press URL:customwire.ap.org/dynamic/fron	9. (0.417) Honey Locator URL:www.honeylocator.com	9. (0.631) **Top Secret Recipes on the Web** **URL:www.topsecretrecipes.com**
10. (0.987) News From The Associated Press URL:customwire.ap.org/dynamic/fron	10. (0.404) Empty title field URL:iparentingstore.com	10. (0.631) Le Web des icirc;les URL:www.chez.com/zanozile

HubAvg	Max	AT-med
1. (1.000) Le Web des icirc;les URL:www.chez.com/zanozile	1. (1.000) **EPICURIOUS: WORLD'S** **URL:www.epicurious.com**	1. (1.000) **EPICURIOUS: WORLD'S** **URL:www.epicurious.com**
2. (0.991) Please stand by.. URL:www.sofcom.com.au	2. (0.777) *Food Network* *URL:www.foodtv.com*	2. (0.820) *Food Network* *URL:www.foodtv.com*
3. (0.968) Sign in - Yahoo! Groups URL:groups.yahoo.com/group/mauriti	3. (0.709) **All Recipes — Recipes** **URL:www.allrecipes.com**	3. (0.796) **All Recipes — Recipes** **URL:www.allrecipes.com**
4. (0.005) Microsoft bCentral - FastCounte URL:fastcounter.bcentral.com/fc-jo	4. (0.623) **RecipeSource: Your Source for** **URL:www.recipesource.com**	4. (0.674) **RecipeSource: Your Source for** **URL:www.recipesource.com**
5. (0.004) **Recipes are Cooking at NetCoo** **URL:www.netcooks.com**	5. (0.490) **Top Secret Recipes on the Web** **URL:www.topsecretrecipes.com**	5. (0.550) **Top Secret Recipes on the Web** **URL:www.topsecretrecipes.com**
6. (0.004) *Mauritian cuisine, cooking and* *URL:ile-maurice.tripod.com*	6. (0.480) *Find Lost Recipes atnbsp; Reci* *URL:www.recipelink.com*	6. (0.543) *Find Lost Recipes atnbsp; Reci* *URL:www.recipelink.com*
7. (0.004) Mauritius Australia Connection URL:www.cjp.net	7. (0.440) *www.BettyCrocker.com* *URL:www.bettycrocker.com*	7. (0.495) *www.BettyCrocker.com* *URL:www.bettycrocker.com*
8. (0.003) Mauritius Australia Connection URL:www.users.bigpond.com/clancy/t	8. (0.431) **VegWeb - Vegan/Vegetarian** **URL:www.vegweb.com**	8. (0.482) **FATFREE: The Low Fat** **URL:www.fatfree.com**
9. (0.003) SleepAngel.com - Are you snorin URL:wcpsecure.com/app/aftrack.asp?	9. (0.430) **FATFREE: The Low Fat** **URL:www.fatfree.com**	9. (0.472) **VegWeb - Vegan/Vegetarian** **URL:www.vegweb.com**
10. (0.002) *Chef Jobs Foodservice Culinary* *URL:chef2chef.net*	10. (0.421) What You Need to Know About84 URL:www.about.com	10. (0.461) What You Need to Know About84 URL:www.about.com

AT-avg	Norm	BFS
1. (1.000) **EPICURIOUS: WORLD'S** **URL:www.epicurious.com**	1. (1.000) HonoluluAdvertiser.com gt; Haw URL:www.hawaiisclassifieds.com	1. (1.000) **EPICURIOUS: WORLD'S** **URL:www.epicurious.com**
2. (0.861) **All Recipes — Recipes** **URL:www.allrecipes.com**	2. (0.999) Gannett Company, Inc. URL:www.gannett.com	2. (1.000) Cutting-edge natural health tre URL:www.youngagain.com
3. (0.847) *Food Network* *URL:www.foodtv.com*	3. (0.996) AP MoneyWire URL:apmoneywire.mm.ap.org	3. (0.997) *Food Network* *URL:www.foodtv.com*
4. (0.715) **RecipeSource: Your Source for** **URL:www.recipesource.com**	4. (0.982) e.thePeople : Honolulu Advertis URL:www.e-thepeople.com/affiliates	4. (0.985) What You Need to Know About84 URL:www.about.com
5. (0.606) **Top Secret Recipes on the Web** **URL:www.topsecretrecipes.com**	5. (0.980) News From The Associated Press URL:customwire.ap.org/dynamic/fron	5. (0.981) **All Recipes — Recipes** **URL:www.allrecipes.com**
6. (0.599) *Find Lost Recipes atnbsp; Reci* *URL:www.recipelink.com*	6. (0.976) Honolulu Traffic Cameras, City URL:www.co.honolulu.hi.us/cameras/	6. (0.977) **RecipeSource: Your Source for** **URL:www.recipesource.com**
7. (0.542) *www.BettyCrocker.com* *URL:www.bettycrocker.com*	7. (0.976) News From The Associated Press URL:customwire.ap.org/dynamic/fron	7. (0.962) *Mauritian cuisine, cooking and* *URL:ile-maurice.tripod.com*
8. (0.502) **FATFREE: The Low Fat** **URL:www.fatfree.com**	8. (0.976) News From The Associated Press URL:customwire.ap.org/dynamic/fron	8. (0.956) **Top Secret Recipes on the Web** **URL:www.topsecretrecipes.com**
9. (0.477) **VegWeb - Vegan/Vegetarian** **URL:www.vegweb.com**	9. (0.976) News From The Associated Press URL:customwire.ap.org/dynamic/fron	9. (0.953) **VegWeb - Vegan/Vegetarian** **URL:www.vegweb.com**
10. (0.468) **Meals For You - Thousands** **URL:www.mealsforyou.com**	10. (0.976) News From The Associated Press URL:customwire.ap.org/dynamic/fron	10. (0.953) *Find Lost Recipes atnbsp; Reci* *URL:www.recipelink.com*

Table C.28: Query “recipes”

