Stability and Similarity of Link Analysis Ranking Algorithms *

Debora Donato, Stefano Leonardi, and Panayiotis Tsaparas

[1] Universita di Roma,“La Sapienza”, donato@dis.uniroma1.it
[2] Universita di Roma,“La Sapienza”, leon@dis.uniroma1.it
[3] University of Helsinki, tsaparas@cs.helsinki.fi

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 Web pages. One of the seminal works in this area is that of Kleinberg [15], 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. [7], we prove that 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, a situation observed in the actual Web graph [9]. 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 [15] and Brin and Page [8] introduced the area of *Link Analysis Ranking*, where hyperlink structures are used to rank the results of search queries. Their work was followed by a plethora of modifications, generalizations and improvements (see [7] and references within). As a result, today there exists a wide range of Link Analysis Ranking (LAR) algorithms, many of which are variations of each other.

The multitude of LAR algorithms creates the need for a formal framework for assessing and comparing their properties. Borodin et al., introduced such a theoretical framework in [7]. In this framework an LAR algorithm is defined as a function from a class of graphs of size n to an n-dimensional real vector that assigns an *authority weight* to each node in the graph. The nodes are ranked in decreasing order of their weights. Borodin et al. [7] define various properties of LAR algorithms. 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. Similarity studies how close the outputs of two algorithms are on the same graph.

* Partially supported by the EU under contract 001907 (DELIS) and 33555 (COSIN), and by the Italian MIUR under contract ALGO-NEXT.

Borodin et al. [7] considered the question of stability and similarity over an unrestricted class of graphs. They studied a variety of algorithms, and they proved that no pair of these algorithms is similar, and almost all algorithms are unstable. It appears that the class of all possible graphs is too broad to allow for positive results. This raises naturally 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 Web, various stochastic models have been proposed for the Web graph [4, 5, 16, 3]. The model we consider, which was proposed by Azar et al. [4], is the following: assume that every node i in the graph comes with two parameters a_i and h_i which 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 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 [13] to include graphs where the *expected* degrees follow specific distributions. This is of particular interest since it is well known [16, 9] that the in-degrees of the nodes in the Web graph follow a power law distribution.

Our contribution. In this paper we study the behavior of the HITS algorithm, proposed by Kleinberg [15], on the class of product graphs. The study of HITS on product graphs was initiated by Azar et al. [4] 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. [4] in the framework of Borodin et al. [7]. We extend the definitions of stability and similarity for classes of random graphs, and we demonstrate the link 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 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 as the one observed on the real Web. 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 real Web. We conclude with a discussion on the conditions that guarantee similarity of HITS and INDEGREE for the class of all possible graphs.

2 Related Work

Link Analysis Ranking Algorithms: Let P be a collection of n Web pages 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 Web page 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 i-th 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 [15] in the seminal paper that introduced the hubs and authorities paradigm. In this framework, every page can be thought of as having a *hub* and an *authority* weight. Let $\boldsymbol{h}$ and $\boldsymbol{a}$ denote the n-dimensional hub and authority weight vectors. Kleinberg proposed an iterative algorithm, termed HITS, for computing the vectors $\boldsymbol{h}$ and $\boldsymbol{a}$; the algorithm is essentially a power method computation of the principle eigenvectors of the matrices WW^T and W^TW respectively. These are the principal *singular vectors* of the matrix W. The HITS algorithm returns the vector $\boldsymbol{a}$, the right singular vector of matrix W. More information about Singular Value Decomposition and HITS can be found in Appendix A.1.

Independently from Kleinberg, Brin and Page developed the celebrated PAGERANK algorithm [8], which outputs the stationary distribution of a random walk on the Web graph. The works of Kleinberg [15] and Brin and Page [8] were followed by numerous modifications and extensions (see [7] and references within). Of particular interest is the SALSA algorithm by Lempel and Moran [18], which performs a random walk that alternates between hubs and authorities.

