Internet Mathematics Vol. 3, No. 4: 479-507

Stability and Similarity of Link Analysis Ranking Algorithms

Debora Donato, Stefano Leonardi, and Panayiotis Tsaparas

Abstract. Recently, there has been a surge of research activity in the area of *link analysis ranking*, where hyperlink structures are used to determine the relative *authority* of webpages. One of the seminal works in this area is that of Kleinberg [Kleinberg 98], who proposed the HITS algorithm. In this paper, we undertake a theoretical analysis of the properties of the HITS algorithm on a broad class of random graphs. Working within the framework of Borodin et al. [Borodin et al. 05], we prove that, under some assumptions, on this class (a) the HITS algorithm is stable with high probability and (b) the HITS algorithm is similar to the InDegree heuristic that assigns to each node weight proportional to the number of incoming links. We demonstrate that our results go through for the case that the expected in-degrees of the graph follow a power law distribution. We also study experimentally the similarity between HITS and InDegree, and we investigate the general conditions under which the two algorithms are similar.

1. Introduction

In the past years there has been increasing research interest in the analysis of the web graph for the purpose of improving the performance of search engines. The seminal works of Kleinberg [Kleinberg 98] and Brin and Page [Brin and Page 98] introduced the area of *link analysis ranking* (LAR), where hyperlink structures are used to rank the results of search queries. Their work was followed by a plethora of modifications, generalizations, and improvements [Bharat and Henzinger 98, Lempel and Moran 00, Rafiei and Mendelzon 00, Ng et al. 01b, Achlioptas et al. 01, Borodin et al. 01, Haveliwala 02, Tomlin 03, Tsaparas 04]. As a result, today there exists a wide range of link analysis ranking algorithms, many of which are variations of each other.

1542-7951/06 $0.50 per page

The wide usage of LAR algorithms raises naturally the question of defining a formal framework for assessing and comparing their properties. Borodin et al. introduced a theoretical framework for the analysis of LAR algorithms [Borodin et al. 01, Borodin et al. 05]. In their framework a LAR algorithm is defined as a function from a class of graphs of size n to an n-dimensional real vector. Every node in the graph is associated with a *weight* that captures the relative *authority* of the node. The nodes are ranked in decreasing order of their weights. Borodin et al. studied various properties of LAR algorithms such as stability, similarity, monotonicity, and locality [Borodin et al. 01]. In this work we focus on stability and similarity. *Stability* considers the effect of small changes in the graph to the output of an LAR algorithm. An LAR algorithm is stable if small changes in the graph result in small changes in the output of the algorithm. *Similarity* studies how close are the results of two algorithms on the same graph. Similar algorithms should produce similar rankings on the same graph.

Borodin et al. considered the question of stability and similarity over an unrestricted class of graphs [Borodin et al. 05]. They studied a variety of algorithms, and they proved that no pair of these algorithms is similar and that almost all algorithms are unstable. It appears that the class of all possible graphs is too broad to allow for positive results. This naturally raises the question whether it is possible to prove positive results if we restrict ourselves to a smaller class of graphs. Since the explosion of the World Wide Web, various stochastic models have been proposed for the web graph [Azar et al. 01, Barabasi and Albert 99, Kumar et al. 00a, Aiello et al. 01]. The model we consider, that was proposed by Azar et al. [Azar et al. 01], is the following: assume that every node i in the graph comes with two parameters a_i and h_i that take values in $[0, 1]$. For some node i, the value h_i can be thought of as the probability of node i to be a good *hub*, while the value a_i is the probability of the node i to be a good *authority*. We then generate an edge from i to j with probability proportional to $h_i a_j$. We will refer to this model as the *product model* and to the corresponding class of graphs as the class of *product graphs*. The product graph model generalizes the traditional random graph model of Erdös and Rèny [Erdős and Rèny 60] to include graphs where the *expected* degrees follow specific distributions. This is of particular interest since it is well known [Kumar et al. 00a, Broder et al. 00] that the in-degrees of the nodes in the web graph follow a power law distribution. This model has recently attracted considerable attention [Azar et al. 01, Chung and Lu 02, Mihail and Papadimitriou 02, Chung et al. 03], as a model for real-life networks such as the Internet and the World Wide Web.

1.1. Our Contribution

In this paper we study the behavior of the HITS algorithm, proposed by Kleinberg [Kleinberg 98], on the class of product graphs. The study of HITS on product graphs was initiated by Azar et al. [Azar et al. 01], who showed that

under some assumptions the HITS algorithm returns weights that are very close to the authority parameters. We formalize the findings of Azar et al. [Azar et al. 01] in the framework of Borodin et al. [Borodin et al. 05]. We extend the definitions of stability and similarity for classes of random graphs, and we demonstrate a connection between stability and similarity. We then prove that, with high probability, under some restrictive assumptions the HITS algorithm is stable on the class of product graphs and is similar to the InDegree heuristic that ranks pages according to their in-degree. This similarity result is the main contribution of the paper. The implication of the result is that on product graphs, with high probability, the HITS algorithm reduces to simple in-degree count. We show that our assumptions are general enough to capture graphs where the expected degrees follow a power law distribution, and we provide conditions for the stability and similarity of LAR algorithms on the Erdös-Rèny model. We also analyze the correlation between InDegree and HITS on a large sample of the web graph. The experimental analysis reveals that similarity between HITS and InDegree can also be observed on the World Wide Web. We conclude with a discussion on the conditions that guarantee similarity of HITS and InDegree for the class of all possible graphs.

Our work focuses on the theoretical understanding of LAR algorithms, and the relationship between HITS and InDegree. For this analysis we focus on the product graph model, which has received considerable attention in the network literature. Although the product model cannot capture the full complexity of the web graph, it is still an interesting model to study since it may be proven to be an adequate model for some specialized subset of the World Wide Web, or for some other real-life network. The motivation of our work comes from web search: however, LAR algorithms can be applied to any setting where given a network we want to infer a measure of authority for the nodes in the network using its link structure. Applications include diverse fields such as databases [Bhalotia et al. 02], immunology [Dezso and Barabasi 02], and social network analysis [Newman 01].

2. Related Work

2.1. Link Analysis Ranking Algorithms

Let P be a collection of n webpages that need to be ranked. This collection may be the whole web, or a query dependent subset of the web. We construct the underlying *hyperlink graph* $G = (P, E)$ by creating a node for each webpage in the collection, and a directed edge for each hyperlink between two pages. The input to a LAR algorithm is the $n \times n$ adjacency matrix W of the graph G. The output of the algorithm is an n-dimensional *authority weight vector* $\boldsymbol{w}$, where w_i, the ith coordinate of $\boldsymbol{w}$, is the authority weight of node i.

We now describe the two LAR algorithms we consider in this paper: the InDegree algorithm and the HITS algorithm. The InDegree algorithm is the simple heuristic that assigns to each node weight equal to the number of incoming links in the graph G. The HITS algorithm was proposed by Kleinberg in the seminal paper [Kleinberg 98] that introduced the hubs and authorities paradigm. In the HITS framework, every page can be thought of as having a *hub* and an *authority* identity. There is a mutually reinforcing relationship between the two. A good hub is a page that points to many good authorities, while a good authority is a page that is pointed to by many 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*, and he proposed the following iterative algorithm, called HITS, for computing these weights. Let $\boldsymbol{h}$ and $\boldsymbol{a}$ respectively denote the n-dimensional hub and authority weight vectors. Initially, all weights are set to 1. At each iteration the algorithm updates sequentially the hub and authority weights. For some node i, the authority weight of node i is set to be the sum of the hub weights of the nodes that point to i, while the hub weight of node i is the authority weight of the nodes pointed by i. In matrix-vector terms this is equivalent to setting setting $\boldsymbol{h} = W\boldsymbol{a}$ and $\boldsymbol{a} = W^T\boldsymbol{h}$. A normalization step is then applied, so that the vectors $\boldsymbol{a}$ and $\boldsymbol{h}$ become unit vectors in some norm. After a sufficient number of iterations, the vectors $\boldsymbol{a}$ and $\boldsymbol{h}$ converge to the principal eigenvectors of the matrices W^TW and WW^T, respectively. The vectors $\boldsymbol{a}$ and $\boldsymbol{h}$ respectively correspond to the right and left *singular vectors* of the matrix W, as these vectors are computed by the Singular Value Decomposition. The HITS algorithm returns the vector $\boldsymbol{a}$, the right singular vector of matrix W. More information on Singular Value Decomposition can be found in Section 4.1.

Independently from Kleinberg, Brin and Page developed the celebrated PageRank algorithm [Brin and Page 98]. The algorithm performs a random walk on the web graph, following links uniformly at random and occasionally resetting the random walk by jumping to a random page. The output of the algorithm is the stationary distribution of the random walk. The works of Kleinberg [Kleinberg 98] and Brin and Page [Brin and Page 98] were followed by numerous modifications and extensions [Bharat and Henzinger 98, Lempel and Moran 00, Rafiei and Mendelzon 00, Ng et al. 01b, Achlioptas et al. 01, Borodin et al. 01, Haveliwala 02, Tomlin 03, Tsaparas 04]. Of particular interest is the SALSA algorithm by Lempel and Moran [Lempel and Moran 00]. This is a hybrid between the HITS and PageRank algorithms that performs a random walk that alternates between hubs and authorities.

2.2. Theoretical Study of LAR Algorithms

In [Borodin et al. 01], Borodin et al. defined a theoretical framework for the study of LAR algorithms, which was later refined [Borodin et al. 05]. In their framework they provided formal definitions for stability and similarity between

LAR algorithms. They considered various algorithms, including HITS, SALSA, InDegree, and variants of HITS defined in their paper and proved that no pair of algorithms is similar, and, except for the InDegree algorithm, all other algorithms are unstable on the class of all possible graphs.

Ng, Zheng, and Jordan studied the stability of the HITS and PageRank algorithms [Ng et al. 01a]. Using perturbation theory (see [Stewart and Sun 90]), Ng et al. prove that the HITS algorithm is stable if the first and second singular values are well separated. Moreover, for the PageRank algorithm they show that the perturbation of the authority weights of the algorithm depends on the authority weight of the nodes whose outgoing links are changed. Their result was later improved by Bianchini et al. [Bianchini et al. 02]. Lee and Borodin [Lee and Borodin 03] modified the Borodin et al. definition of stability [Borodin et al. 01] so that the effect of a change in the graph to the output of the algorithm is allowed to depend upon the importance of the perturbed pages. They proved that, under this definition, the PageRank algorithm and a modified version of the SALSA algorithm are stable, while the HITS algorithm is unstable.

Borodin et al. defined also the notion of *rank stability* and *rank similarity*, where instead of considering the weights output by the algorithms, they considered the ordinal ranks induced by the weight vectors [Borodin et al. 05]. The results remain negative. The InDegree algorithm is the only algorithm that is rank stable, and no pair of algorithms is rank similar. Lempel and Moran extended their results to the class of irreducible graphs, and they also proved that the PageRank algorithm is rank unstable on this class [Lempel and Moran 03].

