Generalizing PageRank: Damping Functions for Link-Based Ranking Algorithms

Ricardo Baeza-Yates
Yahoo! Research
Barcelona, Spain & Santiago, Chile
ricardo@baeza.cl

Paolo Boldi*
Università degli Studi di Milano
Milan, Italy
boldi@dsi.unimi.it

Carlos Castillo
Università di Roma "La Sapienza"
Rome, Italy
castillo@dis.uniroma1.it

ABSTRACT

This paper introduces a family of link-based ranking algorithms that propagate page importance through links. In these algorithms there is a damping function that decreases with distance, so a direct link implies more endorsement than a link through a long path. PageRank is the most widely known ranking function of this family.

The main objective of this paper is to determine whether this family of ranking techniques has some interest *per se*, and how different choices for the damping function impact on rank quality and on convergence speed. Even though our results suggest that PageRank can be approximated with other simpler forms of rankings that may be computed more efficiently, our focus is of more speculative nature, in that it aims at separating the kernel of PageRank, that is, link-based importance propagation, from the way propagation decays over paths.

We focus on three damping functions, having linear, exponential, and hyperbolic decay on the lengths of the paths. The exponential decay corresponds to PageRank, and the other functions are new. Our presentation includes algorithms, analysis, comparisons and experiments that study their behavior under different parameters in real Web graph data.

Among other results, we show how to calculate a linear approximation that induces a page ordering that is almost identical to PageRank's using a fixed small number of iterations; comparisons were performed using Kendall's τ on large domain datasets.

Categories and Subject Descriptors: H.4.m [Information Systems Applications]: Miscellaneous

General Terms: Algorithms.

Keywords: Link analysis, Link-based ranking, Web graphs.

1. INTRODUCTION

While traditional Information Retrieval (IR) methods are used by web search engines to some extent, the web is much more massive, dynamic and less coherent than traditional text collections [2].

*Partially supported by MIUR COFIN Project "Linguaggi formali e automi".

SIGIR'06, August 6–11, 2006, Seattle, Washington, USA.

While for any given topic there might be thousands or even millions of pages available, the problem of *ranking* those pages to generate a short list is probably one of the key problems of Web IR, and this requires some kind of relevance estimation.

One of the measures of importance of a scientific paper is the number of citations that the article receives. Following this idea, several authors proposed to use links for ranking web pages [28, 21, 25]; however, it quickly became clear that just counting the links was not a very reliable measure of authoritativeness (it was not in scientific citations either), because it is very easy to manipulate in the context of the web, where creating a page costs nearly nothing.

The PageRank technique, introduced by Page *et al.* [31], actually tries to mend this problem by looking at the importance of a page in a recursive manner: "a page with high PageRank is a page referenced by many pages with high PageRank". The algorithm not only counts the direct links to a page, but also includes indirect links. The same is valid for scientific and bibliographic citations in general.

PageRank is a simple, robust and reliable way to measure the importance of web pages, has a clear interpretation as a Markovian process, and can be computed in a very efficient way. For these reasons, most of today's commercial search engines are believed to use it as a part of their ranking function. In this paper we:

- describe general ranking functions that depend on incoming paths of varying lengths,
- show that PageRank belongs to this class of functions,
- show how to compute these rankings,
- compare the ranking orders induced by different ranking functions in real data, finding ways of approximating PageRank up to a very high precision.

The rest of this paper is organized as follows: Section 2 introduces the notion of functional ranking, Section 3 describes three damping functions, Section 4 compares them analytically, and Section 5 experiments with different parameters for each function. Finally, the last section presents our conclusions.