Theoretical study of LAR algorithms: Borodin et al. [7], in the paper that introduced the theoretical framework for the analysis of LAR algorithms, considered various algorithms, including HITS, SALSA, INDEGREE, and variants of HITS defined in their paper. They proved that, on the class of all possible graphs, no pair of algorithms is similar, and only the INDEGREE algorithm is stable. They also defined the notion of *rank stability* and *rank similarity*, where they considered the ordinal rankings induced by the weight vectors. The same results carry over in this case. Their work was extended by Lempel and Moran [19], and Lee and Borodin [17]. The stability of HITS and PAGERANK has also been studied elsewhere [22, 6].

The product graph model: 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. [4]. The undirected case, where the $h_i = a_i$ and the edges are undirected, has been studied extensively [20, 10–12]. The focus of these works is on the case where the parameters follow a power law distribution, as it is the case with most real-life networks.

3 The theoretical framework

In this section we review the definitions of Borodin et al. [7], 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. [7], 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 $\|\boldsymbol{w}\|_p = \left(\sum_{i=1}^{n} |w_i|^p\right)^{1/p}$.

Distance measures: In order to compare the behavior of different algorithms, or the behavior of the same algorithm on different graphs, Borodin et al. [7] defined various distance measures between authority weight vectors. The distance functions we consider are defined using the L_q norm. The d_q distance between two weight vectors $\boldsymbol{w}_1, \boldsymbol{w}_2$ 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 the $d_2(\boldsymbol{a}, \boldsymbol{b}) = \|\boldsymbol{a} - \boldsymbol{b}\|$, and thus the d_2 distance is a metric. The proof appears in Appendix A.2.

Similarity: Borodin et al. [7] 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 1. *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 2. *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[4], then the transitivity property holds. It is easy to show that if algorithms $\mathcal{A}_1$ and $\mathcal{A}_2$ 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).

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 $d_\ell(G, G') =$

[4] A near metric [14] 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 $\boldsymbol{u}, \boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_k, \boldsymbol{v}$, $d(\boldsymbol{u}, \boldsymbol{v}) \leq c(d(\boldsymbol{u}, \boldsymbol{w}_1) + d(\boldsymbol{w}_1, \boldsymbol{w}_2) + \cdots + d(\boldsymbol{w}_k, \boldsymbol{v}))$.

$|(E \cup E') \setminus (E \cap E')|$ 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. [7] give the following generic definition of stability.

Definition 3. *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 4. *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, 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 $1 - o(1)$.

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. [7]. The proof appears in Appendix A.3.

Lemma 1. *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 $\overline{\mathcal{G}}_n$.*

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 can be thought of as the *latent* hub and authority vectors. 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. The matrix W can be written 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 refer to matrix R as the *rounding* matrix, that rounds the entries of M to 0 or 1. We can think of the matrix W as a perturbation of the matrix $M = \boldsymbol{h}\boldsymbol{a}^T$ by the rounding matrix R. The matrix M is a rank-one matrix. 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 the two algorithms remain similar.

More formally, let LATENT denote the (imaginary) LAR algorithm which, 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. We also review some properties of matrix norms in Appendix A.1.

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. The proof of the lemma appears in Appendix A.4.

Lemma 2. *Let W be a matrix, and let $W + E$ be a perturbation of the matrix. Let $\boldsymbol{u}$ and $\boldsymbol{v}$ 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, 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 $\|\boldsymbol{v}' - \boldsymbol{v}\|_2 = o(1)$.*

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 [1].

Theorem 1. *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 $1 - (m+n)^{-\alpha^2}$,

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

Chernoff bounds: We will make use of standard Chernoff bounds. The following theorem can be found in the textbook of Motwani and Raghavan [21].

Theorem 2. *Let $X_1, X_2, \ldots, X_n$ be independent Poisson trials such that, for $1 \le i \le n$, $\Pr[X_i = 1] = p_i$, where $0 \le p_i \le 1$. Let $X = \sum_{i=1}^n X_i, \mu = E[X] = \sum_{i=1}^n p_i$. Then, for $0 < \delta \le 1$, we have that*

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

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 (Appendix A.5) follows directly from Lemma 2, and the fact that the perturbation matrix E has L_2 norm bounded by a constant.