2.3. The Product Graph Model

Early measurements on the web graph [Kleinberg et al. 99, Kumar et al. 00b, Broder et al. 00] indicated that the in-degrees of the nodes in the web graph follow a power law distribution [Kleinberg et al. 99, Kumar et al. 00b, Broder et al. 00]. Following this discovery, it became clear that the Erdös-Rènyi random graph model [Erdős and Rèny 60] is not sufficient for modeling the web graph. This resulted in intensive research activity [Barabasi and Albert 99, Kumar et al. 00a, Aiello et al. 00, Aiello et al. 01, Pandurangan et al. 02, Pennock et al. 02] aimed at finding new stochastic models that adhere better with the characteristics of the World Wide Web. Product graphs (also known as random graphs with given expected degrees) were first considered as a model for the web graph by Azar et al. [Azar et al. 01]. The undirected case, where the h_i and a_i values are equal and edges are undirected, has been studied extensively, as a model for generating the Internet graph by Mihail and Papadimitriou [Mihail and Papadimitriou 02] and Chung et al. [Chung and Lu 02, Chung and Lu 03, Chung et al. 03, Chung et al. 04]. This model was introduced to study the case where the parameters follow a power law distribution. It has been shown [Mihail and Papadimitriou 02, Chung et al. 03, Chung et al. 04] that in this case the eigenvalues of the adjacency

matrix also follow a power law distribution, a fact that serves as an explanation for the observed power law distribution of the eigenvalues of the Internet graph [Faloutsos et al. 99]. Chung et al. also studied other related matrices [Chung et al. 04] and other properties of the product graphs, such as the average and maximum distance between two nodes [Chung and Lu 03] and the distribution of connected components [Chung and Lu 02].

3. The Theoretical Framework

In this section we review the definitions of Borodin et al. [Borodin et al. 05], and we extend them for classes of random graphs. Let $\mathcal{G}_n$ denote the set of all possible graphs of size n. The size of a graph is the number of nodes in the graph. Let $\overline{\mathcal{G}}_n \subseteq \mathcal{G}_n$ denote a collection of graphs in $\mathcal{G}_n$. Following the work of Borodin et al. [Borodin et al. 05], 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 produced by the algorithm $\mathcal{A}$ on graph G. The weight vector $\mathcal{A}(G)$ is normalized under some chosen norm L, that is, the algorithm maps the graphs in $\overline{\mathcal{G}}_n$ onto the unit L-sphere. Typically, the weights are normalized under some L_p norm. The L_p norm of a vector $\boldsymbol{w}$ is defined as $\|\boldsymbol{w}\|_p = \left(\sum_{i=1}^{n} |w_i|^p\right)^{1/p}$.

3.1. Distance Measures

In order to compare the behavior of different algorithms, or the behavior of the same algorithm on different graphs, Borodin et al. defined various distance measures between authority weight vectors [Borodin et al. 05]. The distance functions we consider are defined using the L_q norm as well. The d_q distance between two weight vectors $\boldsymbol{w}_1$ and $\boldsymbol{w}_2$ is defined as follows:

$$d_q(\boldsymbol{w}_1, \boldsymbol{w}_2) = \min_{\gamma_1, \gamma_2 \geq 1} \|\gamma_1 \boldsymbol{w}_1 - \gamma_2 \boldsymbol{w}_2\|_q .$$

The constants γ_1 and γ_2 serve the purpose of alleviating differences due to different normalization factors. When using distance d_q we will assume that the vectors are normalized in the L_q norm. In this paper we consider mainly the d_2 distance measure. We can prove that $d_2(\boldsymbol{a}, \boldsymbol{b}) = \|\boldsymbol{a} - \boldsymbol{b}\|_2$, and thus the d_2 distance is a metric. For the following lemma and the proof, we use $\|\cdot\|$ to denote the L_2 norm.

Lemma 3.1. *Let $\boldsymbol{a}$ and $\boldsymbol{b}$ be two unit vectors in the L_2 norm. For the distance measure d_2, we have that $d_2(\boldsymbol{a}, \boldsymbol{b}) = \|\boldsymbol{a} - \boldsymbol{b}\|$.*

Proof. By definition of the d_2 distance measure for any two weight vectors $\boldsymbol{a}$ and $\boldsymbol{b}$, we have that $d_2(\boldsymbol{a}, \boldsymbol{b}) \leq \|\boldsymbol{a} - \boldsymbol{b}\|$. We will now prove that $d_2(\boldsymbol{a}, \boldsymbol{b}) \geq \|\boldsymbol{a} - \boldsymbol{b}\|$, which implies that $d_2(\boldsymbol{a}, \boldsymbol{b}) = \|\boldsymbol{a} - \boldsymbol{b}\|$.

Borodin et al. proved that at least one of the constants γ_1 and γ_2 should be equal to 1. Without loss of generality, assume that $\gamma_1 = 1$ [Borodin et al. 05]. We have that $d_2(\boldsymbol{a}, \boldsymbol{b}) = \min_{\gamma \geq 1} \|\boldsymbol{a} - \gamma \boldsymbol{b}\|$. Given two vectors $\boldsymbol{a}$ and $\boldsymbol{b}$, let $\cos(\boldsymbol{a}, \boldsymbol{b})$ denote the cosine of the angle of the vectors $\boldsymbol{a}$ and $\boldsymbol{b}$. For two unit vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ it is easy to show that $\|\boldsymbol{a} - \boldsymbol{b}\|^2 = 2 - 2\cos(\boldsymbol{a}, \boldsymbol{b})$. We also have that

$$\begin{aligned} \|\boldsymbol{a} - \gamma \boldsymbol{b}\|^2 &= \|\boldsymbol{a}\|^2 + \|\gamma \boldsymbol{b}\|^2 - 2\|\boldsymbol{a}\|\|\gamma \boldsymbol{b}\| \cos(\boldsymbol{a}, \gamma \boldsymbol{b}) \\ &\geq 2\gamma - 2\gamma \cos(\boldsymbol{a}, \boldsymbol{b}) \geq \|\boldsymbol{a} - \boldsymbol{b}\|^2. \end{aligned}$$

The first inequality follows from the fact that $1 + \gamma^2 \geq 2\gamma$. □

3.2. Similarity

Borodin et al. [Borodin et al. 05] give the following general definition of similarity for any distance function d and any normalization norm L. In the following we define $M_n(d, L) = \sup_{\|\boldsymbol{w}_1\| = \|\boldsymbol{w}_2\| = 1} d(\boldsymbol{w}_1, \boldsymbol{w}_2)$ to be the maximum distance between any two n-dimensional vectors with unit norm $L = ||\cdot||$.

Definition 3.2. Algorithms $\mathcal{A}_1$ and $\mathcal{A}_2$ are (L, d)-similar on the class $\overline{\mathcal{G}}_n$ if as $n \to \infty$,

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

Consider now the case that the class $\overline{\mathcal{G}}_n$ is a class of random graphs, generated according to some random process. That is, we define a probability space $\langle \overline{\mathcal{G}}_n, \mathcal{P} \rangle$, where $\mathcal{P}$ is a probability distribution over the class $\overline{\mathcal{G}}_n$. We extend the definition of similarity on the class $\overline{\mathcal{G}}_n$ as follows.

Definition 3.3. Algorithms $\mathcal{A}_1$ and $\mathcal{A}_2$ are (L, d)-similar *with high probability* on the class of random graphs $\overline{\mathcal{G}}_n$ if for a graph G drawn from $\overline{\mathcal{G}}_n$, as $n \to \infty$,

$$d\left(\mathcal{A}_1(G), \mathcal{A}_2(G)\right) = o\left(M_n(d, L)\right)$$

with probability $1 - o(1)$.

We note that when we consider (L_q, d_q)-similarity, we have that $M_n(d_q, L_q) = \Theta(1)$. Furthermore, if the distance function d is a metric, or a near metric,[1] then the transitivity property holds. It is easy to show that if algorithms $\mathcal{A}_1$ and

[1] A near metric [Fagin et al. 03] is a distance function that is reflexive and symmetric, and there exists a constant c independent of n such that, for all $k > 0$ and all vectors $\boldsymbol{u}, \boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_k, \boldsymbol{v}$, $d(\boldsymbol{u}, \boldsymbol{v}) \leq c(d(\boldsymbol{u}, \boldsymbol{w}_1) + d(\boldsymbol{w}_1, \boldsymbol{w}_2) + \cdots + d(\boldsymbol{w}_k, \boldsymbol{v}))$.

$\mathcal{A}_2$ are similar (with high probability), and algorithms $\mathcal{A}_2$ and $\mathcal{A}_3$ are similar (with high probability), then algorithms A_1 and $\mathcal{A}_3$ are also similar (with high probability).

3.3. Stability

Let $\overline{\mathcal{G}}_n$ be a class of graphs, and let $G = (P, E)$ and $G' = (P, E')$ be two graphs in $\overline{\mathcal{G}}_n$. The *link distance* d_ℓ between graphs G and G' is defined as $d_\ell(G, G') = |(E \cup E') \setminus (E \cap E')|$. 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.

Given a class of graphs $\overline{\mathcal{G}}_n$, let $\mathcal{C}_k(G) = \{G' \in \overline{\mathcal{G}}_n : d_\ell(G, G') \leq k\}$ denote the set of all graphs that have link distance at most k from graph G. Borodin et al. [Borodin et al. 05] give the following definition of stability.

Definition 3.4. [Borodin et al. 05] An algorithm $\mathcal{A}$ is (L, d)-stable on the class of graphs $\overline{\mathcal{G}}_n$ if for every fixed positive integer k, we have as $n \to \infty$,

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

Given a class of random graphs $\overline{\mathcal{G}}_n$ we define stability with high probability as follows.

Definition 3.5. An algorithm $\mathcal{A}$ is (L, d)-stable *with high probability* on the class of random graphs $\overline{\mathcal{G}}_n$ if for every fixed positive integer k and for a graph G drawn from $\overline{\mathcal{G}}_n$, we have as $n \to \infty$,

$$\max_{G' \in \mathcal{C}_k(G)} d(\mathcal{A}(G), \mathcal{A}(G')) = o\left(M_n(d, L)\right)$$

with probability $1 - o(1)$.

3.4. Stability and Similarity

The following lemma shows the connection between stability and similarity. The lemma is a generalization of a lemma by Borodin et al. [Borodin et al. 05].

Lemma 3.6. *Let d be a metric or near metric distance function, L a norm, and $\overline{\mathcal{G}}_n$ a class of random graphs. If algorithm $\mathcal{A}_1$ is (L, d)-stable with high probability on the class $\overline{\mathcal{G}}_n$ and algorithm $\mathcal{A}_2$ is (L, d)-similar to A_1 with high probability on the class $\overline{\mathcal{G}}_n$, then A_2 is (L, d)-stable with high probability on the class $\overline{\mathcal{G}}_n$.*

Proof. Let $G \in \overline{\mathcal{G}}_n$ be a graph drawn from the class $\overline{\mathcal{G}}_n$. Also, let $M = M_n(d, L)$. Since $\mathcal{A}_1$ and $\mathcal{A}_2$ are (L, d)-similar with high probability on the class $\overline{\mathcal{G}}_n$, it

follows that

$$p_1 = Pr[d(\mathcal{A}_2(G), \mathcal{A}_1(G)) = \Omega(M)] = o(1).$$

Furthermore, since A_1 is (L, d)-stable with high probability on the class $\overline{\mathcal{G}}_n$, we have that

$$p_2 = Pr[\max_{G' \in \mathcal{C}_k(G)} d(\mathcal{A}_1(G), \mathcal{A}_1(G')) = \Omega(M)] = o(1).$$

Define the graphs

$$G_1 = \arg \max_{G' \in \mathcal{C}_k(G)} d(\mathcal{A}_1(G), \mathcal{A}_1(G')),$$
$$G_2 = \arg \max_{G' \in \mathcal{C}_k(G)} d(\mathcal{A}_2(G), \mathcal{A}_2(G')).$$

By definition of the graph G_1, we have that $d(\mathcal{A}_1(G), \mathcal{A}_1(G_2)) \leq d(\mathcal{A}_1(G), \mathcal{A}_1(G_1))$, thus

$$p_3 = Pr[d(\mathcal{A}_1(G), \mathcal{A}_1(G_2)) = \Omega(M)] = o(1).$$

From the metric or near metric property of the function d, we have that

$$d(\mathcal{A}_2(G), \mathcal{A}_2(G_2)) \leq$$
$$c\,(d(\mathcal{A}_2(G), \mathcal{A}_1(G)) + d(\mathcal{A}_1(G), \mathcal{A}_1(G_2)) + d(\mathcal{A}_1(G_2), \mathcal{A}_2(G_2)))$$

Therefore, $Pr[d(\mathcal{A}_2(G), \mathcal{A}_2(G_2)) = \Omega(M)] \leq p_1 + p_2 + p_3 = o(1)$. Therefore, $\mathcal{A}_2$ is (L, d)-stable with high probability. □

4. Stability and Similarity on the Class of Product Graphs

The class of product graphs $\mathcal{G}_n^p(\boldsymbol{h}, \boldsymbol{a})$ (or, for brevity, $\mathcal{G}_n^p$) is defined with two parameters $\boldsymbol{h}$ and $\boldsymbol{a}$, which are two n-dimensional real vectors, with h_i and a_i taking values in $[0, 1]$. These two vectors can be thought of as the *latent* hub and authority vectors, respectively. A link is generated from node i to node j with probability $h_i a_j$.