Hits	PageRank	InDegree
1. (1.000) Site Meter - Counter and Statis URL:www.sitemeter.com/stats.asp?si	1. (1.000) Fan Forum: Entertainment 4 Fans URL:www.fanforum.com	1. (1.000) *Roswell, NM* *URL:www.roswellnm.org*
2. (0.998) Dreambook - Camarila's Image Ga URL:books.dreambook.com/camarila/a	2. (0.244) Weather Underground: Roswell, N URL:www.wunderground.com/US/NM/Ros	2. (0.822) *Welcome to Roswell Rods.com* *URL:www.roswellrods.com*
3. (0.998) Camarila's Image Galleries's Dr URL:books.dreambook.com/camarila/m	3. (0.208) San Antonio Express-News Archiv URL:archives.newsbank.com/saenews	3. (0.813) Fan Forum: Entertainment 4 Fans URL:www.fanforum.com
4. (0.998) Camarila's Image Galleries of F URL:books.dreambook.com/camarila/a	4. (0.207) Rackspace Managed Hosting - Ded URL:www.rackspace.com/?supbid=mysa	4. (0.738) *general5* *URL:www.roswellproof.homestead.com*
5. (0.530) Camarila's Fantasy Image Galler URL:members.fortunecity.com/camari	5. (0.179) Forums 4 Fans URL:www.forums4fans.com	5. (0.710) *————————Welcome to* *URL:roswell_land.tripod.com*
6. (0.530) Camarila's Sci-Fi Image Galleri URL:members.fortunecity.com/camari	6. (0.178) *Roswell, NM* *URL:www.roswellnm.org*	6. (0.673) San Antonio Express-News Archiv URL:archives.newsbank.com/saenews
7. (0.530) Camarila's Horror Image Galleri URL:members.fortunecity.com/camari	7. (0.173) *Crashdown.com* *URL:www.crashdown.com*	7. (0.673) Site Meter - Counter and Statis URL:www.sitemeter.com/stats.asp?si
8. (0.530) Camarila's Artists Gallery List URL:members.fortunecity.com/camari	8. (0.165) Fan Forum: Contact Fan Forum URL:www.e4fans.com/fanforum/contac	8. (0.664) Rackspace Managed Hosting - Ded URL:www.rackspace.com/?supbid=mysa
9. (0.530) MIDI PAGE URL:rivendell.fortunecity.com/redg	9. (0.165) Fan Forum: Advertising URL:www.e4fans.com/fanforum/advert	9. (0.664) Dreambook - Camarila's Image Ga URL:books.dreambook.com/camarila/a
10. (0.530) Camarila's X-Files Pages -9 sea URL:members.fortunecity.com/camari	10. (0.165) Fan Forum: Privacy Policy URL:www.e4fans.com/fanforum/privac	10. (0.664) Camarila's Image Galleries's Dr URL:books.dreambook.com/camarila/m

HubAvg	Max	AT-med
1. (1.000) Fan Forum: Entertainment 4 Fans URL:www.fanforum.com	1. (1.000) *Roswell, NM* *URL:www.roswellnm.org*	1. (1.000) *Roswell, NM* *URL:www.roswellnm.org*
2. (0.248) *Forums 4 Fans: Roswell (1)* *URL:www.forums4fans.com/ultimatebb*	2. (0.548) *Welcome to Roswell Rods.com* *URL:www.roswellrods.com*	2. (0.717) *Welcome to Roswell Rods.com* *URL:www.roswellrods.com*
3. (0.140) *Crashdown.com* *URL:www.crashdown.com*	3. (0.345) Welcome to Adobe GoLive 6 URL:www.roswell.org	3. (0.624) *general5* *URL:www.roswellproof.homestead.com*
4. (0.016) *Welcome to Roswell Rods.com* *URL:www.roswellrods.com*	4. (0.332) *general5* *URL:www.roswellproof.homestead.com*	4. (0.541) *————————Welcome to* *URL:roswell_land.tripod.com*
5. (0.012) William Sadler - Wild on the We URL:www.williamsadler.com	5. (0.285) *————————Welcome to* *URL:roswell_land.tripod.com*	5. (0.505) The Chaparral Rockhounds Gem URL:www.chaparralrockhounds.com
6. (0.009) Adobe Acrobat Reader - Download URL:www.adobe.com/products/acrobat	6. (0.251) **Roswell UFO Crash of July 194** **URL:www.roswellufocrash.com**	6. (0.467) *MP3.com: Lights Over Roswell* *URL:www.lightsoverroswell.com*
7. (0.007) *————————Welcome to* *URL:roswell_land.tripod.com*	7. (0.238) The Chaparral Rockhounds Gem URL:www.chaparralrockhounds.com	7. (0.461) **Roswell UFO Crash of July 194** **URL:www.roswellufocrash.com**
8. (0.007) *Roswell Movie [Campaign]* *URL:www.roswellmovie.net*	8. (0.233) *MP3.com: Lights Over Roswell* *URL:www.lightsoverroswell.com*	8. (0.429) Empty title field URL:www.roswellsearch.com
9. (0.006) *Roswell: Crashdown (Episodes)* *URL:www.crashdown.com/episodes*	9. (0.227) *Empty title field* *URL:www.mufon.com*	9. (0.397) Welcome to Adobe GoLive 6 URL:www.roswell.org
10. (0.006) *general5* *URL:www.roswellproof.homestead.com*	10. (0.195) Empty title field URL:www.roswellsearch.com	10. (0.374) Church Christ - ChurchChrist - URL:www.churchchrist.org

AT-avg	Norm	BFS
1. (1.000) Site Meter - Counter and Statis URL:www.sitemeter.com/stats.asp?si	1. (1.000) Site Meter - Counter and Statis URL:www.sitemeter.com/stats.asp?si	1. (1.000) *Roswell, NM* *URL:www.roswellnm.org*
2. (0.996) Dreambook - Camarila's Image Ga URL:books.dreambook.com/camarila/a	2. (0.994) Dreambook - Camarila's Image Ga URL:books.dreambook.com/camarila/a	2. (0.969) *general5* *URL:www.roswellproof.homestead.com*
3. (0.996) Camarila's Image Galleries's Dr URL:books.dreambook.com/camarila/m	3. (0.994) Camarila's Image Galleries's Dr URL:books.dreambook.com/camarila/m	3. (0.964) Church Christ - ChurchChrist - URL:www.churchchrist.org
4. (0.996) Camarila's Image Galleries of F URL:books.dreambook.com/camarila/a	4. (0.994) Camarila's Image Galleries of F URL:books.dreambook.com/camarila/a	4. (0.958) Geography Home Page URL:geography.miningco.com
5. (0.617) Ampira Hosting - web hosting, d URL:www.ampira.com	5. (0.616) Ampira Hosting - web hosting, d URL:www.ampira.com	5. (0.949) Paris PC Consult - MapInfo GIS URL:www.paris-pc-gis.com
6. (0.604) Edward Gorey, amphigorey URL:www.geocities.com/SoHo/Canvas/	6. (0.595) Edward Gorey, amphigorey URL:www.geocities.com/SoHo/Canvas/	6. (0.938) The Chaparral Rockhounds Gem URL:www.chaparralrockhounds.com
7. (0.589) Camarila's Angels and Fairies P URL:www.geocities.com/gabriella66/	7. (0.582) Camarila's Angels and Fairies P URL:www.geocities.com/gabriella66/	7. (0.936) **Roswell UFO Crash of July 194** **URL:www.roswellufocrash.com**
8. (0.463) Smallville Pages, Episode Guide URL:www.geocities.com/gabriella66/	8. (0.463) Smallville Pages, Episode Guide URL:www.geocities.com/gabriella66/	8. (0.926) *————————Welcome to* *URL:roswell_land.tripod.com*
9. (0.463) Camarila's Total Recall-2070 Ep URL:www.geocities.com/camarilasout	9. (0.463) Camarila's Total Recall-2070 Ep URL:www.geocities.com/camarilasout	9. (0.915) *Index of /* *URL:www.roswell-record.com*
10. (0.421) Highlander Pages-6 seasons epis URL:www.geocities.com/akasha7_7/hi	10. (0.421) Highlander Pages-6 seasons epis URL:www.geocities.com/akasha7_7/hi	10. (0.883) *MP3.com: Lights Over Roswell* *URL:www.lightsoverroswell.com*