Theorem 3. *The* HITS *algorithm is (L_2, d_2)-stable on the class of graphs $\mathcal{G}_n^\sigma$.*

Theorem 3 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. [22]. Translating their result in the framework of Borodin et al. [7], 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.

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

Assumption 1 *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 1 places a direct lower bound on the principal singular value of the matrix $M = \boldsymbol{h}\boldsymbol{a}^T$. Also, let $A = \sum_{i=1}^n a_i$, denote the sum of the authority values, and let $H = \sum_{j=1}^n h_j$ 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 $HA \ge \|\boldsymbol{a}\|_2 \|\boldsymbol{h}\|_2$, thus, from Assumption 1, $HA = \omega(\sqrt{n})$. This implies that the graph is not too sparse.

Lemma 3. *The algorithms* HITS *and* LATENT *are (L_2, d_2)-similar with high probability on the class $\mathcal{G}_n^p$, subject to Assumption 1.*

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 $M = \boldsymbol{h}_2^T \|\boldsymbol{h}\|_2 \|\boldsymbol{a}\|_2 \boldsymbol{a}_2$. 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 maximum value and variance bounded by 1. Using Theorem 1, 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 2, we have that $\|\boldsymbol{w}_2 - \boldsymbol{a}_2\|_2 = o(1)$ with probability $1 - o(1)$. □

Assumption 1 guarantees also the stability of HITS on $\mathcal{G}_n^p$. The proof (Appendix A.6) follows from the fact that if $G \in \mathcal{G}_n^p$, then $G \in \mathcal{G}_n^\sigma$, with high probability.

Theorem 4. *The* HITS *algorithm is* (L_2, d_2)*-stable with high probability on the class of graphs* $\mathcal{G}_n^p$*, subject to Assumption 1.*

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 $\boldsymbol{r} = [r_1, \ldots, r_n]^T$. Vector $\boldsymbol{d}$ can be expressed as $\boldsymbol{d} = H\boldsymbol{a} + \boldsymbol{r}$.

We first prove the following auxiliary lemma.

Lemma 4. *For every* $q \in [1, \infty)$*, if* $H\|a\|_q = \omega(n^{1/q} \ln n)$*, 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 $1 - 1/n$. We have assumed that $H\|\boldsymbol{a}\| = \omega(n^{1/q} \ln n)$, so it is sufficient to show that $\|\boldsymbol{r}\| = O(n^{1/q} \ln n)$, or equivalently that for all $1 \leq i \leq n$, $|r_i| = O(\ln n)$ with probability at least $1 - 1/n^2$. Note that $r_i = d_i - Ha_i$, so essentially we need to bound the deviation of d_i from its expectation.

We 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, node 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 Theorem 2, Equation 2, 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 2, 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. The details of the proof appear in Appendix A.7.

Lemma 5. *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 $H\|\boldsymbol{a}\|_q = \omega(n^{1/q}\ln n)$.*

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

Assumption 2 *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 2 implies that the expected number of edges in the graph satisfies $HA = \omega(\sqrt{n}\ln n)$. Note that we can satisfy Assumption 2 by requiring $HA = \omega(n\ln n)$, that is, the graph is dense enough. We can satisfy both Assumption 1 and 2 by requiring that $\sigma_1(M) = \|\boldsymbol{h}\|_2\|\boldsymbol{a}\|_2 = \omega(\sqrt{n}\ln n)$.

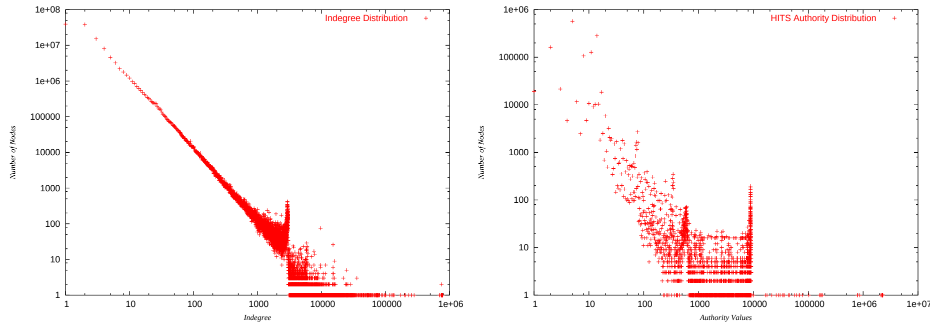
The INDEGREE and LATENT algorithms are (L_2, d_2)-similar subject to Assumption 2. The following theorem follows from the transitivity property of similarity.