Let $G \in \mathcal{G}_n^p$, and let W be the adjacency matrix of the graph G. Following Azar et al. [Azar et al. 01], we express the matrix W as $W = \boldsymbol{h}\boldsymbol{a}^T + R$, where R is a random matrix such that

$$R[i, j] = \begin{cases} -h_i a_j & \text{with probability } 1 - h_i a_j, \\ 1 - h_i a_j & \text{with probability } h_i a_j. \end{cases}$$

We can think of the matrix W as a perturbation of the matrix $M = \boldsymbol{h}\boldsymbol{a}^T$ by the matrix R. We refer to matrix R as the *rounding* matrix, which rounds the entries

of M to 0 or 1. The matrix M is a rank-one matrix since all columns (rows) are multiples of the same vector. If we run HITS on the matrix M (assuming a small modification of the algorithm so that it runs on weighted graphs), the algorithm will reconstruct the latent vectors $\boldsymbol{a}$ and $\boldsymbol{h}$, which are the singular vectors of matrix M. Note also that if we run the InDegree algorithm on the matrix M (assuming again that we take the weighted in-degrees), the algorithm will also output the latent vector $\boldsymbol{a}$. So, on rank-one matrices the two algorithms are identical. The question is how the addition of the rounding matrix R affects the output of the two algorithms. We will show that it has only a small effect and that the two algorithms remain similar.

More formally, let Latent denote the (imaginary) LAR algorithm that, for any graph G in the class $\mathcal{G}_n^p(\boldsymbol{h}, \boldsymbol{a})$, outputs the vector $\boldsymbol{a}$. We will show that both HITS and InDegree are similar to Latent with high probability. This implies that the two algorithms are similar with high probability. Furthermore, we will show that it also implies the stability of the HITS algorithm.

4.1. Mathematical Tools

We now introduce some mathematical tools that we will use for the remaining of this section.

4.1.1. Matrix Norms. Let M be an $n \times n$ matrix. The L_2 norm, $\|M\|_2$ (also referred to as the spectral norm), and the Frobenius norm, $\|M\|_F$, of matrix M are defined as follows:

$$\|M\|_2 = \max_{v:\|v\|=1} \|Mv\|_2$$

and

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

Both norms are unitary invariant. That is, for unitary matrices U and V (i.e., $U^T U = V^T V = I$), we have that $\|U^T M V\| = \|M\|$. For the L_2 norm we have that $\|U\|_2 = \|V\|_2 = 1$. Furthermore, both norms are consistent, that is, for any two matrices M and W, we have that $\|MW\| \leq \|M\|\|W\|$. The two norms are related by the inequality $\|M\|_2 \leq \|M\|_F \leq \sqrt{n}\|M\|_2$.

4.1.2. Singular Value Decomposition. Let M be an $n \times n$ matrix. The *Singular Value Decomposition* of the matrix M is a factorization of the form $M = U\Sigma V^T$, where U and V are $n \times n$ unitary matrices and Σ is a diagonal matrix, $\Sigma = \text{diag}(\sigma_1, \sigma_2, \ldots, \sigma_n)$, where $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_n \geq 0$. The values $\sigma_1, \ldots, \sigma_n$ are called the *singular values* of the matrix M. The pair $(\boldsymbol{u}_k, \boldsymbol{v}_k)$ of the kth column vectors of matrices U and V, respectively, is a pair of the kth principal *singular vectors* of the matrix M. The column vectors of U are the left singular vectors of M, and the columns of V are the right singular vectors of M. The left singular

vectors of M are also the eigenvectors of MM^T, while the right singular vectors of M are the eigenvectors of M^TM. Given the Singular Value Decomposition of M, we can express the matrix M as $M = \sum_{i=1}^{n} \sigma_i \boldsymbol{u}_i \boldsymbol{v}_i^T$, that is, as the sum of n rank-one matrices. We can think of each of these matrices as capturing a linear trend in the vector space defined by M. The value of the corresponding singular value captures the strength of the linear trend.

The matrix norms can be computed using the singular values. Specifically, we have that $\|M\|_2 = \sigma_1$ and $\|M\|_F^2 = \sigma_1^2 + \sigma_2^2 + \cdots + \sigma_n^2$. Furthermore, let $M_k = \sum_{i=1}^{k} \sigma_i \boldsymbol{u}_i \boldsymbol{v}_i^T$ denote a rank-k approximation of the matrix M. It can be proved that M_k is the best rank-k approximation with respect to both the L_2 and Frobenius norms.

4.1.3. Perturbation Theory. Perturbation theory studies how adding a perturbation matrix E to a matrix M affects the eigenvalues and eigenvectors of M. Let G and G' be two graphs, and let W and W' denote the respective adjacency matrices. The matrix W' can be written as $W' = W + E$, where E is a matrix with entries in $\{-1, 0, 1\}$. The entry $E[i, j]$ is 1 if we add a link from i to j and -1 if we remove a link from i to j. Therefore, we can think of the matrix W' as a perturbation of the matrix W by a matrix E. Note that if we assume that only a constant number of links is added and removed, then both the Frobenius and the L_2 norms of E are bounded by a constant.

We now introduce an important lemma that we will use in the following.

Lemma 4.1. *Let W be a matrix, and let $W + E$ be a perturbation of the matrix. Let $\boldsymbol{u}$ and $\boldsymbol{v}$ respectively denote the left and right principal singular vectors of the matrix W, and $\boldsymbol{u}'$ and $\boldsymbol{v}'$ the principal singular vectors of the perturbed matrix. Let σ_1 and σ_2 denote the first and second singular values of the matrix W. If $\sigma_1 - \sigma_2 = \omega(\|E\|_2)$, then $\|\boldsymbol{u}' - \boldsymbol{u}\|_2 = o(1)$ and $\|\boldsymbol{v}' - \boldsymbol{v}\|_2 = o(1)$.*

Lemma 4.1 says that if the "eigengap" between the first and the second singular values of matrix M is large with respect to the L_2 norm (singular value) of the perturbation matrix E, then adding E to M will cause only a small perturbation to the principal singular vectors of M. The proof of the lemma appears in Section 9. Intuitively, the first and the second singular values respectively capture the strength of the strongest and second strongest linear trends in matrix M, while $\|E\|_2$ captures the strongest linear trend in the perturbation matrix E. The value of $\|E\|_2$ also captures the effect that E can have on the matrix M. If the separation between σ_1 and σ_2 is larger than $\|E\|_2$, then adding E to M cannot cause another linear trend to emerge as dominant, and thus the principal singular vectors are not significantly perturbed.

4.1.4. Norms of random matrices. We also make use of the following theorem for concentration bounds on the L_2 norm of random symmetric matrices. We state the theorem as it appears in [Achlioptas and McSherry 01].

Theorem 4.2. *Given an $m \times n$ matrix A and any $\epsilon > 0$, let $\widehat{A}$ be any random matrix such that for all i, j: $E[\widehat{A}_{ij}] = A_{ij}$, $Var(\widehat{A}_{ij}) \leq \sigma^2$, and $|\widehat{A}_{ij} - A_{ij}| \leq K$, where*

$$K = \left(\frac{4\epsilon}{4+3\epsilon}\right)^3 \frac{\sigma\sqrt{m+n}}{\log^3(m+n)}.$$

For any $\alpha > 0$ and $m + n \geq 20$, with probability at least $1 - (m+n)^{-\alpha^2}$,

$$\|\widehat{A} - A\|_2 < (2 + \alpha + \epsilon)\sigma\sqrt{m+n}.$$

4.1.5. Chernoff bounds. We will make use of standard Chernoff bounds. The following theorem can be found in the textbook of Motwani and Raghavan [Motwani and Raghavan 95].

Theorem 4.3. *Let $X_1, X_2, \ldots, X_n$ be independent Poisson trials such that, for $1 \leq i \leq n$, $\Pr[X_i = 1] = p_i$, where $0 \leq p_i \leq 1$. Let $X = \sum_{i=1}^n X_i$, and $\mu = E[X] = \sum_{i=1}^n p_i$. Then, for $0 < \delta \leq 1$, we have that*

$$\Pr[X < (1-\delta)\mu] < exp(-\mu\delta^2/2), \tag{4.1}$$

$$\Pr[X > (1+\delta)\mu] < exp(-\mu\delta^2/4). \tag{4.2}$$

4.2. Conditions for the Stability of HITS

We first provide general conditions for the stability of the HITS algorithm. Let $\mathcal{G}_n^\sigma$ denote the class of graphs with adjacency matrix W that satisfies $\sigma_1(W) - \sigma_2(W) = \omega(1)$. The proof of the following theorem follows directly from Lemma 4.1, and the fact that the perturbation matrix E has L_2 norm bounded by a constant.

Theorem 4.4. *The HITS algorithm is (L_2, d_2)-stable on the class of graphs $\mathcal{G}_n^\sigma$.*

Proof. The proof follows directly from Lemma 4.1. Given a graph $G \in \mathcal{G}_n^\sigma$ with adjacency matrix W and a graph $G' \in \mathcal{C}_k(G)$ with adjacency matrix W', let $E = W - W'$. We have $\|E\|_2 \leq \|E\|_F = \sqrt{k}$. Therefore, $\sigma_1 - \sigma_2 = \omega(\|E\|_2)$. If $\boldsymbol{a}$ and $\boldsymbol{a}'$ are the weight vectors of the HITS algorithm (normalized under the L_2 norm) on the graphs G and G', then $\|\boldsymbol{a} - \boldsymbol{a}'\|_2 = o(1)$. □

Theorem 4.4 provides a sufficient condition for the stability of HITS on general graphs, and it will be useful when considering stability on the class of product graphs.

The class $\mathcal{G}_n^\sigma$ is actually a subset of the class defined by the result of Ng et al. [Ng et al. 01a]. Translating their result in the framework of Borodin et al.

[Borodin et al. 05], they prove that the HITS algorithm is stable on the class of graphs with $\sigma_1(W)^2 - \sigma_2(W)^2 = \omega(\sqrt{d})$, where d is the maximum out-degree. Note that we can rewrite this as $\sigma_1 - \sigma_2 = \omega(\frac{\sqrt{d}}{\sigma_1+\sigma_2})$. This is a weaker condition than $\sigma_1(W) - \sigma_2(W) = \omega(1)$. We will show this, by showing that $\frac{\sqrt{d}}{\sigma_1+\sigma_2} \leq 1$.