Table C.29: Query "roswell"

Hits	PageRank	InDegree
1. (1.000) **AltaVista** **URL:www.altavista.com**	1. (1.000) **Google Groups** **URL:www.dejanews.com**	1. (1.000) **AltaVista** **URL:www.altavista.com**
2. (0.981) Ego Surf - EgoSurf - egosurf.co URL:www.egosurf.com	2. (0.975) *Lycos Zone* *URL:www.terralycos.com*	2. (0.885) **Yahoo!** **URL:www.yahoo.com**
3. (0.979) *Yahoo! Danmark* *URL:www.yahoo.dk*	3. (0.958) **Google** **URL:www.google.com**	3. (0.885) **Google** **URL:www.google.com**
4. (0.973) **AltaVista Text-Only Search** **URL:ragingsearch.altavista.com**	4. (0.761) **Yahoo!** **URL:www.yahoo.com**	4. (0.727) **Lycos Home Page** **URL:www.lycos.com**
5. (0.972) **Euroseek** **URL:euroseek.net**	5. (0.756) *bruceclay.com - Web Site Promot* *URL:www.bruceclay.com*	5. (0.710) **Homepage HotBot Web Search** **URL:www.hotbot.com**
6. (0.972) *Your Search Engine Internet dir* *URL:www.searchpalm.com*	6. (0.743) **AltaVista** **URL:www.altavista.com**	6. (0.700) **Dogpile. Unleash the power of** **URL:www.dogpile.com**
7. (0.971) **About Web Search - Guide to** **URL:websearch.about.com**	7. (0.612) **iSleuth Meta Crawler Freebies** **URL:myfreebies.com/?a=cd42313**	7. (0.672) **My Excite** **URL:www.excite.com**
8. (0.970) **Abacho - THE POWERFUL** **URL:www.abacho.co.uk**	8. (0.588) CNET.com URL:www.cnet.com/frontdoor/0-1.htm	8. (0.580) **WebCrawler Index** **URL:www.webcrawler.com**
9. (0.970) careerhighway.com URL:www.careerhighway.com	9. (0.551) **My Excite** **URL:www.excite.com**	9. (0.480) **Northern Light** **URL:www.northernlight.com**
10. (0.970) **Ananzi South Africa - Search** **URL:www.ananzi.co.za**	10. (0.535) **Lycos Home Page** **URL:www.lycos.com**	10. (0.470) **AlltheWeb.com** **URL:www.alltheweb.com**

HubAvg	Max	AT-med
1. (1.000) **AltaVista** **URL:www.altavista.com**	1. (1.000) **AltaVista** **URL:www.altavista.com**	1. (1.000) **AltaVista** **URL:www.altavista.com**
2. (0.980) **Yahoo!** **URL:www.yahoo.com**	2. (0.851) **Yahoo!** **URL:www.yahoo.com**	2. (0.932) **Yahoo!** **URL:www.yahoo.com**
3. (0.961) **Google** **URL:www.google.com**	3. (0.844) **Google** **URL:www.google.com**	3. (0.866) **Google** **URL:www.google.com**
4. (0.789) **Lycos Home Page** **URL:www.lycos.com**	4. (0.699) **Lycos Home Page** **URL:www.lycos.com**	4. (0.800) **Lycos Home Page** **URL:www.lycos.com**
5. (0.745) **My Excite** **URL:www.excite.com**	5. (0.682) **Homepage HotBot Web Search** **URL:www.hotbot.com**	5. (0.784) **Homepage HotBot Web Search** **URL:www.hotbot.com**
6. (0.742) **Homepage HotBot Web Search** **URL:www.hotbot.com**	6. (0.653) **Dogpile. Unleash the power of** **URL:www.dogpile.com**	6. (0.738) **My Excite** **URL:www.excite.com**
7. (0.643) **Dogpile. Unleash the power of** **URL:www.dogpile.com**	7. (0.642) **My Excite** **URL:www.excite.com**	7. (0.728) **Dogpile. Unleash the power of** **URL:www.dogpile.com**
8. (0.582) **WebCrawler Index** **URL:www.webcrawler.com**	8. (0.550) **WebCrawler Index** **URL:www.webcrawler.com**	8. (0.630) **WebCrawler Index** **URL:www.webcrawler.com**
9. (0.453) **Northern Light** **URL:www.northernlight.com**	9. (0.458) **Northern Light** **URL:www.northernlight.com**	9. (0.524) **Northern Light** **URL:www.northernlight.com**
10. (0.415) **AlltheWeb.com** **URL:www.alltheweb.com**	10. (0.439) **AlltheWeb.com** **URL:www.alltheweb.com**	10. (0.483) **AlltheWeb.com** **URL:www.alltheweb.com**

AT-avg	Norm	BFS
1. (1.000) **AltaVista** **URL:www.altavista.com**	1. (1.000) **AltaVista** **URL:www.altavista.com**	1. (1.000) **AltaVista** **URL:www.altavista.com**
2. (0.887) **Yahoo!** **URL:www.yahoo.com**	2. (0.713) Ego Surf - EgoSurf - egosurf.co URL:www.egosurf.com	2. (0.981) **Google** **URL:www.google.com**
3. (0.827) **Google** **URL:www.google.com**	3. (0.670) *Yahoo! Danmark* *URL:www.yahoo.dk*	3. (0.931) **Lycos Home Page** **URL:www.lycos.com**
4. (0.801) **Lycos Home Page** **URL:www.lycos.com**	4. (0.666) **About Web Search - Guide to** **URL:websearch.about.com**	4. (0.918) **Yahoo!** **URL:www.yahoo.com**
5. (0.797) **Homepage HotBot Web Search** **URL:www.hotbot.com**	5. (0.638) **AltaVista Text-Only Search** **URL:ragingsearch.altavista.com**	5. (0.865) **Homepage HotBot Web Search** **URL:www.hotbot.com**
6. (0.763) **Dogpile. Unleash the power of** **URL:www.dogpile.com**	6. (0.637) *Empty title field* *URL:www.portalhub.com*	6. (0.859) **Dogpile. Unleash the power of** **URL:www.dogpile.com**
7. (0.752) **My Excite** **URL:www.excite.com**	7. (0.629) **Euroseek** **URL:euroseek.net**	7. (0.857) **My Excite** **URL:www.excite.com**
8. (0.656) **WebCrawler Index** **URL:www.webcrawler.com**	8. (0.628) *Your Search Engine Internet dir* *URL:www.searchpalm.com*	8. (0.852) Ego Surf - EgoSurf - egosurf.co URL:www.egosurf.com
9. (0.573) **Northern Light** **URL:www.northernlight.com**	9. (0.621) careerhighway.com URL:www.careerhighway.com	9. (0.842) **Search Engine Watch: Tips** **URL:searchenginewatch.com**
10. (0.522) **Search.com** **URL:www.search.com**	10. (0.619) CIA - The World Factbook 2002 URL:www.odci.gov/cia/publications/	10. (0.841) **Northern Light** **URL:www.northernlight.com**