Theorem 5. *The* HITS *and* INDEGREE *algorithms are (L_2, d_2)-similar with high probability on the class $\mathcal{G}_n^p$, subject to Assumptions 1 and 2.*

4.5 Power law graphs

A discrete random variable X follows a power law distribution with parameter α, if $Pr[X = x] \propto x^{-\alpha}$. Closely related to the power-law distribution is the Zipfian distribution, also known as Zipf's law [24]. Zipf's law states that the r-th largest value of the random variable X is proportional to $r^{-\beta}$. It can be proved [2] 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 1 and 2 are general enough to include graphs with *expected* in-degrees that follow Zipf's law with parameter $\beta < 1$.

Without loss of generality we assume that $a_1 \geq a_2 \geq \cdots \geq a_n$. For some constant $c \leq 1$ the i-th authority value is defined as $a_i = ci^{-\beta}$, for $\beta < 1$. This implies a power law distribution on the *expected* in-degrees with exponent $\alpha > 2$. This is typical for most real-life graphs. The exponent of the in-degree distribution for the Web graph is 2.1 [9]. For the hub values we assume that $h_i = \Theta(1)$, for all $1 \leq i \leq n$. Therefore, we have that $H = \Theta(n)$, and $\|\boldsymbol{h}\|_2 = \Theta(\sqrt{n})$. Furthermore, it is easy to show that for $\beta < 1$, $\|\boldsymbol{a}\|_2^2 = \sum_{i=1}^{n} \frac{c}{i^{2\beta}} = \omega(1)$.



(a) INDEGREE (b) HITS

Fig. 1. INDEGREE and HITS distributions on the Web graph.

Therefore, $\|\boldsymbol{a}\|_2\|\boldsymbol{h}\|_2 = \omega(\sqrt{n})$, and $H\|a\|_2 = \omega(n)$, thus satisfying Assumptions 1 and 2. Therefore, we can conclude that HITS and INDEGREE are similar with high probability when the expected degrees follow a power law distribution. Note that on this graph we have that the expected number of edges is $HA = \omega(n \ln n)$.

5 Experimental analysis

In this section we study experimentally the similarity of HITS and INDEGREE on a large sample of the Web. We analyze a sample of 136M vertices and about 1,2 billion edges of the Web graph collected in 2001 by the WebBase project[5] 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 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, while 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. These results suggest that although the Web, as expected, is not a product graph, the HITS authority weights can be well approximated by the in-degrees.

	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

[5] `http://www-diglib.stanford.edu/~testbed/doc2/WebBase/`

6 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 i-th hub vector. We can prove the following proposition. The proof appears in Appendix A.8.

Proposition 1. *For a graph $G \in \overline{\mathcal{G}}_n$, the d_2 distance between* HITS *and* INDEGREE *is*

$$d_2(\boldsymbol{a}, \boldsymbol{w}) = \sqrt{\left(\frac{\sigma_2 H_2}{\sigma_1 H_1}\right)^2 + \cdots + \left(\frac{\sigma_n H_n}{\sigma_1 H_1}\right)^2} \tag{3}$$

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 $|H_i| \leq \sqrt{n}$. The following conditions guarantee the similarity of HITS and INDEGREE: (a) $\sigma_2/\sigma_1 = o(1/\sqrt{n})$, and there exists a constant k such that $\sigma_{k+1}/\sigma_1 = o(1/n)$; (b) $H_1 = \Theta(\sqrt{n})$, and $\sigma_2/\sigma_1 = o(1)$, and there exists a constant k such that $\sigma_{k+1}/\sigma_1 = o(1/n)$; (c) $H_1 = \Theta(\sqrt{n})$, 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 (3) 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 $R = W - W_1$ is called the residual matrix, and it has singular values $\sigma_2, \ldots, \sigma_n$. We have that

$$d_2(\boldsymbol{a}, \boldsymbol{w}) = o\left(\|W - W_1\|_F\right) \qquad \text{and} \qquad d_2(\boldsymbol{a}, \boldsymbol{w}) = o\left(\sqrt{\|W\|_F^2 - \|W\|_2^2}\right) \tag{4}$$