We have that $\sigma_1+\sigma_2 \geq \sigma_1$. Furthermore, by definition, $\sigma_1 = max_{x:\|x\|=1}\|Wx\|$. Let i be the node with the maximum out-degree, and let W_i be the ith row in the matrix W. Setting $x = W_i/\sqrt{d}$, we have that $\|Wx\| \geq \sqrt{d}$. Therefore, $\sigma_1 \geq \sqrt{d}$, and $\frac{\sqrt{d}}{\sigma_1+\sigma_2} \leq 1$.

4.3. Similarity of HITS and Latent

We now turn our attention to product graphs, and we prove that HITS and Latent are similar on this class. A result of similar spirit is shown in the work of Azar et al. [Azar et al. 01]. We make the following assumption for the vectors $\boldsymbol{a}$ and $\boldsymbol{h}$.

Assumption 4.5. For the class $\mathcal{G}_n^p(\boldsymbol{h}, \boldsymbol{a})$, the latent vectors $\boldsymbol{a}$ and $\boldsymbol{h}$ satisfy $\|\boldsymbol{a}\|_2\|\boldsymbol{h}\|_2 = \omega(\sqrt{n})$.

As we show below, Assumption 4.5 places a direct lower bound on the principal singular value of the matrix $M = \boldsymbol{h}\boldsymbol{a}^T$. Therefore, we require that the matrix M defines a strong linear trend that will not disappear when perturbing by the rounding matrix R. In the resulting adjacency matrix, this linear trend will translate to a tightly knit community of hubs and authorities in the graph.

Furthermore, let $A = \sum_{i=1}^n a_i$ denote the sum of the authority values, and $H = \sum_{j=1}^n h_j$ the sum of the hub values. Since the values are positive, we have $A = \|\boldsymbol{a}\|_1$ and $H = \|\boldsymbol{h}\|_1$. The product HA is equal to expected number of edges in the graph. We have that $HA \geq \|\boldsymbol{a}\|_2\|\boldsymbol{h}\|_2$, thus, from Assumption 4.5, $HA = \omega(\sqrt{n})$. However, this lower bound seems too weak; it does not seem possible to satisfy Assumption 4.5 while $HA = \Theta(\sqrt{n})$. We also have that $HA \leq n\|\boldsymbol{a}\|_2\|\boldsymbol{h}\|_2$, thus Assumption 4.5 can be satisfied by requiring that $HA = \omega(n^{3/2})$, which implies that the underlying graph is dense. However, it is possible to satisfy Assumption 4.5 while $HA = o(n^{\frac{3}{2}})$. For example, if we set all the values of $\boldsymbol{h}$ to some value $c = \Theta(1)$, $\log n$ values of $\boldsymbol{a}$ to c as well, and the remaining values of $\boldsymbol{a}$ to $1/n$, then we have that $\|\boldsymbol{h}\|_2\|\boldsymbol{a}\|_2 = \Theta(\sqrt{n \log n})$ and $HA = \Theta(n \log n)$. Note that these $\Theta(n \log n)$ edges define the tightly knit community in the underlying graph.

Lemma 4.6. *The algorithms HITS and Latent are (L_2, d_2)-similar with high probability on the class $\mathcal{G}_n^p$, subject to Assumption 4.5.*

Proof. The singular vectors of the matrix M are the L_2-unit vectors $\boldsymbol{a}_2 = \boldsymbol{a}/\|\boldsymbol{a}\|_2$ and $\boldsymbol{h}_2 = \boldsymbol{h}/\|\boldsymbol{h}\|_2$. The matrix M can be expressed as $M = \boldsymbol{h}_2\|\boldsymbol{h}\|_2\|\boldsymbol{a}\|_2\boldsymbol{a}_2^T$. Therefore, the principal singular value of M is $\sigma_1 = \|\boldsymbol{h}\|_2\|\boldsymbol{a}\|_2 = \omega(\sqrt{n})$. Since M is rank-one, $\sigma_i = 0$, for all $i = 2, 3, \ldots, n$. Therefore, for matrix M we have that $\sigma_1 - \sigma_2 = \omega(\sqrt{n})$.

Matrix R is a random matrix, where each entry is a independent random variable with mean 0 and the maximum value and variance bounded by 1. Using Theorem 4.2, we observe that $K = 1$ and $\sigma = 1$. Setting $\epsilon = 1$ and $\alpha = 1$, we get that $Pr[\|R\|_2 \leq 8\sqrt{n}] \geq 1 - o\,(1/n)$, thus $\|R\|_2 = O(\sqrt{n})$ with high probability.

Therefore,we have that $\sigma_1 - \sigma_2 = \omega(\|R\|_2)$ with probability $1 - o(1)$. If $\boldsymbol{w}_2$ is the right singular vector of matrix W normalized in the L_2 norm, then using Lemma 4.1, we have that $\|\boldsymbol{w}_2 - \boldsymbol{a}_2\|_2 = o(1)$ with probability $1 - o(1)$. □

Assumption 4.5 guarantees also the stability of HITS on $\mathcal{G}_n^p$. The proof follows from the fact that if $G \in \mathcal{G}_n^p$, then $G \in \mathcal{G}_n^\sigma$ with high probability, that is, the resulting matrix has a large eigengap between the first and the second singular values. This follows from the fact that the "base" rank-1 matrix M has eigengap significantly larger than the singular value of the rounding matrix R. Adding the rounding matrix R to M cannot decrease the eigengap significantly.

Theorem 4.7. *The HITS algorithm is (L_2, d_2)-stable with high probability on the class of graphs $\mathcal{G}_n^p$, subject to Assumption 4.5.*

Proof. Assumption 4.5 guarantees that the principal singular value of matrix M is $\omega(\sqrt{n})$. Furthermore, since the matrix M is a rank-one matrix, $\sigma_2 = 0$, thus $\sigma_1 - \sigma_2 = \omega(\sqrt{n})$. The L_2 norm of the rounding matrix R is $O(\sqrt{n})$ with high probability. Perturbation theory [Stewart and Sun 90] guarantees that the singular values of the matrix M cannot be perturbed more than $\|R\|_2$, that is $|\sigma_i(M + R) - \sigma_i(M)| \leq \|R\|_2$, for every singular value σ_i. We have that $\sigma_1(M) = \omega(\sqrt{n})$; therefore, $\sigma_1(M + R) = \omega(\sqrt{n})$. Furthermore, $\sigma_2(M) = 0$, so $\sigma_2(M + R) = O(\sqrt{n})$. It follows that for the matrix $W = M + R$ we have that $\sigma_1(W) - \sigma_2(W) = \omega(\sqrt{n})$ with high probability. From Theorem 4.4 it follows that HITS is stable on $\mathcal{G}_n^p$ with high probability. □

4.4. Similarity of InDegree and Latent

We now consider the (L_q, d_q)-similarity of InDegree and Latent, for all $1 \leq q < \infty$. Again, let $A = \sum_{i=1}^n a_i$, and let $H = \sum_{j=1}^n h_j$. Also, let $\boldsymbol{d}$ denote the vector of the InDegree algorithm before any normalization is applied. That is, d_i is the in-degree of node i. For some node i, we have that

$$d_i = \sum_{j=1}^{n} W[j,i] = \sum_{j=1}^{n} M[j,i] + \sum_{j=1}^{n} R[j,i].$$

We have that $\sum_{j=1}^{n} M[j,i] = Ha_i$. Furthermore, let $r_i = \sum_{j=1}^{n} R[j,i]$, and let $\boldsymbol{r} = [r_1, \ldots, r_n]^T$. The vector $\boldsymbol{d}$ can be expressed as $\boldsymbol{d} = H\boldsymbol{a} + \boldsymbol{r}$. The vector $H\boldsymbol{a}$ is the vector of expected degrees, and thus $\boldsymbol{r}$ is the vector of the deviations of the actual degrees from their expected values. We will now show that if the L_q-norm of the vector of the expected degrees is large, then the L_q norm of $\boldsymbol{r}$ is small relative to that of $H\boldsymbol{a}$.

Lemma 4.8. *For every $q \in [1, \infty)$, if $H\|\boldsymbol{a}\|_q = \omega(n^{1/q} \ln n)$, then $\|\boldsymbol{r}\|_q = o(H\|\boldsymbol{a}\|_q)$ with high probability.*

Proof. For the following we will use $\|\cdot\|$ to denote the L_q norm, for some $q \in [1, \infty)$. We will prove that $\|\boldsymbol{r}\| = o(H\|\boldsymbol{a}\|)$ with probability at least $1-1/n$. It is sufficient to show that, for all $1 \leq i \leq n$, $r_i = o(Ha_i)$ with probability $1 - 1/n^2$. We note again that $r_i = d_i - Ha_i$, so essentially we need to bound the deviation of d_i from its expectation. When Ha_i is large, that is, $Ha_i = \omega(\ln n)$, this is easy to do, using standard Chernoff bounds. The bounds cannot be applied when Ha_i is small, that is, $Ha_i = O(\ln n)$. However, in this case, although r_i is comparable to Ha_i, it is also small, and it does not contribute much to the norm $\|\boldsymbol{r}\|$. It thus suffices to show that $|r_i| = O(\ln n)$ with probability at least $1 - 1/n^2$. If for all $1 \leq i \leq n$, $|r_i| = O(\ln n)$, then $\|\boldsymbol{r}\| = O(n^{1/q} \ln n) = o(H\|\boldsymbol{a}\|)$.

We thus partition the nodes into two sets S and B. Set S contains all nodes such that $Ha_i = O(\ln n)$, that is, nodes with "small" expected in-degree, and set B contains all nodes such that $Ha_i = \omega(\ln n)$, that is, nodes with "big" expected in-degree.

Consider a node $i \in S$. We have that $Ha_i \leq c \ln n$, for some constant c. Using Equation (4.2) from Theorem 4.3, we set $\delta = k \ln n/(Ha_i)$, where k is a constant such that $k \geq \sqrt{8c}$, and we get that $Pr[d_i - Ha_i \geq k \ln n] \leq exp(-2 \ln n)$. Therefore, for all nodes in S we have that $|r_i| = O(\ln n)$ with probability at least $1 - 1/n^2$. This implies that $\sum_{i \in S} |r_i|^q = O(n \ln^q n) = o(H^q \|\boldsymbol{a}\|^q)$, with probability $1 - 1/n$.

Consider now a node $i \in B$. We have that $Ha_i = \omega(\ln n)$, thus $Ha_i = (\ln n)/s(n)$, where $s(n)$ is a function such that $s(n) = o(1)$. Using Theorem 4.3, we set $\delta = k\sqrt{s(n)}$, where k is a constant such that $k \geq \sqrt{8}$, and we get that $Pr[|d_i - Ha_i| \geq \delta Ha_i] \leq \exp(-2 \ln n)$. Therefore, for the nodes in B, we have that $|r_i| = o(Ha_i)$ with probability at least $1 - 1/n^2$. Thus, $\sum_{i \in B} |r_i|^q = o(H^q \|\boldsymbol{a}\|^q)$, with probability $1 - 1/n$.

Putting everything together we have that $\|\boldsymbol{r}\|^q = \sum_{i \in S} |r_i|^q + \sum_{i \in B} |r_i|^q = o(H^q \|\boldsymbol{a}\|^q)$, with probability $1-2/n$. Therefore, $\|\boldsymbol{r}\| = o(H\|\boldsymbol{a}\|)$ with probability $1 - 2/n$. This concludes our proof. □

We are now ready to prove the similarity of InDegree and Latent. The following lemma follows from Lemma 4.8.