Table C.30: Query "search engines"

Hits	PageRank	InDegree
1. (1.000) *The Oregon Shakespeare Festival* *URL:www.orshakes.org*	1. (1.000) Forbes.com Best of the Web URL:www.forbes.com/bow	1. (1.000) *Mr. William Shakespeare and the* *URL:daphne.palomar.edu/shakespeare*
2. (0.895) *Shakespeare Company* *URL:www.shakespeare-company.org*	2. (0.983) *Alabama Shakespeare Festival* *URL:www.asf.net*	2. (0.807) **The Complete Works of** **URL:the-tech.mit.edu/Shakespeare/w**
3. (0.870) *Idaho Shakespeare Festival* *URL:www.idahoshakespeare.org*	3. (0.742) Yahoo! URL:www.yahoo.com	3. (0.765) **shakespeare.com home** **URL:www.shakespeare.com**
4. (0.832) *Welcome to the Utah Shakespeare* *URL:www.bard.org*	4. (0.738) The University of Birmingham URL:www.bham.ac.uk	4. (0.735) **Shakespeare's Globe Theatre** **URL:www.rdg.ac.uk/globe**
5. (0.811) *The Shakespeare Theatre* *URL:www.shakespearedc.org*	5. (0.625) *Shakespeare's Globe Theatre, Ba* *URL:www.shakespeares-globe.org*	5. (0.597) **The Complete Works of** **URL:the-tech.mit.edu/Shakespeare**
6. (0.800) *Alabama Shakespeare Festival* *URL:www.asf.net*	6. (0.618) *Shakespeare Fishing Tackle* *URL:www.shakespeare-fishing.com*	6. (0.542) *RSC - Royal Shakespeare Company* *URL:www.rsc.org.uk*
7. (0.784) *Mr. William Shakespeare and the* *URL:daphne.palomar.edu/shakespeare*	7. (0.584) *Shakespeare Composites and Elec* *URL:www.shakespeare-ce.com*	7. (0.513) *Shakespeare Oxford Society Home* *URL:www.shakespeare-oxford.com*
8. (0.777) *Shakespeare's Globe Theatre, Ba* *URL:www.shakespeares-globe.org*	8. (0.569) **The Complete Works of** **URL:the-tech.mit.edu/Shakespeare/w**	8. (0.508) *Shakespeare's Globe Theatre, Ba* *URL:www.shakespeares-globe.org*
9. (0.773) *Welcome to Georgia Shakespeare* *URL:www.gashakespeare.org*	9. (0.546) **Shakespeare's Plays and Sonne** **URL:www.allshakespeare.com**	9. (0.492) *Shakespeare Navigators* *URL:www.clicknotes.com*
10. (0.771) *Kentucky Shakespeare Festival W* *URL:www.kyshakes.org*	10. (0.502) *Mr. William Shakespeare and the* *URL:daphne.palomar.edu/shakespeare*	10. (0.487) **Shakespeare: Internet Editio** **URL:web.uvic.ca/shakespeare**

HubAvg	Max	AT-med
1. (1.000) *Mr. William Shakespeare and the* *URL:daphne.palomar.edu/shakespeare*	1. (1.000) *Mr. William Shakespeare and the* *URL:daphne.palomar.edu/shakespeare*	1. (1.000) *Mr. William Shakespeare and the* *URL:daphne.palomar.edu/shakespeare*
2. (0.900) **The Complete Works of** **URL:the-tech.mit.edu/Shakespeare/w**	2. (0.578) **The Complete Works of** **URL:the-tech.mit.edu/Shakespeare/w**	2. (0.662) **The Complete Works of** **URL:the-tech.mit.edu/Shakespeare/w**
3. (0.693) **shakespeare.com home** **URL:www.shakespeare.com**	3. (0.507) **Shakespeare's Globe Theatre** **URL:www.rdg.ac.uk/globe**	3. (0.617) **shakespeare.com home** **URL:www.shakespeare.com**
4. (0.620) **Shakespeare's Globe Theatre** **URL:www.rdg.ac.uk/globe**	4. (0.503) **shakespeare.com home** **URL:www.shakespeare.com**	4. (0.600) **Shakespeare's Globe Theatre** **URL:www.rdg.ac.uk/globe**
5. (0.422) **The Complete Works of** **URL:the-tech.mit.edu/Shakespeare**	5. (0.371) **The Complete Works of** **URL:the-tech.mit.edu/Shakespeare**	5. (0.417) **The Complete Works of** **URL:the-tech.mit.edu/Shakespeare**
6. (0.327) *RSC - Royal Shakespeare Company* *URL:www.rsc.org.uk*	6. (0.340) **Shakespeare: Internet Edition** **URL:web.uvic.ca/shakespeare**	6. (0.397) **Shakespeare: Internet Edition** **URL:web.uvic.ca/shakespeare**
7. (0.324) *Shakespeare Oxford Society Home* *URL:www.shakespeare-oxford.com*	7. (0.303) *Shakespeare Oxford Society Home* *URL:www.shakespeare-oxford.com*	7. (0.352) *Shakespeare Oxford Society Home* *URL:www.shakespeare-oxford.com*
8. (0.322) *Shakespeare Magazine* *URL:www.shakespearemag.com*	8. (0.296) *Shakespeare Magazine* *URL:www.shakespearemag.com*	8. (0.337) *Shakespeare Magazine* *URL:www.shakespearemag.com*
9. (0.306) **Shakespeare: Internet Edition** **URL:web.uvic.ca/shakespeare**	9. (0.248) **Shakespeare Resource Center** **URL:www.bardweb.net**	9. (0.286) **The Shakespeare Birthplace Tr** **URL:www.shakespeare.org.uk**
10. (0.302) *Shakespeare's Globe Theatre, Ba* *URL:www.shakespeares-globe.org*	10. (0.246) **The Shakespeare Birthplace T** **URL:www.shakespeare.org.uk**	10. (0.285) **Shakespeare Resource Center** **URL:www.bardweb.net**