Equation (4) says that the similarity of 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 [1]. The similarity of the HITS and INDEGREE algorithms depends upon the contribution of the strongest linear trend to the sum of linear trends.

7 Conclusions

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.

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. Also, 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 [5, 16].

References

1. D. Achlioptas and F. McSherry. Fast computation of low rank matrix approximations. In *ACM Symposium on Theory of Computing (STOC)*, 2001.
2. L. A. Adamic and B. A. Huberman. Zipf's law and the internet. *Glottometrics*, 3:143–150, 2002.
3. W. Aiello, F. R. K. Chung, and L. Lu. Random evolution in massive graphs. In *IEEE Symposium on Foundations of Computer Science*, pages 510–519, 2001.
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)*, Greece, 2001.
5. A.-L. Barabasi and R. Albert. Emergence of scaling in random networks. *Science*, 286:509–512, 1999.
6. M. Bianchini, M. Gori, and F. Scarselli. Pagerank: A circuital analysis. In *Proceedings of the Eleventh International World Wide Web (WWW) Conference*, 2002.
7. A. Borodin, G. O. Roberts, J. S. Rosenthal, and P. Tsaparas. Link Analysis Ranking: Algorithms, Theory, and Experiments. *ACM Transactions on Internet Technology*, 05(1), 2005.
8. 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.
9. A. Broder, R. Kumar, F. Maghoul, P. Raghavan, S. Rajagopalan, R. Stata, A. Tomikns, and W. Wiener. Graph structure in the Web. In *Proceedings of WWW9*, 2000.
10. F. Chung and L. Lu. Connected components in random graphs with given degree sequences. *Annals of Combinatorics*, 6:125–145, 2002.
11. F. Chung and L. Lu. The average distances in random graphs with given expected degrees. *Internet Mathematics*, 1:91–114, 2003.
12. F. Chung, L. Lu, and V. Vu. Eigenvalues of random power law graphs. *Annals of Combinatorics*, 7:21–33, 2003.
13. P. Erdös and A. Rènyi. On the evolution of random graphs. *Publ. Math. Inst. Hung. Acad. Sci.*, 5:17–61, 1960.
14. R. Fagin, R. Kumar, and D. Sivakumar. Comparing top k lists. In *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2003.
15. J. Kleinberg. Authoritative sources in a hyperlinked environment. In *Proceedings of the Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 668–677, 1998.
16. 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*, 2000.
17. H. C. Lee and A. Borodin. Perturbation of the hyperlinked environment. In *Proceedings of the Ninth International Computing and Combinatorics Conference*, 2003.
18. 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*, 2000.
19. R. Lempel and S. Moran. Rank stability and rank similarity of link-based web ranking algorithms in authority connected graphs. In *Second Workshop on Algorithms and Models for the Web-Graph (WAW 2003)*, 2003.

20. M. Mihail and C. H. Papadimitriou. On the eigenvalue power law. In *Proceedings of the 6th International Workshop on Randomization and Approximation Techniques*, 2002.
21. R. Motwani and P. Raghavan. *Randomized Algorithms.* Cambridge University Press, Cambridge, England, June 1995.
22. A. Y. Ng, A. X. Zheng, and M. I. Jordan. Link analysis, eigenvectors, and stability. In *Proceedings of the International Joint Conference on Artifi cial Intelligence (IJCAI)*, 2001.
23. G. W. Stewart and J. Sun. *Matrix Perturbation Theory*. Academic Press, 1990.
24. G. K. Zipf. *Human Behavior and the principle of least effort*. Addison-Wesley, 1949.