Lemma 4.9. *For every $q \in [1, \infty)$, the InDegree and Latent algorithms are (L_q, d_q)-similar with high probability on the class $\mathcal{G}_n^p$, when the latent vectors $\boldsymbol{a}$ and $\boldsymbol{h}$ satisfy $H\|\boldsymbol{a}\|_q = \omega(n^{1/q} \ln n)$.*

Proof. For the following we will use $\|\cdot\|$ to denote the L_q norm, for some $q \in [1, \infty)$. Let $\boldsymbol{d}_q$ and $\boldsymbol{a}_q$ respectively denote the $\boldsymbol{d}$ and $\boldsymbol{a}$ vectors when normalized under the L_q norm. We will now bound the difference $\|\gamma_1 \boldsymbol{a}_q - \gamma_2 \boldsymbol{d}_q\|$ for $\gamma_1, \gamma_2 \geq 1$.

First we observe that since $\boldsymbol{d} = H\boldsymbol{a} + \boldsymbol{r}$, using norm properties, we can easily show that

$$H\|\boldsymbol{a}\| - \|\boldsymbol{r}\| \leq \|\boldsymbol{d}\| \leq H\|\boldsymbol{a}\| + \|\boldsymbol{r}\|.$$

Since we have that $\|\boldsymbol{r}\| = o(H\|\boldsymbol{a}\|)$, it follows that $\|\boldsymbol{d}\| = \Theta(H\|\boldsymbol{a}\|)$.

Now consider two cases. If $\|\boldsymbol{d}\| \geq H\|\boldsymbol{a}\|$, then let $\gamma_1 = 1$ and $\gamma_2 = \frac{\|\boldsymbol{d}\|}{H\|\boldsymbol{a}\|} \geq 1$. We have that

$$\|\gamma_1 \boldsymbol{a}_q - \gamma_2 \boldsymbol{d}_q\| = \left\| \frac{\boldsymbol{a}}{\|\boldsymbol{a}\|} - \frac{\|\boldsymbol{d}\|}{H\|\boldsymbol{a}\|} \frac{H\boldsymbol{a} + \boldsymbol{r}}{\|\boldsymbol{d}\|} \right\| = \frac{\|\boldsymbol{r}\|}{H\|\boldsymbol{a}\|} .$$

If $\|\boldsymbol{d}\| \leq H\|\boldsymbol{a}\|$, then let $\gamma_1 = \frac{H\|\boldsymbol{a}\|}{\|\boldsymbol{d}\|} \geq 1$ and $\gamma_2 = 1$. We have that

$$\|\gamma_1 \boldsymbol{a}_q - \gamma_2 \boldsymbol{d}_q\| = \left\| \frac{H\|\boldsymbol{a}\|}{\|\boldsymbol{d}\|} \frac{\boldsymbol{a}}{\|\boldsymbol{a}\|} - \frac{H\boldsymbol{a} + \boldsymbol{r}}{\|\boldsymbol{d}\|} \right\| \leq \frac{\|\boldsymbol{r}\|}{\|\boldsymbol{d}\|} \leq c \frac{\|\boldsymbol{r}\|}{H\|\boldsymbol{a}\|}$$

for some constant c, such that $\|\boldsymbol{d}\| \geq cH\|\boldsymbol{a}\|$.

Therefore, we have that $\|\gamma_1 \boldsymbol{a}_q - \gamma_2 \boldsymbol{d}_q\| \leq c \frac{\|\boldsymbol{r}\|}{H\|\boldsymbol{a}\|}$. When $H\|\boldsymbol{a}\| = \omega(n^{1/q} \ln n)$, we have that $\|\boldsymbol{r}\| = o(H\|\boldsymbol{a}\|)$. Therefore, $\|\gamma_1 \boldsymbol{a}_q - \gamma_2 \boldsymbol{d}_q\| = o(1)$, which concludes the proof. □

We now make the following assumption for vectors $\boldsymbol{a}$ and $\boldsymbol{h}$.

Assumption 4.10. For the class $\mathcal{G}_n^p(\boldsymbol{h}, \boldsymbol{a})$, the latent vectors $\boldsymbol{a}$ and $\boldsymbol{h}$ satisfy $H\|\boldsymbol{a}\|_2 = \omega(\sqrt{n} \ln n)$.

Assumption 4.10 places a direct lower bound on the L_2-norm of the expected in-degree sequence. Using the fact that $\|\boldsymbol{a}\|_2 \leq A \leq \sqrt{n}\|\boldsymbol{a}\|_2$, we can show that if the assumption is satisfied then $HA = \omega(\sqrt{n} \ln n)$. Furthermore, we can satisfy Assumption 4.10 by requiring $HA = \omega(n \ln n)$, placing a direct bound on the expected number of edges in the graph. We note that for the same expected number of edges, the L_2 norm will produce higher values if the expected in-degree sequence is uneven (in the extreme case where all edges are attached to a single node, the two norms are the same).

The InDegree and Latent algorithms are (L_2, d_2)-similar subject to Assumption 4.10. The following theorem follows from the transitivity property of similarity.

Theorem 4.11. *The HITS and InDegree algorithms are (L_2, d_2)-similar with high probability on the class $\mathcal{G}_n^p$, subject to Assumptions 4.5 and 4.10.*

Intuitively, a graph that belongs to class $\mathcal{G}_n^p$ that satisfies both Assumptions 4.5 and 4.10 should be dense enough, and it should contain a large enough tightly knit community. We note that we can satisfy both Assumptions 4.5 and 4.10 either by requiring that $HA = \omega(n^{3/2})$, or by requiring that $\sigma_1(M) = \|\boldsymbol{h}\|_2\|\boldsymbol{a}\|_2 = \omega(\sqrt{n}\ln n)$.

5. Case Studies

5.1. The Erdős–Rèny Model

The Erdős–Rèny $\mathcal{G}(n,p)$ model is a special case of the product model, when we set $a_i = h_i = \sqrt{p}$, for all $1 \le i \le n$. Therefore, Assumptions 4.5 and 4.10 can be used to derive conditions on p that guarantee the stability of HITS and its similarity with InDegree. We have that $\|\boldsymbol{a}\|_2\|\boldsymbol{h}\|_2 = np$, the average degree of the graph. Therefore, Assumption 4.5 is satisfied when $p = \omega(\frac{1}{\sqrt{n}})$. It is straightforward to see that in that case Assumption 4.10 is also satisfied. Therefore, we have the following theorem.

Theorem 5.1. *For $p = \omega(\frac{1}{\sqrt{n}})$, the HITS algorithm is (L_2, d_2)-stable with high probability on the on the class $\mathcal{G}(n,p)$. Furthermore, the HITS and InDegree algorithms are (L_2, d_2)-similar with high probability.*

5.2. Power Law Graphs

A discrete random variable X follows a power law distribution with parameter α if $Pr[X = x] \propto x^{-\alpha}$. For variable X it holds that the cumulative distribution follows a power law with exponent $\alpha - 1$, that is, $Pr[X \ge x] \propto x^{-\alpha+1}$ [Newman 03]. Closely related to the power law distribution is the Zipfian distribution, also known as Zipf's law [Zipf 49]. Zipf's law states that the rth largest value of the random variable X is proportional to $r^{-\beta}$. It can be proved [Adamic and Huberman 02] that if X follows a Zipfian distribution with exponent β, then it also follows a power law distribution with parameter $\alpha = 1 + 1/\beta$.

We will now prove that Assumptions 4.5 and 4.10 are general enough to include the case that the authority values follow a power law distribution with exponent $\alpha \ge 3$. Therefore, in the resulting graphs the *expected* in-degrees follow a power law distribution with exponent also $\alpha \ge 3$.

First, we set the hub values such that $h_i = \Theta(1)$, for all $1 \le i \le n$. Therefore, we have that $H = \Theta(n)$ and $\|\boldsymbol{h}\|_2 = \Theta(\sqrt{n})$. Having $H = \Theta(n)$ guarantees that $H\|\boldsymbol{a}\|_2 = \omega(\sqrt{n}\log n)$ for any $\boldsymbol{a}$ such that $\|\boldsymbol{a}\|_2 = \Omega(1)$, thus satisfying

Assumption 4.5. Furthermore, since $\|\boldsymbol{h}\|_2 = \Theta(\sqrt{n})$, we need that $\|\boldsymbol{a}\|_2 = \omega(1)$ in order to satisfy Assumption 4.10. Therefore, our objective becomes to set the authority values a_i such that they follow a power law and $\|\boldsymbol{a}\|_2 = \omega(1)$.

We consider two possible ways of setting the authority values. In the first case, we select the authority values so that they follow Zipf's law, with exponent β. Without loss of generality, we assume that $a_1 \geq a_2 \geq \cdots \geq a_n$. For some constant $c \leq 1$ the ith authority value is defined as $a_i = ci^{-\beta}$, for $\beta > 0$. We have that $\|\boldsymbol{a}\|_2^2 = \sum_{i=1}^{n} \frac{c}{i^{2\beta}}$. This sum converges to a constant for $\beta > 1/2$, while for $\beta \leq 1/2$ we have that $\|\boldsymbol{a}\|_2^2 = \Omega(\log n)$. This implies a power law distribution on the authority values with exponent $\alpha \geq 3$.

A different way of obtaining a power law distribution on the authority values with exponent α is by making the cumulative distribution follow a power law with exponent $\alpha - 1$. We accomplish this as follows. Assume that $n = m^{\alpha-1}$ for some $m > 0$. We then generate numbers $x_1, \cdots, x_n$ such that $n/k^{\alpha-1}$ take value at least k. Therefore, the fraction of x_i's that take value at least k is $1/k^{\alpha-1}$, and the x_i values are power law distributed with exponent α. The maximum value is m.

We now define $a_i = x_i/m$. We are interested in finding the values of α for which $\|\boldsymbol{a}\|_2 = \sum_{i=1}^{n} a_i^2 = \omega(1)$. Let N_k denote the number of x_i values that are equal to k. For all $1 \leq k \leq m-1$, we have that

$$N_k = \frac{n}{k^{\alpha-1}} - \frac{n}{(k+1)^{\alpha-1}}$$

and $N_m = 1$. Therefore,

$$\begin{aligned}
\|\boldsymbol{a}\|_2^2 &= \frac{1}{m^2}\sum_{i=1}^{n} x_i^2 = \frac{1}{m^2}\sum_{k=1}^{m} N_k x_k^2 \\
&= \frac{1}{m^2}\left(\left(n - \frac{n}{2^{\alpha-1}}\right)1^2 + \left(\frac{n}{2^{\alpha-1}} - \frac{n}{3^{\alpha-1}}\right)2^2 + \cdots \right. \\
&\quad \left. + \left(\frac{n}{k^{\alpha-1}} - \frac{n}{(k+1)^{\alpha-1}}\right)k^2 + \cdots + \frac{n}{m^{a-1}}m^2\right) \\
&= \frac{n}{m^2}\sum_{k=1}^{m} \frac{1}{k^{\alpha-1}}\left((k+1)^2 - k^2\right) = \frac{n}{m^2}\sum_{k=1}^{m}\frac{2k+1}{k^{\alpha-1}} \\
&= m^{\alpha-3}\Theta\left(\sum_{k=1}^{m}\frac{1}{k^{\alpha-2}}\right),
\end{aligned}$$

where in the last equality we replaced n with $m^{\alpha-1}$.