AT-avg	Norm	BFS
1. (1.000) *Mr. William Shakespeare and the* *URL:daphne.palomar.edu/shakespeare*	1. (1.000) *Mr. William Shakespeare and the* *URL:daphne.palomar.edu/shakespeare*	1. (1.000) *Mr. William Shakespeare and the* *URL:daphne.palomar.edu/shakespeare*
2. (0.661) **shakespeare.com home** **URL:www.shakespeare.com**	2. (0.641) **The Complete Works of** **URL:the-tech.mit.edu/Shakespeare/w**	2. (0.955) **The Complete Works of** **URL:the-tech.mit.edu/Shakespeare/w**
3. (0.655) **Shakespeare's Globe Theatre** **URL:www.rdg.ac.uk/globe**	3. (0.637) **shakespeare.com home** **URL:www.shakespeare.com**	3. (0.948) **shakespeare.com home** **URL:www.shakespeare.com**
4. (0.650) **The Complete Works of** **URL:the-tech.mit.edu/Shakespeare/w**	4. (0.616) **Shakespeare's Globe Theatre** **URL:www.rdg.ac.uk/globe**	4. (0.924) *Shakespeare Oxford Society Home* *URL:www.shakespeare-oxford.com*
5. (0.457) **Shakespeare: Internet Edition** **URL:web.uvic.ca/shakespeare**	5. (0.423) **The Complete Works of** **URL:the-tech.mit.edu/Shakespeare**	5. (0.921) **Shakespeare's Globe Theatre** **URL:www.rdg.ac.uk/globe**
6. (0.444) **The Complete Works of** **URL:the-tech.mit.edu/Shakespeare**	6. (0.423) **Shakespeare: Internet Edition** **URL:web.uvic.ca/shakespeare**	6. (0.917) **Shakespeare Resource Center** **URL:www.bardweb.net**
7. (0.400) *Shakespeare Oxford Society Home* *URL:www.shakespeare-oxford.com*	7. (0.381) *Shakespeare Oxford Society Home* *URL:www.shakespeare-oxford.com*	7. (0.893) *Shakespeare's Globe Theatre, Ba* *URL:www.shakespeares-globe.org*
8. (0.379) *Shakespeare Magazine* *URL:www.shakespearemag.com*	8. (0.360) *Shakespeare Magazine* *URL:www.shakespearemag.com*	8. (0.890) *Shakespeare Navigators* *URL:www.clicknotes.com*
9. (0.324) **Shakespeare Resource Center** **URL:www.bardweb.net**	9. (0.318) *Shakespeare's Globe Theatre, Ba* *URL:www.shakespeares-globe.org*	9. (0.877) **Shakespeare: Internet Edition** **URL:web.uvic.ca/shakespeare**
10. (0.321) **The Shakespeare Birthplace T** **URL:www.shakespeare.org.uk**	10. (0.315) **The Shakespeare Birthplace T** **URL:www.shakespeare.org.uk**	10. (0.866) **The Shakespeare Birthplace T** **URL:www.shakespeare.org.uk**

Table C.31: Query "shakespeare"

Hits	PageRank	InDegree
1. (1.000) **ITTF** **URL:www.ittf.com**	1. (1.000) *Tibhar-HomePage* *URL:www.tibhar.de*	1. (1.000) **ITTF** **URL:www.ittf.com**
2. (0.600) **Empty title field** **URL:www.usatt.org**	2. (0.969) **ITTF** **URL:www.ittf.com**	2. (0.608) **Empty title field** **URL:www.usatt.org**
3. (0.599) **ETTU - European Table Tennis** **URL:www.ettu.org**	3. (0.899) *Table Tennis/Ping-Pong Classifi* *URL:adlistings.tabletennis.about.c*	3. (0.446) **ETTU - European Table Tennis** **URL:www.ettu.org**
4. (0.480) **ETTA: English Table Tennis As** **URL:www.etta.co.uk**	4. (0.741) Adobe Acrobat Reader - Download URL:www.adobe.com/products/acrobat	4. (0.412) *Table Tennis and Ping Pong Acce* *URL:www.butterflyonline.com*
5. (0.268) *Denis' Table Tennis / Ping-Pong* *URL:www.tabletennis.gr*	5. (0.737) URL:www.people.com.cn	5. (0.392) **ETTA: English Table Tennis As** **URL:www.etta.co.uk**
6. (0.265) What You Need to Know About84 URL:www.about.com	6. (0.662) *ITTF HANDBOOK* *URL:www.ittf.com/Regulations/Regul*	6. (0.348) *Tibhar-HomePage* *URL:www.tibhar.de*
7. (0.260) *Welcome to Sweden Table Tennis* *URL:www.tabletennis.se*	7. (0.637) *Table Tennis and Ping Pong Acce* *URL:www.butterflyonline.com*	7. (0.265) *BS Table Tennis* *URL:bstt.cjb.net*
8. (0.257) *BS Table Tennis* *URL:bstt.cjb.net*	8. (0.633) **Empty title field** **URL:www.usatt.org**	8. (0.255) *Denis' Table Tennis / Ping-Pong* *URL:www.tabletennis.gr*
9. (0.252) *Table Tennis / Ping-Pong* *URL:www.megaspin.net*	9. (0.626) TheCounter.com: The Full-Featur URL:www.TheCounter.com	9. (0.255) **TABLE TENNIS CANADA** **URL:www.ctta.ca**
10. (0.239) **TABLE TENNIS CANADA** **URL:www.ctta.ca**	10. (0.554) WebSTAT URL:hits.webstat.com	10. (0.255) *Welcome to Sweden Table Tennis* *URL:www.tabletennis.se*