A Supplementary material

A.1 Linear Algebra Background

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, W, we have that $\|MW\| \leq \|M\|\|W\|$. The two norms are related by the inequality $\|M\|_2 \leq \|M\|_F \leq \sqrt{n}\|M\|_2$.

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, $\Sigma = \mathrm{diag}(\sigma_1, \sigma_2, \ldots, \sigma_n)$, 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 k-th column vectors of matrix U and V respectively, is a pair of the k-th 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 $M^T M$. 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.

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 norm.

The HITS algorithm initializes all weights to one, and then iteratively updates the hub and authority vectors, 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 $W^T W$ and WW^T, respectively. Therefore, the hub vector $\boldsymbol{h}$ is the principal left singular vector of W, while the authority vector $\boldsymbol{a}$ is the principal left singular vector of W.

A.2 Metric property of the d_2 distance measure

For the following we use $\|\cdot\|$ to denote the L_2 norm.

Lemma 6. *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 $d_2(\boldsymbol{a},\boldsymbol{b}) \le \|\boldsymbol{a}-\boldsymbol{b}\|$. We will now prove that $d_2(\boldsymbol{a},\boldsymbol{b}) \ge \|\boldsymbol{a}-\boldsymbol{b}\|$, which implies that $d_2(\boldsymbol{a},\boldsymbol{b}) = \|\boldsymbol{a}-\boldsymbol{b}\|$.

Borodin et al. [7] prove that at least one of the constants γ_1,γ_2 should be equal to 1. Without loss of generality, assume that $\gamma_1 = 1$. We have that $d_2(\boldsymbol{a},\boldsymbol{b}) = \min_{\gamma\ge 1}\|\boldsymbol{a}-\gamma\boldsymbol{b}\|$. 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 $\|\boldsymbol{a}-\boldsymbol{b}\|^2 = 2-2\cos(\boldsymbol{a},\boldsymbol{b})$. Also we have that

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

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

A.3 Proof of Lemma 1

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 graph $G_1 = \arg\max_{G'\in\mathcal{C}_k(G)} d(\mathcal{A}_1(G),\mathcal{A}_1(G'))$ and graph $G_2 = \arg\max_{G'\in\mathcal{C}_k(G)} d(\mathcal{A}_2(G),\mathcal{A}_2(G'))$. By definition of the graph G_1, we have that $d(\mathcal{A}_1(G),\mathcal{A}_1(G_2)) \le d(\mathcal{A}_1(G),\mathcal{A}_1(G_1))$, 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

$$\begin{aligned}&d(\mathcal{A}_2(G),\mathcal{A}_2(G_2)) \le\\ &\qquad c\left(d(\mathcal{A}_2(G),\mathcal{A}_1(G)) + d(\mathcal{A}_1(G),\mathcal{A}_1(G_2)) + d(\mathcal{A}_1(G_2),\mathcal{A}_2(G_2))\right) .\end{aligned}$$

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

A.4 Proof of Lemma 2

We use results from perturbation theory [23] 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 which are defined as follows.

$$B = \begin{bmatrix} 0 & W^T \\ W & 0 \end{bmatrix} \qquad \text{and} \qquad F = \begin{bmatrix} 0 & E^T \\ E & 0 \end{bmatrix} \tag{5}$$

If σ_i is the i-th 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 [23] for the symmetric case).

Theorem 6. *Suppose 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 2.

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 6, define matrices B and F as in the Equation (5). 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 i-th 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,

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

□

A.5 Proof of Theorem 3

Proof. The proof follows directly from Lemma 2. 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 $\|\boldsymbol{a} - \boldsymbol{a}'\|_2 = o(1)$. □

A.6 Proof of Theorem 4

Proof. Assumption 1 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 [23] 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 3 it follows that HITS is stable on $\mathcal{G}_n^p$ with high probability. □

A.7 Proof of Lemma 5

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 $\boldsymbol{a}_q$ 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 $\gamma_1 = \frac{H\|\boldsymbol{a}\|}{\|\boldsymbol{d}\|} > 1$ 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. □

A.8 Proof of Proposition 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 i-th 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

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

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,

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

□