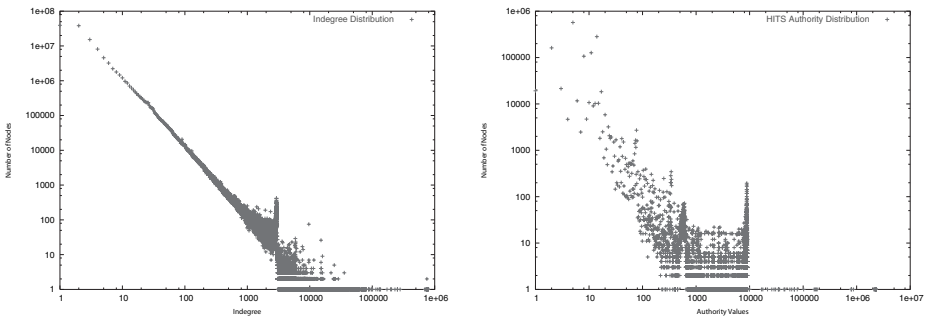
For $\alpha = 3$ we have that $\|\boldsymbol{a}\|_2^2 = \Theta(\log m) = \Theta(\log n)$, while for $\alpha > 3$ we have that $\|\boldsymbol{a}\|_2^2 = \Theta(m^{\alpha-3}) = \Theta(n^{\delta})$ for $\delta = \frac{\alpha-3}{\alpha-1}$; therefore, our requirement holds. For $\alpha < 3$ the sum

$$S_m(\alpha) = \sum_{k=1}^{m} \frac{1}{k^{\alpha-2}}$$

can be shown [Apostol 76] to be $S_m(\alpha) = \Theta(m^{3-\alpha})$. Therefore, $\|\boldsymbol{a}\|_2 = \Theta(1)$.

It appears that when setting the authority values to follow an exact power law distribution, we can satisfy Assumption 4.5 only if $\alpha \geq 3$.[2] This is rather unfortunate since the exponent for the web graph distribution is estimated to be around 2.1, and in most real-life networks $2 < \alpha < 3$. One possible way to enforce Assumption 4.5, while having a distribution that is *almost* a power law distribution, is to explicitly set $\omega(1)$ number of authority values to be $\Theta(1)$, resulting in a power law with a "fatter" tail. If $\omega(n^{1/2})$ authority and hub values are set to be $\Theta(1)$, then we can have the hub values also follow a power law distribution.

6. Experimental Analysis



(a) InDegree (b) HITS

Figure 1. InDegree and HITS distributions on the web graph.

In this section we study experimentally the similarity of HITS and InDegree on a large sample of the World Wide Web. We analyze a sample of 136 million vertices and about 1.2 billion edges of the web graph collected in 2001 by the WebBase project[3] at Stanford. Figures 1(a) and 1(b) show the distributions of the InDegree and HITS authority values. The in-degree distribution, as it is well known, follows a power law distribution. The HITS authority weights also

[2]In the preliminary version of the paper, we incorrectly claim that the assumptions are satisfied when $\alpha \geq 2$.

[3]http://www-diglib.stanford.edu/$\sim$testbed/doc2/WebBase/

	authority/in-degree	hub/out-degree
d_2 distance	0.36	1.23
correlation coefficient	0.93	0.005

Table 1. Similarity between HITS and InDegree.

follow a "fat" power law distribution in the central part of the plot. Table 1 summarizes our findings on the relationship between InDegree and HITS. Since we only have a single graph and not a sequence of graphs, the distance measures are not very informative, so we also compute the correlation coefficient between the two weight vectors. We observe a strong correlation between the authority weights of HITS and the in-degrees and almost no correlation between the hub weights and the out-degrees. Similar trends are observed for the d_2 distance, where the distance between hub weights and out-degrees is much larger than that between authority weights and in-degrees.

In order to better understand the high correlation value between the authority weights and the in-degrees, we looked at how the two sequences correlate when we look at different percentiles of the authority weights. For most ranking applications the interest is in the top part of the ranking. We would like the correlation between the two algorithms to be due to agreement on the high ranked nodes rather than on the tail of the distribution. We thus removed the nodes that correspond to the top 10% of the authority values and computed again the correlation coefficient for the remaining nodes. We observed that the correlation coefficient dropped from 0.93 to 0.06, indicating that there is very little correlation in the tail of distribution. We then zoomed in on the top 10% of the authority values,

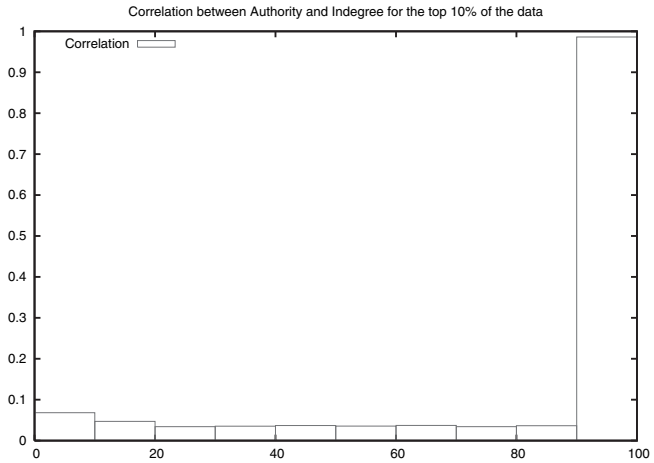


Figure 2. Correlation for different percentiles of the top 10% of the HITS values with the InDegree values.

and we computed the correlation coefficient for different percentiles of the data. The results are shown in Figure 2. From the plot we see that the correlation coefficient is low, even when considering 90% of the top values, and it grows close to 1 only when the top 10% of the top values is included. Given that both the authority and the in-degree values follow a power law distribution, we can thus conclude that the strong correlation between the HITS and InDegree is due to a few nodes, that both HITS and InDegree rank highly, and to which nodes they both assign very high weights. These nodes dominate the computation of the correlation coefficient.

In conclusion, although the World Wide Web, as expected, is not a rank-one matrix, there is strong correlation between HITS and InDegree that is due to their agreement for the top of the ranking. For these nodes, the HITS authority weights can be well approximated by the in-degrees.

7. Similarity of HITS and InDegree

In this section we study the general conditions under which the HITS and InDegree algorithms are similar. Consider a graph $G \in \mathcal{G}_n$ and the corresponding adjacency matrix W. Let $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_n$ be the singular values of W, and let $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n$ and $\boldsymbol{h}_1, \ldots, \boldsymbol{h}_n$ denote the right (authority) and left (hub) singular vectors, respectively. All vectors are unit vectors in the L_2 norm. The HITS algorithm outputs the vector $\boldsymbol{a} = \boldsymbol{a}_1$. Let $\boldsymbol{w}$ denote the output of the InDegree algorithm (normalized in L_2). Also, let $H_i = \sum_{j=1}^{n} h_i(j)$ be the sum of the entries of the ith hub vector. We can prove the following proposition.

Proposition 7.1. *For a graph $G \in \overline{\mathcal{G}}_n$, the d_2 distance between HITS and InDegree is*

$$d_2(\boldsymbol{a}, \boldsymbol{w}) = \sqrt{\left(\frac{\sigma_2 H_2}{\sigma_1 H_1}\right)^2 + \cdots + \left(\frac{\sigma_n H_n}{\sigma_1 H_1}\right)^2}. \tag{7.1}$$

Proof. The adjacency matrix W of graph G can be decomposed as $W = \sigma_1 \boldsymbol{h}_1 \boldsymbol{a}_1^T + \cdots + \sigma_n \boldsymbol{h}_n \boldsymbol{a}_n^T$. Let $\boldsymbol{d}$ denote the vector such that the ith entry $d(i)$ of this vector is the in-degree of node i (not normalized). We have that $d(i) = \sigma_1 H_1 \boldsymbol{a}_1(i) + \cdots + \sigma_n H_n \boldsymbol{a}_n(i)$ and $\boldsymbol{d} = \sigma_1 H_1 \boldsymbol{a}_1 + \cdots + \sigma_n H_n \boldsymbol{a}_n$. Note that

$$\begin{aligned} \|\boldsymbol{d}\|^2 &= (\sigma_1 H_1 \boldsymbol{a}_1 + \cdots + \sigma_n H_n \boldsymbol{a}_n)^T (\sigma_1 H_1 \boldsymbol{a}_1 + \cdots + \sigma_n H_n \boldsymbol{a}_n) \\ &= \sigma_1^2 H_1^2 + \cdots + \sigma_n^2 H_n^2 \geq \sigma_1^2 H_1^2, \end{aligned}$$

where the last equation follows from the fact that $\boldsymbol{a}_i^T \boldsymbol{a}_i = 1$ and $\boldsymbol{a}_i^T \boldsymbol{a}_j = 0$.

The output of InDegree is $\boldsymbol{w} = \boldsymbol{d}/\|\boldsymbol{d}\|$, and the output of HITS is $\boldsymbol{a} = \boldsymbol{a}_1$. We are interested in bounding $\|\boldsymbol{a} - \gamma\boldsymbol{w}\|$, where $\gamma = \|\boldsymbol{d}\|/\sigma_1 H_1 \geq 1$. We have that

$$\begin{aligned}
\|\boldsymbol{a} - \gamma\boldsymbol{w}\|^2 &= \left\| \frac{\sigma_2 H_2}{\sigma_1 H_1}\boldsymbol{a}_2 + \cdots + \frac{\sigma_n H_n}{\sigma_1 H_1}\boldsymbol{a}_n \right\|^2 \\
&= \left(\frac{\sigma_2 H_2}{\sigma_1 H_1}\boldsymbol{a}_2 + \cdots + \frac{\sigma_n H_n}{\sigma_1 H_1}\boldsymbol{a}_n \right)^T \cdot \left(\frac{\sigma_2 H_2}{\sigma_1 H_1}\boldsymbol{a}_2 + \cdots + \frac{\sigma_n H_n}{\sigma_1 H_1}\boldsymbol{a}_n \right) \\
&= \left(\frac{\sigma_2 H_2}{\sigma_1 H_1} \right)^2 + \cdots + \left(\frac{\sigma_n H_n}{\sigma_1 H_1} \right)^2.
\end{aligned}$$

Therefore,

$$d_2(\boldsymbol{a}, \boldsymbol{w}) = \sqrt{\left(\frac{\sigma_2 H_2}{\sigma_1 H_1} \right)^2 + \cdots + \left(\frac{\sigma_n H_n}{\sigma_1 H_1} \right)^2}.$$

□

We now study the conditions under which $d_2(\boldsymbol{a}, \boldsymbol{w}) = o(1)$. Since the values of $\boldsymbol{h}_1$ are positive, we have that $H_1 = \|\boldsymbol{h}_1\|_1$ and $1 \leq H_1 \leq \sqrt{n}$. For every $i > 1$, we have that $|H_i| \leq \|\boldsymbol{h}_i\|_1$ and $|H_i| \leq \sqrt{n}$. Any of the following conditions guarantees the similarity of HITS and InDegree:

1. $\sigma_2/\sigma_1 = o(1/\sqrt{n})$ and there exists a constant k such that $\sigma_{k+1}/\sigma_1 = o(1/n)$;
2. $H_1 = \Theta(\sqrt{n})$, $\sigma_2/\sigma_1 = o(1)$, and there exists a constant k such that $\sigma_{k+1}/\sigma_1 = o(1/n)$;
3. $H_1 = \Theta(\sqrt{n})$ and $\sigma_2/\sigma_1 = o(1/\sqrt{n})$.