HubAvg	Max	AT-med
1. (1.000) **ITTF** **URL:www.ittf.com**	1. (1.000) **ITTF** **URL:www.ittf.com**	1. (1.000) **ITTF** **URL:www.ittf.com**
2. (0.392) **Empty title field** **URL:www.usatt.org**	2. (0.456) **Empty title field** **URL:www.usatt.org**	2. (0.500) **Empty title field** **URL:www.usatt.org**
3. (0.280) **ETTU - European Table Tennis** **URL:www.ettu.org**	3. (0.409) **ETTU - European Table Tennis** **URL:www.ettu.org**	3. (0.460) **ETTU - European Table Tennis** **URL:www.ettu.org**
4. (0.184) **ETTA: English Table Tennis As** **URL:www.etta.co.uk**	4. (0.302) **ETTA: English Table Tennis As** **URL:www.etta.co.uk**	4. (0.327) **ETTA: English Table Tennis As** **URL:www.etta.co.uk**
5. (0.157) *Tibhar-HomePage* *URL:www.tibhar.de*	5. (0.162) *Denis' Table Tennis / Ping-Pong* *URL:www.tabletennis.gr*	5. (0.179) *Denis' Table Tennis / Ping-Pong* *URL:www.tabletennis.gr*
6. (0.142) *Table Tennis and Ping Pong Acce* *URL:www.butterflyonline.com*	6. (0.152) What You Need to Know About84 URL:www.about.com	6. (0.172) What You Need to Know About84 URL:www.about.com
7. (0.096) **TABLE TENNIS CANADA** **URL:www.ctta.ca**	7. (0.148) *Welcome to Sweden Table Tennis* *URL:www.tabletennis.se*	7. (0.166) *Welcome to Sweden Table Tennis* *URL:www.tabletennis.se*
8. (0.093) *Welcome to Sweden Table Tennis* *URL:www.tabletennis.se*	8. (0.147) *Table Tennis / Ping-Pong* *URL:www.megaspin.net*	8. (0.163) *Table Tennis / Ping-Pong* *URL:www.megaspin.net*
9. (0.092) *Denis' Table Tennis / Ping-Pong* *URL:www.tabletennis.gr*	9. (0.136) *Table Tennis and Ping Pong Acce* *URL:www.butterflyonline.com*	9. (0.148) *Table Tennis and Ping Pong Acce* *URL:www.butterflyonline.com*
10. (0.080) What You Need to Know About84 URL:www.about.com	10. (0.134) **TABLE TENNIS CANADA** **URL:www.ctta.ca**	10. (0.148) **TABLE TENNIS CANADA** **URL:www.ctta.ca**

AT-avg	Norm	BFS
1. (1.000) **ITTF** **URL:www.ittf.com**	1. (1.000) **ITTF** **URL:www.ittf.com**	1. (1.000) **ITTF** **URL:www.ittf.com**
2. (0.541) **Empty title field** **URL:www.usatt.org**	2. (0.489) **Empty title field** **URL:www.usatt.org**	2. (0.860) **Empty title field** **URL:www.usatt.org**
3. (0.512) **ETTU - European Table Tennis** **URL:www.ettu.org**	3. (0.443) **ETTU - European Table Tennis** **URL:www.ettu.org**	3. (0.832) **ETTU - European Table Tennis** **URL:www.ettu.org**
4. (0.374) **ETTA: English Table Tennis As** **URL:www.etta.co.uk**	4. (0.324) **ETTA: English Table Tennis As** **URL:www.etta.co.uk**	4. (0.781) **ETTA: English Table Tennis As** **URL:www.etta.co.uk**
5. (0.204) *Denis' Table Tennis / Ping-Pong* *URL:www.tabletennis.gr*	5. (0.178) *Denis' Table Tennis / Ping-Pong* *URL:www.tabletennis.gr*	5. (0.773) What You Need to Know About84 URL:www.about.com
6. (0.194) What You Need to Know About84 URL:www.about.com	6. (0.168) What You Need to Know About84 URL:www.about.com	6. (0.767) *Table Tennis and Ping Pong Acce* *URL:www.butterflyonline.com*
7. (0.188) *Welcome to Sweden Table Tennis* *URL:www.tabletennis.se*	7. (0.164) *Welcome to Sweden Table Tennis* *URL:www.tabletennis.se*	7. (0.761) *BS Table Tennis* *URL:bstt.cjb.net*
8. (0.187) *Table Tennis / Ping-Pong* *URL:www.megaspin.net*	8. (0.162) *Table Tennis / Ping-Pong* *URL:www.megaspin.net*	8. (0.736) *Leamington amp; District Table* *URL:leamingtontt.tripod.com*
9. (0.175) **TABLE TENNIS CANADA** **URL:www.ctta.ca**	9. (0.153) *Table Tennis and Ping Pong Acce* *URL:www.butterflyonline.com*	9. (0.731) *Denis' Table Tennis / Ping-Pong* *URL:www.tabletennis.gr*
10. (0.164) *Table Tennis and Ping Pong Acce* *URL:www.butterflyonline.com*	10. (0.151) **TABLE TENNIS CANADA** **URL:www.ctta.ca**	10. (0.709) *Welcome to Sweden Table Tennis* *URL:www.tabletennis.se*

Table C.32: Query “table tennis”

Hits	PageRank	InDegree
1. (1.000) **NOAA - National Weather** **URL:www.nws.noaa.gov**	1. (1.000) **NOAA Home Page** **URL:www.noaa.gov**	1. (1.000) **NOAA Home Page** **URL:www.noaa.gov**
2. (0.986) **NOAA Home Page** **URL:www.noaa.gov**	2. (0.611) **NOAA - National Weather** **URL:www.nws.noaa.gov**	2. (0.973) **NOAA - National Weather** **URL:www.nws.noaa.gov**
3. (0.473) NOAA - National Weather Service URL:www.nws.noaa.gov/disclaimer.ht	3. (0.490) New York Times Company URL:www.nytco.com	3. (0.547) **weather.com** **URL:www.weather.com**
4. (0.425) *National Hurricane Center / Tro* *URL:www.nhc.noaa.gov*	4. (0.443) The New York Times: Travel URL:www.nytimes.com/pages/travel	4. (0.469) *National Hurricane Center / Tro* *URL:www.nhc.noaa.gov*
5. (0.403) *Climate Prediction Center* *URL:www.cpc.ncep.noaa.gov*	5. (0.361) Department of Commerce Home URL:www.doc.gov	5. (0.399) NOAA - National Weather Service URL:www.nws.noaa.gov/disclaimer.ht
6. (0.363) **weather.com** **URL:www.weather.com**	6. (0.325) FirstGov – Your First Click to URL:www.firstgov.gov	6. (0.375) **Intellicast - Weather For Act** **URL:www.intellicast.com**
7. (0.358) NOAA Home Page - Privacy amp; URL:www.noaa.gov/privacy.html	7. (0.233) *NESDIS Home Page* *URL:www.nesdis.noaa.gov*	7. (0.343) **Weather Underground** **URL:www.wunderground.com**
8. (0.293) **Intellicast - Weather For Act** **URL:www.intellicast.com**	8. (0.213) *Climate Prediction Center* *URL:www.cpc.ncep.noaa.gov*	8. (0.333) *Climate Prediction Center* *URL:www.cpc.ncep.noaa.gov*
9. (0.259) **NWS page** **URL:www.wrh.noaa.gov/wrhq/nwspage.**	9. (0.207) NOAA Home Page - Privacy amp; URL:www.noaa.gov/privacy.html	9. (0.304) NOAA Home Page - Privacy amp; URL:www.noaa.gov/privacy.html
10. (0.243) **Weather Underground** **URL:www.wunderground.com**	10. (0.199) *GEOSTATIONARY SATELLITE* *URL:www.goes.noaa.gov*	10. (0.286) **UM Weather** **URL:cirrus.sprl.umich.edu/wxnet**

HubAvg	Max	AT-med
1. (1.000) **NOAA Home Page** **URL:www.noaa.gov**	1. (1.000) **NOAA Home Page** **URL:www.noaa.gov**	1. (1.000) **NOAA Home Page** **URL:www.noaa.gov**
2. (0.872) **NOAA - National Weather** **URL:www.nws.noaa.gov**	2. (0.958) **NOAA - National Weather** **URL:www.nws.noaa.gov**	2. (0.984) **NOAA - National Weather** **URL:www.nws.noaa.gov**
3. (0.460) NOAA - National Weather Service URL:www.nws.noaa.gov/disclaimer.ht	3. (0.394) NOAA - National Weather Service URL:www.nws.noaa.gov/disclaimer.ht	3. (0.483) NOAA - National Weather Service URL:www.nws.noaa.gov/disclaimer.ht
4. (0.297) NOAA Home Page - Privacy amp; URL:www.noaa.gov/privacy.html	4. (0.354) *National Hurricane Center / Tro* *URL:www.nhc.noaa.gov*	4. (0.345) NOAA Home Page - Privacy amp; URL:www.noaa.gov/privacy.html
5. (0.194) *Climate Prediction Center* *URL:www.cpc.ncep.noaa.gov*	5. (0.308) **weather.com** **URL:www.weather.com**	5. (0.337) *National Hurricane Center / Tro* *URL:www.nhc.noaa.gov*
6. (0.183) **NWS page** **URL:www.wrh.noaa.gov/wrhq/nwspage.**	6. (0.302) *Climate Prediction Center* *URL:www.cpc.ncep.noaa.gov*	6. (0.335) *Climate Prediction Center* *URL:www.cpc.ncep.noaa.gov*
7. (0.171) *National Hurricane Center / Tro* *URL:www.nhc.noaa.gov*	7. (0.301) NOAA Home Page - Privacy amp; URL:www.noaa.gov/privacy.html	7. (0.266) **weather.com** **URL:www.weather.com**
8. (0.152) Department of Commerce Home URL:www.doc.gov	8. (0.228) **Intellicast - Weather For Act** **URL:www.intellicast.com**	8. (0.233) **NWS page** **URL:www.wrh.noaa.gov/wrhq/nwspage.**
9. (0.151) *NOAA - National Weather Service* *URL:www.nws.noaa.gov/pa*	9. (0.212) **NWS page** **URL:www.wrh.noaa.gov/wrhq/nwspage.**	9. (0.203) **Intellicast - Weather For Act** **URL:www.intellicast.com**
10. (0.151) NOAA - National Weather Service URL:www.nws.noaa.gov/notice.html	10. (0.191) **Weather Underground** **URL:www.wunderground.com**	10. (0.199) *NOAA - National Weather Service* *URL:www.nws.noaa.gov/pa*

AT-avg	Norm	BFS
1. (1.000) **NOAA Home Page** **URL:www.noaa.gov**	1. (1.000) **NOAA Home Page** **URL:www.noaa.gov**	1. (1.000) **NOAA - National Weather** **URL:www.nws.noaa.gov**
2. (0.991) **NOAA - National Weather** **URL:www.nws.noaa.gov**	2. (0.975) **NOAA - National Weather** **URL:www.nws.noaa.gov**	2. (0.983) **NOAA Home Page** **URL:www.noaa.gov**
3. (0.504) NOAA - National Weather Service URL:www.nws.noaa.gov/disclaimer.ht	3. (0.445) NOAA - National Weather Service URL:www.nws.noaa.gov/disclaimer.ht	3. (0.888) **weather.com** **URL:www.weather.com**
4. (0.371) NOAA Home Page - Privacy amp; URL:www.noaa.gov/privacy.html	4. (0.343) *National Hurricane Center / Tro* *URL:www.nhc.noaa.gov*	4. (0.855) *National Hurricane Center / Tro* *URL:www.nhc.noaa.gov*
5. (0.361) *Climate Prediction Center* *URL:www.cpc.ncep.noaa.gov*	5. (0.327) NOAA Home Page - Privacy amp; URL:www.noaa.gov/privacy.html	5. (0.820) **Weather Underground** **URL:www.wunderground.com**
6. (0.350) *National Hurricane Center / Tro* *URL:www.nhc.noaa.gov*	6. (0.324) *Climate Prediction Center* *URL:www.cpc.ncep.noaa.gov*	6. (0.813) **Intellicast - Weather For Act** **URL:www.intellicast.com**
7. (0.277) **weather.com** **URL:www.weather.com**	7. (0.282) **weather.com** **URL:www.weather.com**	7. (0.793) **UM Weather** **URL:cirrus.sprl.umich.edu/wxnet**
8. (0.252) **NWS page** **URL:www.wrh.noaa.gov/wrhq/nwspage.**	8. (0.225) **NWS page** **URL:www.wrh.noaa.gov/wrhq/nwspage.**	8. (0.752) Google URL:www.google.com
9. (0.218) *NOAA - National Weather Service* *URL:www.nws.noaa.gov/pa*	9. (0.214) **Intellicast - Weather For Act** **URL:www.intellicast.com**	9. (0.744) *Climate Prediction Center* *URL:www.cpc.ncep.noaa.gov*
10. (0.214) **Intellicast - Weather For Ac** **URL:www.intellicast.com**	10. (0.189) *NOAA - National Weather Service* *URL:www.nws.noaa.gov/pa*	10. (0.738) **CNN.com - Weather** **URL:www.cnn.com/WEATHER**

Table C.33: Query “weather”