Assume now that $|H_i|/(\sigma_1 H_1) = o(1)$, for all $i \geq 2$. One possible way to obtain this bound is to assume that $\sigma_1 = \omega(\sqrt{n})$, or that $H_1 = \Theta(\sqrt{n})$ and $\sigma_1 = \omega(1)$. Then, we can obtain the following characterization of the distance between HITS and InDegree. From Equation (7.1) we have that $d_2(\boldsymbol{a}, \boldsymbol{w}) = o\left(\sqrt{\sigma_2^2 + \cdots + \sigma_n^2}\right)$. Let $W_1 = \sigma_1 \boldsymbol{h}_1 \boldsymbol{a}_1^T$ denote the rank-one approximation of W. The matrix $D = W - W_1$ is called the residual matrix, and it has singular values $\sigma_2, \ldots, \sigma_n$. We have that

$$d_2(\boldsymbol{a}, \boldsymbol{w}) = o\left(\|W - W_1\|_F\right) \qquad \text{and} \qquad d_2(\boldsymbol{a}, \boldsymbol{w}) = o\left(\sqrt{\|W\|_F^2 - \|W\|_2^2}\right). \tag{7.2}$$

Equation (7.2) says that the similarity of the HITS and InDegree algorithms depends on the Frobenius norm of the residual matrix. Furthermore, the similarity of the HITS and InDegree algorithms depends on the difference between the Frobenius and the spectral (L_2) norm of matrix W. The L_2 norm measures the

strength of the strongest linear trend in the matrix, while the Frobenius norm captures the sum of the strengths of all linear trends in the matrix [Achlioptas and McSherry 01]. The similarity of the HITS and InDegree algorithms depends upon the contribution of the strongest linear trend to the sum of linear trends.

8. Conclusions

Our work opens a number of interesting directions for future work. First, it would be interesting to determine a necessary condition for the stability of the HITS algorithm, that is, the converse of Theorem 4.4. Ng, Zheng, and Jordan show that if the gap between the singular values is small then there is a perturbation matrix with small norm that can cause a large perturbation on the singular vectors [Ng et al. 01a]. However, this perturbation does not necessarily produce an adjacency matrix. It is not clear how to modify the proof to work for perturbations that transform graphs.

It would be interesting to study the stability and similarity of other LAR algorithms on product graphs, such as the PageRank and the SALSA algorithms. Finally, it would be interesting to study other classes of random graphs [Barabasi and Albert 99, Kumar et al. 00a].

In this paper we studied the behavior of the HITS algorithm on the class of product graphs. We proved that under some assumptions the HITS algorithm is stable, and it is similar to the InDegree algorithm. Our assumptions include graphs with expected degrees that follow a power law distribution.

It would also be interesting to find a condition that characterizes the family of graphs on which the PageRank algorithm is stable. The work on stability of PageRank so far has shown that the perturbation on the PageRank values is bounded by the weights of the nodes whose out-links are perturbed. Thus, the existence of a node with PageRank value in $O(1)$ is a necessary condition for the instability of PageRank. However, this does not provide a characterization of the graphs on which PageRank is stable.

For the product graph model, it would be interesting to examine if it is possible to weaken Assumptions 4.5 and 4.10. It would also be interesting to study whether the results in Section 5.2 can be extended for power law distributions with exponent less than 3, or show that this is not possible.

Furthermore, for the proof of similarity between HITS and InDegree, we used the fact that both algorithms produce the latent authority values on a rank-one matrix. There are other algorithm that on a rank-one matrix also reconstruct the vector $\boldsymbol{a}$. The PageRank algorithm, the SALSA algorithm, and the HubAvg algorithm [Borodin et al. 05] are such algorithms. Is it possible to prove or disprove that these algorithms are similar to the InDegree algorithm? This would be especially interesting for the case of the PageRank algorithm.

It would also be interesting to study the *rank similarity* between the algorithms on the class of product graphs. This is likely more difficult, since linear algebra and perturbation theory cannot help us for this task.

Finally, it would be interesting to study theoretically the questions of stability and similarity for the classes of random graphs defined by other generative models [Barabasi and Albert 99, Kumar et al. 00a].

9. Appendix: Proof of Lemma 4.1

We use results from perturbation theory [Stewart and Sun 90] to study how the principal singular vectors of a matrix W change when we add the matrix E. The theorems that we use assume that both the matrix W and the perturbation E are symmetric, so instead of using the matrices W and E, we will consider the matrices B and F defined as follows:

$$B = \begin{bmatrix} 0 & W^T \\ W & 0 \end{bmatrix} \qquad \text{and} \qquad F = \begin{bmatrix} 0 & E^T \\ E & 0 \end{bmatrix}. \tag{9.1}$$

If σ_i is the ith singular value of W, and $(\boldsymbol{u}_i, \boldsymbol{v}_i)$ is the corresponding pair of singular vectors, then the matrix B has eigenvalues $\pm\sigma_i$, with eigenvectors $[\boldsymbol{v}_i, \boldsymbol{u}_i]^T$ for the eigenvalue σ_i, and $[\boldsymbol{v}_i, -\boldsymbol{u}_i]^T$ for the eigenvalue $-\sigma_i$. Therefore, instead of studying the perturbation of the singular values and vectors of matrix $W + E$, we will study the eigenvalues and eigenvectors of matrix $B + F$. Note also that $\|F\|_2 = \|E\|_2$ and that $\|F\|_F = \sqrt{2}\|E\|_F$.

We make use of the following theorem by Stewart (Theorem V.2.8 in [Stewart and Sun 90] for the symmetric case).

Theorem 9.1. *Suppose that B and $B + F$ are n by n symmetric matrices and that*

$$Q = [\boldsymbol{q}, Q_2]$$

is a unitary matrix, such that the vector $\boldsymbol{q}$ is an eigenvector for the matrix B. Partition the matrices $Q^T BQ$ and $Q^T FQ$ as follows:

$$Q^T BQ = \begin{bmatrix} \lambda & 0 \\ 0 & B_{22} \end{bmatrix} \quad \text{and} \quad Q^T FQ = \begin{bmatrix} f_{11} & \boldsymbol{f}_{21}^T \\ \boldsymbol{f}_{21} & F_{22} \end{bmatrix}.$$

Let

$$\delta = \min_{\mu \in \lambda(B_{22})} |\lambda - \mu| - |f_{11}| - \|F_{22}\|_2,$$

where $\lambda(B_{22})$ denotes the set of eigenvalues of B_{22}. If $\delta > 0$ and $\delta > 2\|\boldsymbol{f}_{21}\|_2$, then there exists a vector $\boldsymbol{p}$ such that

$$\|\boldsymbol{p}\|_2 < 2\frac{\|\boldsymbol{f}_{21}\|_2}{\delta}$$

and

$$\boldsymbol{q}' = \boldsymbol{q} + Q_2\boldsymbol{p}$$

is an eigenvector of the matrix $B + F$. For the eigenvalue λ' that corresponds to the eigenvector $\boldsymbol{q}'$, we have that

$$\lambda' = \lambda + f_{11} + \boldsymbol{f}_{21}^T\boldsymbol{p}.$$

We now give the proof of Lemma 4.1.

Proof. In the following, we will argue that under condition $\sigma_1 - \sigma_2 = \omega(\|E\|_2)$, perturbing matrix W by E causes only a small perturbation of the principal left and right singular vectors of W. Moreover, we will prove that the perturbed singular vectors remain the principal singular vectors of W since the perturbation does not change the relative order of the first and the second singular values.

In Theorem 9.1, define matrices B and F as in the Equation (9.1). Now, set $\boldsymbol{q} = [\boldsymbol{u}, \boldsymbol{v}]^T$, where $\boldsymbol{u}$ and $\boldsymbol{v}$ are the left and right singular vectors of W, respectively. We have that $\lambda = \sigma_1$. We have that

$$\delta = \sigma_1 - \sigma_2 - |f_{11}| - \|F_{22}\|_2.$$

Note that $f_{11} = \boldsymbol{q}^T F\boldsymbol{q}$, $F_{22} = Q_2^T F Q_2$, and $\boldsymbol{f}_{21} = Q_2^T F\boldsymbol{q}$. Since $\|AB\|_2 \leq \|A\|_2\|B\|_2$ and unitary matrices have L_2 norm 1, we have that $|f_{11}| \leq \|F\|_2$, $\|F_{22}\|_2 \leq \|F\|_2$, and $\|\boldsymbol{f}_{21}\|_2 \leq \|F\|_2$.

Note that $\|F\|_2 = \|E\|_2$. If $\sigma_1 - \sigma_2 = \omega(\|E\|_2)$, then $\delta = \omega(\|E\|_2)$ and obviously $\delta > 0$ and $\delta > 2\|\boldsymbol{f}_{21}\|_2$. Therefore, there exists a vector $\boldsymbol{p}$ with $\|\boldsymbol{p}\|_2 < \|\boldsymbol{f}_{21}\|_2/\delta$, such that the vector

$$\boldsymbol{q}' = \boldsymbol{q} + Q_2\boldsymbol{p}$$

is an eigenvector of the matrix $B + F$. We also have that $\|\boldsymbol{p}\| = o(1)$ since $\|\boldsymbol{f}_{21}\| \leq \|E\|_2$ and $\delta = \omega(\|E\|_2)$.

The eigenvalue associated with the vector $\boldsymbol{q}'$ is $\lambda' = \lambda + f_{11} + \boldsymbol{f}_{21}^T\boldsymbol{p}$. Therefore,

$$\begin{aligned} |\lambda - \lambda'| &= |f_{11} + \boldsymbol{f}_{21}^T\boldsymbol{p}| \leq |f_{11}| + \|\boldsymbol{f}_{21}^T\|_2\|\boldsymbol{p}\|_2 \\ &\leq \|E\|_2 + o(\|E\|_2) = O(\|E\|_2). \end{aligned}$$

The first and second inequalities follow from the well-known property of the absolute value and the properties of the L_2 vector norm. The last inequality follows from the fact that $\|\boldsymbol{f}_{21}^T\|_2 = O(\|E\|_2)$ and $\|\boldsymbol{p}\|_2 = o(1)$.

Note that $\lambda = \sigma_1$ is the principal singular value of the matrix W. Let σ_i' denote the ith singular value of the matrix $W' = W + E$. We know that for any singular value σ_i, $|\sigma_i - \sigma_i'| \leq \|E\|_2$. We have that $|\sigma_1 - \sigma_1'| \leq \|E\|_2$ and $|\sigma_2 - \sigma_2'| \leq \|E\|_2$. We have assumed that $\sigma_1 - \sigma_2 = \omega(\|E\|_2)$. Therefore, it

must be that $\sigma_1' - \sigma_2' = \omega(\|E\|_2)$. Since $|\lambda - \lambda'| = O(\|E\|_2)$, it follows that $\lambda' = \sigma_1'$. Thus, the vector $\boldsymbol{q}'$ is the principal eigenvector of the matrix $B + F$, and $\boldsymbol{q}' = [\boldsymbol{u}', \boldsymbol{v}']^T$, where $\boldsymbol{u}'$ and $\boldsymbol{v}'$ are the left and right singular vectors of W'. Since $\|Q_2\boldsymbol{p}\|_2 \leq \|\boldsymbol{p}\|_2$, it follows that $\|\boldsymbol{q} - \boldsymbol{q}'\|_2 = o(1)$. Therefore,

$$\|\boldsymbol{v}' - \boldsymbol{v}\|_2 = o(1) \quad \text{and} \quad \|\boldsymbol{u}' - \boldsymbol{u}\|_2 = o(1).$$

□

Acknowledgments. A preliminary version of this paper has appeared in ICALP 2005. This work was partially supported by the EU under contract 001907 (DELIS) and 33555 (COSIN) and by the Italian MIUR under contract ALGO-NEXT.

References

[Achlioptas and McSherry 01] D. Achlioptas and F. McSherry. "Fast Computation of Low Rank Matrix Approximations." In *Proceedings of the Thiry-Third Annual ACM Symposium on Theory of Computing*, pp. 611–618. New York: ACM Press, 2001.

[Achlioptas et al. 01] D. Achlioptas, A. Fiat, A. Karlin, and F. McSherry. "Web Search through Hub Synthesis." In *Proceedings of the 42nd IEEE Symposium on Foundation of Computer Science*, pp. 500–509. Los Alamitos, CA: IEEE Press, 2001.

[Adamic and Huberman 02] L. A. Adamic and B. A. Huberman. "Zipf's Law and the Internet." *Glottometrics* 3 (2002), 143–150.

[Aiello et al. 00] W. Aiello, F. R. K. Chung, and L. Lu. "A Random Graph Model for Massive Graphs." In *Proceedings of the Thirty-Second Annual ACM Symposium on Theory of Computing*, pp. 171–180. New York: ACM Press, 2000.

[Aiello et al. 01] William Aiello, Fan R. K. Chung, and Linyuan Lu. "Random Evolution in Massive Graphs." In *Proceedings of the 42nd IEEE Symposium on Foundations of Computer Science*, pp. 510–519. Los Alamitos, CA: IEEE Press, 2001.

[Apostol 76] Tom M. Apostol. *Introduction to Analytic Number Theory.* Berlin: Springer-Verlag, 1976.

[Azar et al. 01] Y. Azar, A. Fiat, A. Karlin, F. McSherry, and J. Saia. "Spectral Analysis of Data." In *Proceedings of the Thirty-Third Annual ACM Symposium on Theory of Computing*, pp. 619–626. New York: ACM Press, 2001.

[Barabasi and Albert 99] A.-L. Barabasi and R. Albert. "Emergence of Scaling in Random Networks." *Science* 286 (1999), 509–512.

[Bhalotia et al. 02] G. Bhalotia, C. Nakhe, A. Hulgeri, S. Chakrabarti, and S. Sudarshan. "Keyword Searching and Browsing in Databases Using BANKS." In *Proceedings of the 18th International Conference on Data Engineering*, pp. 431–440. Los Alamitos, CA: IEEE Press, 2002.

[Bharat and Henzinger 98] K. Bharat and M. R. Henzinger. "Improved Algorithms for Topic Distillation in a Hyperlinked Environment." In *Proceedings of the 21st Annual International ACM SIGIR Confrence on Research and Development in Information Retrieval*, pp. 104–111. New York: ACM Press, 1998.

[Bianchini et al. 02] M. Bianchini, M. Gori, and F. Scarselli. "PageRank: A Circuital Analysis." In *Eleventh International World Wide Web Conference Posters*. Available at http://www2002.org/CDROM/poster/165.pdf, 2002.

[Borodin et al. 01] 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 Conference on World Wide Web*, pp. 415–429. New York: ACM Press, 2001.

[Borodin et al. 05] A. Borodin, G. O. Roberts, J. S. Rosenthal, and P. Tsaparas. "Link Analysis Ranking: Algorithms, Experiments, and Theory." *ACM Transactions on Internet Technology* 5:1 (2005), 231–297.

[Brin and Page 98] S. Brin and L. Page. "The Anatomy of a Large-Scale Hypertextual Web Search Engine." *Proceedings of WWW7, Computer Networks* 30:1–7 (1998), 107–117.

[Broder et al. 00] A. Broder, R. Kumar, F. Maghoul, P. Raghavan, S. Rajagopalan, R. Stata, A. Tomikns, and Wiener Wiener. "Graph Structure in the Web." In *Proceedings of the Ninth Internation World Wide Web Conference*. Available at http://www9.org/w9cdrom/index.html, 2000.

[Chung and Lu 02] F. Chung and L. Lu. "Connected Components in Random Graphs with Given Degree Sequences." *Annals of Combinatorics* 6 (2002), 125–145.

[Chung and Lu 03] F. Chung and L. Lu. "The Average Distances in Random Graphs with Given Expected Degrees." *Internet Mathematics* 1:1 (2003), 91–114.

[Chung et al. 03] F. Chung, L. Lu, and V. Vu. "Eigenvalues of Random Power Law Graphs." *Annals of Combinatorics* 7 (2003), 21–33.

[Chung et al. 04] F. Chung, L. Lu, and V. Vu. "The Spectra of Random Graphs with Given Expected Degrees." *Internet Mathematics* 1:3 (2004), 257–275.

[Dezso and Barabasi 02] Z. Dezso and A.-L. Barabasi. "Halting Viruses in Scale-Free Networks." *Phys. Rev. E* 65 (2002), 055103.

[Erdős and Rèny 60] P. Erdös and A. Rènyi. "On the Evolution of Random Graphs." *Publ. Math. Inst. Hung. Acad. Sci.* 5 (1960), 17–61.

[Fagin et al. 03] R. Fagin, R. Kumar, and D. Sivakumar. "Comparing Top k Lists." In *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 28–36. Philadelphia: SIAM, 2003.

[Faloutsos et al. 99] Michalis Faloutsos, Petros Faloutsos, and Christos Faloutsos. "On Power-Law Relationships of the Internet Topology." In *Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, pp. 251–262. New York: ACM Press, 1999.

[Haveliwala 02] T. H. Haveliwala. "Topic-Sensitive PageRank." In *Proceedings of the 11th International Conference on World Wide Web*, pp. 517–526. New York: ACM Press, 2002.

[Kleinberg 98] J. Kleinberg. "Authoritative Sources in a Hyperlinked Environment." In *Proceedings of the Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 668–677. Philadelphia: SIAM, 1998.

[Kleinberg et al. 99] J. M. Kleinberg, R. Kumar, P. Raghavan, R. Rajagopalan, and A. Tomkins. "The Web as a Graph: Measurements, Models, and Methods." In *Computing and Combinatorics: 5th Annual International Conference, COCOON'99, Tokyo, Japan, July 26–28, 1999, Proceedings*, Lecture Notes in Computer Science 1627, pp. 1–17. Berlin: Springer, 1999.

[Kumar et al. 00a] R. Kumar, P. Raghavan, S. Rajagopalan, D. Sivakumar, A. Tomkins, and E. Upfal. "Stochastic Models for the Web Graph." In *Proceedings of the 41st Annual Symposium on Foundations of Computer Science*, pp. 57–65. Los Alamitos, CA: IEEE Press, 2000.

[Kumar et al. 00b] S. Ravi Kumar, P. Raghavan, S. Rajagopalan, D. Sivakumar, Andrew Tomkins, and Upfal Upfal. "The Web as a Graph." In *Proceedings of the Nineteenth ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, pp. 1–10. New York: ACM Press, 2000.

[Lee and Borodin 03] H. C. Lee and A. Borodin. "Perturbation of the Hyperlinked Environment." In *Computing and Combinatorics: 9th Annual International Conference, COCOON 2003, Big Sky, MT, USA, July 25–28, 2003, Proceedings*, Lecture Notes in Computer Science 2697, pp. 271–283. Berlin: Springer, 2003.

[Lempel and Moran 00] R. Lempel and S. Moran. "The Stochastic Approach for Link-Structure Analysis (SALSA) and the TKC Effect." In *Proceedings of the Ninth International World Wide Web Conference*. Available at http://www9.org/w9cdrom/index.html, 2000.

[Lempel and Moran 03] R. Lempel and S. Moran. "Rank Stability and Rank Similarity of Link-Based Web Ranking Algorithms in Authority Connected Graphs." Paper presented at the Second Workshop on Algorithms and Models for the Web-Graph (WAW 2003), Budapest, Hungary, May 20–24, 2003.

[Mihail and Papadimitriou 02] Milena Mihail and Christos H. Papadimitriou. "On the Eigenvalue Power Law." In *Randomization and Approximation Techniques in Computer Science: 6th International Workshop, RANDOM 2002, Cambridge, MA, USA, September 13–15, 2002, Proceedings*, Lecture Notes in Computer Science 2483, pp. 254–262. Berlin: Springer, 2002.

[Motwani and Raghavan 95] R. Motwani and P. Raghavan. *Randomized Algorithms.* Cambridge, UK: Cambridge University Press, 1995.

[Newman 01] M. E. J. Newman. "Who is the Best Connected Scientist? A Study of Scientific Coauthorship Networks." *Physical Review E* 64 (2001), 016132.

[Newman 03] M. E. J. Newman. "The Structure and Function of Complex Networks." *SIAM Reviews* 45:2 (2003), 167–256.

[Ng et al. 01a] A. Y. Ng, A. X. Zheng, and M. I. Jordan. "Link Analysis, Eigenvectors, and Stability." In *Proceedings of the Seventeeth International Joint Conference on Artificial Intelligence*, pp. 903–910. San Francisco: Morgan Kaufmann, 2001.

[Ng et al. 01b] A. Y. Ng, A. X. Zheng, and M. I. Jordan. "Stable Algorithms for Link Analysis." In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 258–266. New York: ACM Press, 2001.

[Pandurangan et al. 02] G. Pandurangan, P. Raghavan, and E. Upfal. "Using PageRank to Characterize Web Structure." In *Computing and Combinatorics: 8th Annual International Conference, COCOON 2002, Singapore, August 15–17, 2002, Proceedings*, Lecture Notes in Computer Science 2387, pp. 330–339. Berlin: Springer, 2002.

[Pennock et al. 02] D.M. Pennock, G.W. Flake, S. Lawrence, E.J. Glover, and C.L. Giles. "Winners Don't Take All: Characterizing the Competition for Links on the Web." *Proc. of the National Academy of Sciences* 99:8 (2002), 5207–5211.

[Rafiei and Mendelzon 00] D. Rafiei and A. Mendelzon. "What Is This Page Known For? Computing Web Page Reputations." In *Proceedings of the Ninth International World Wide Web Conference*. Available at http://www9.org/w9cdrom/index.html, 2000.

[Stewart and Sun 90] G. W. Stewart and J. Sun. *Matrix Perturbation Theory*. Boston: Academic Press, 1990.

[Tomlin 03] J. A. Tomlin. "A New Paradigm for Ranking Pages on the World Wide Web." In *Proceedings of the 12th International Conference on World Wide Web*, pp. 350–355. New York: ACM Press, 2003.

[Tsaparas 04] P. Tsaparas. "Using Non-linear Dynamical Systems for Web Searching and Ranking." In *Proceedings of the Twenty-Third ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, pp. 59–70. New York: ACM Press, 2004.

[Zipf 49] G. K. Zipf. *Human Behavior and the Principle of Least Effort*. Reading, MA: Addison-Wesley, 1949.

Debora Donato, Yahoo! Research Barcelona, Octa 1, 1st Floor, 08003 Barcelona, Spain (debora@yahoo-inc.com)

Stefano Leonardi, Dipartimento di Informatica e Sistemistica, Universita di Roma, "La Sapienza," via Ariosto 25, 00185 Rome, Italy (Stefano.Leonardi@dis.uniroma1.it)

Panayiotis Tsaparas, Microsoft Research Silicon Valley, 1065 La Avenida, Mountain View, CA 94043 (panayiotis.tsaparas@microsoft.com)

Received March 1, 2007; accepted July 5, 2007.