Hits	PageRank	InDegree
1. (1.000) Yahoo! URL:www.yahoo.com	1. (1.000) Crain Communications, Inc. URL:www.crain.com	1. (1.000) **MCSCCpix** **URL:www.mcsccpix.homestead.com**
2. (0.976) Yahoo! Terms of Service URL:docs.yahoo.com/info/terms	2. (0.616) *AutoWeek - Premier Source for A* *URL:www.autoweek.com*	2. (0.949) *Awesome Dodge Chargers for sale* *URL:martinpacker.com*
3. (0.972) *Yahoo! Autos* *URL:autos.yahoo.com*	3. (0.523) TimeZone URL:www.timezone.com	3. (0.735) VINTAGE POSTCARDS URL:vintagepostcards1.tripod.com
4. (0.959) *Yahoo! - Autos* *URL:help.yahoo.com/help/autos*	4. (0.414) Online Casino - www.888.com - t URL:rd1.hitbox.com/rd?acct=WQ52083	4. (0.721) **Home,die-cast-models, vintage** **URL:www.vintagediecast.com**
5. (0.958) Yahoo! Autos Sell Your Car URL:classifieds.autos.yahoo.com/cl	5. (0.391) TheCounter.com: The Full-Featur URL:www.TheCounter.com	5. (0.706) **Classic Car Classifieds from** **URL:www.hemmings.com**
6. (0.958) Sign in - Yahoo! Companion URL:us.edit.companion.yahoo.com/co	6. (0.373) Motivational Posters and Inspir URL:www.motivatepost.com	6. (0.706) Yahoo! URL:www.yahoo.com
7. (0.944) Yahoo! Privacy URL:privacy.yahoo.com	7. (0.369) Crain Communications Inc. URL:www.elasesor.com.mx	7. (0.684) F1 is Web F1 - Formula One News URL:www.webf1.net
8. (0.941) Yahoo! Classifieds URL:classifieds.yahoo.com	8. (0.367) SEJOONG NAMO URL:www.namo.com	8. (0.618) **Classic Car - ClassicCar.com** **URL:www.classicar.com**
9. (0.937) Yahoo! Shopping URL:shopping.yahoo.com	9. (0.362) CarePackages.com: Care Packages URL:collegeclub.carepackages.com	9. (0.610) *AutoWeek - Premier Source for A* *URL:www.autoweek.com*
10. (0.936) Yahoo! Media Relations URL:docs.yahoo.com/info/copyright/	10. (0.330) Animal Posters and Prints URL:www.animalposterz.com	10. (0.588) Yahoo! Autos Sell Your Car URL:classifieds.autos.yahoo.com/cl

HubAvg	Max	AT-med
1. (1.000) Yahoo! URL:www.yahoo.com	1. (1.000) **MCSCCpix** **URL:www.mcsccpix.homestead.com**	1. (1.000) **MCSCCpix** **URL:www.mcsccpix.homestead.com**
2. (0.919) Yahoo! Terms of Service URL:docs.yahoo.com/info/terms	2. (0.833) *Awesome Dodge Chargers for sale* *URL:martinpacker.com*	2. (0.955) *Awesome Dodge Chargers for sale* *URL:martinpacker.com*
3. (0.891) Yahoo! Autos Sell Your Car URL:classifieds.autos.yahoo.com/cl	3. (0.577) VINTAGE POSTCARDS URL:vintagepostcards1.tripod.com	3. (0.732) **Home,die-cast-models, vintage** **URL:www.vintagediecast.com**
4. (0.891) Sign in - Yahoo! Companion URL:us.edit.companion.yahoo.com/co	4. (0.568) **Home,die-cast-models, vintage** **URL:www.vintagediecast.com**	4. (0.662) VINTAGE POSTCARDS URL:vintagepostcards1.tripod.com
5. (0.890) *Yahoo! Autos* *URL:autos.yahoo.com*	5. (0.513) F1 is Web F1 - Formula One News URL:www.webf1.net	5. (0.650) F1 is Web F1 - Formula One News URL:www.webf1.net
6. (0.808) *Yahoo! - Autos* *URL:help.yahoo.com/help/autos*	6. (0.391) **Classic Car Classifieds from** **URL:www.hemmings.com**	6. (0.492) **Classic Car Classifieds from** **URL:www.hemmings.com**
7. (0.804) Yahoo! Privacy URL:privacy.yahoo.com	7. (0.350) **infoclassic - auto d'epoca -** **URL:www.infoclassic.net**	7. (0.444) **infoclassic - auto d'epoca -** **URL:www.infoclassic.net**
8. (0.762) Yahoo! Media Relations URL:docs.yahoo.com/info/copyright/	8. (0.322) *Welcome to Barn Hill Minis* *URL:www.barnhillminisusa.com*	8. (0.414) _BuyDomains.com_‘ URL:www.vintageracing.net
9. (0.742) Yahoo! Classifieds URL:classifieds.yahoo.com	9. (0.320) _BuyDomains.com_‘ URL:www.vintageracing.net	9. (0.414) *Welcome to Barn Hill Minis* *URL:www.barnhillminisusa.com*
10. (0.722) Yahoo! Shopping URL:shopping.yahoo.com	10. (0.315) Pacific Coast Alfa Romeo Owners URL:alfaowners.cjb.net	10. (0.396) Owens Export Services, Inc. URL:www.militaryjeep.com

AT-avg	Norm	BFS
1. (1.000) Yahoo! URL:www.yahoo.com	1. (1.000) Yahoo! URL:www.yahoo.com	1. (1.000) **MCSCCpix** **URL:www.mcsccpix.homestead.com**
2. (0.916) Yahoo! Terms of Service URL:docs.yahoo.com/info/terms	2. (0.916) Yahoo! Autos Sell Your Car URL:classifieds.autos.yahoo.com/cl	2. (0.985) *Awesome Dodge Chargers for sale* *URL:martinpacker.com*
3. (0.910) *Yahoo! Autos* *URL:autos.yahoo.com*	3. (0.916) Sign in - Yahoo! Companion URL:us.edit.companion.yahoo.com/co	3. (0.977) **infoclassic - auto d'epoca -** **URL:www.infoclassic.net**
4. (0.910) Yahoo! Autos Sell Your Car URL:classifieds.autos.yahoo.com/cl	4. (0.905) *Yahoo! Autos* *URL:autos.yahoo.com*	4. (0.942) *Shared Top Border* *URL:www.calgaryvintageracing.com*
5. (0.910) Sign in - Yahoo! Companion URL:us.edit.companion.yahoo.com/co	5. (0.904) Yahoo! Terms of Service URL:docs.yahoo.com/info/terms	5. (0.915) *Welcome to Barn Hill Minis* *URL:www.barnhillminisusa.com*
6. (0.872) *Yahoo! - Autos* *URL:help.yahoo.com/help/autos*	6. (0.863) *Yahoo! - Autos* *URL:help.yahoo.com/help/autos*	6. (0.905) VINTAGE POSTCARDS URL:vintagepostcards1.tripod.com
7. (0.849) Yahoo! Privacy URL:privacy.yahoo.com	7. (0.849) Yahoo! Privacy URL:privacy.yahoo.com	7. (0.904) **Home,die-cast-models, vintage** **URL:www.vintagediecast.com**
8. (0.828) Yahoo! Media Relations URL:docs.yahoo.com/info/copyright/	8. (0.832) Yahoo! Media Relations URL:docs.yahoo.com/info/copyright/	8. (0.898) Pacific Coast Alfa Romeo Owners URL:alfaowners.cjb.net
9. (0.825) Yahoo! Classifieds URL:classifieds.yahoo.com	9. (0.830) Yahoo! Classifieds URL:classifieds.yahoo.com	9. (0.889) *World Wide Wheels Classifieds o* *URL:www.specialcar.com*
10. (0.815) Yahoo! Shopping URL:shopping.yahoo.com	10. (0.822) Yahoo! Shopping URL:shopping.yahoo.com	10. (0.881) _BuyDomains.com_‘ URL:www.vintageracing.net

Table C.34: Query “vintage cars”