2. FUNCTIONAL RANKINGS

In this section, we introduce the notion of *functional ranking*, a general family of ranking functions that includes PageRank. To describe PageRank formally, we consider a web graph of N pages. Let $\mathbf{A}_{N\times N}$ be the adjacency matrix in this graph, $a_{i,j} = 1$ iff there is a link from page i to page j. This link matrix is seldom used as it is, mainly for two reasons:

Normalization. In the Web, creating an out-link is free, so there is an incentive for web page authors to create pages with many out-links; this is the reason why a metaphor of "voting" is enforced [26] in which each page has only one "vote" that has to be split among its linked pages. This is typically done in link-based ranking by normalizing **A** row-wise: the normalization process means that every web page can only decide how to divide its own score among the pages it leads to, but it cannot assign more score than it has. Another way to look at normalization is that the matrix is turned into the transition matrix of a stochastic process.

The normalization does not need to give each out-link the same value, as there is evidence that web links have different purposes such as navigating in a multi-page set, expanding the contents of the current page, pointing to another resource, etc. [17]. Also, links within the same site can be considered self-links and as such do not confer as much authority as a link between different sites; indeed, there are ranking methods like BHITS [6] that treat them differently. Other characteristics of links, such as the exploration level at which they appear in Web sites [27], or if they are at the beginning or the bottom of individual pages, or inside a certain HTML element, can also be used for non-uniform normalization [3].

To simplify our treatment, we will assume uniform normalization, so if a page has d out-links, each of those links has a weight of $1/d$, but the results of this paper can be applied to other forms of normalization.

Dangling nodes. Special attention should be paid to the possible presence of nodes with no outgoing arcs (known as "sinks" in graph theory): in fact, dangling nodes fail to produce a row-stochastic matrix, because the rows of dangling nodes are filled with zeroes. Dangling nodes can be dealt with by adding an extra node that is linked to and from all other nodes, or by introducing new arcs from each dangling node to every node in the graph [14]. In our analysis, we shall assume that all dangling nodes have been eliminated already in some way, so that we do not have to worry about their presence. All the algorithms we will present can be modified so that dangling nodes can be dealt with explicitly and with virtually no additional cost.

Let $\mathbf{P}$ be the row-normalized link matrix of the graph with N nodes. PageRank $\mathbf{r}(\alpha)$ is defined as the stationary distribution of the Markov chain with state transitions given by the matrix

$$\alpha\mathbf{P} + (1-\alpha)\mathbf{1}^T\mathbf{v}$$

where $\alpha \in [0,1)$ is a parameter called *damping factor* (sometimes also called a dampening factor), and $\mathbf{v}$ is a fixed *preference vector* that may represent the interests of a particular user, or another ranking vector that is used for weighting pages. Note that the above matrix is ergodic (at least, if every entry of $\mathbf{v}$ is strictly positive), so it has exactly one stationary distribution. Even though most of our results can be easily restated with a non-uniform preference vector $\mathbf{v}$, for the sake of clarity we shall only consider the uniform preference $\mathbf{1}/N$ in the rest of the paper.

As observed in [15, 8], the PageRank vector $\mathbf{r}(\alpha)$ can be written as:

$$\mathbf{r}(\alpha) = (1-\alpha)\sum_{t=0}^{\infty}\alpha^t\frac{1}{N}\mathbf{1}\mathbf{P}^t,$$

or in matricial form:

$$\mathbf{r}(\alpha) = (1-\alpha)\frac{1}{N}\mathbf{1}(\mathbf{I}-\alpha\mathbf{P})^{-1} \qquad ||\alpha\mathbf{P}|| < 1.$$

There is an equivalent, and actually very intriguing way of rewriting this formula, mentioned in [30] that leads to a conclusion similar to those of [10]: given a path, that is, a sequence of edges in the graph $p = \langle x_1, x_2, \ldots, x_k\rangle$, such that node x_i is connected to node x_{i+1}, we define its *branching contribution* as follows

$$\text{branching}(p) = \frac{1}{d_1 d_2 \cdots d_{k-1}}$$

where d_j is the outdegree, this is, the number of outgoing arcs, of node x_j.

Then, the ranking of node i according to PageRank is

$$r_i(\alpha) = \sum_{p\in\text{Path}(-,i)}\frac{(1-\alpha)\alpha^{|p|}}{N}\,\text{branching}(p)$$

where $\text{Path}(-,i)$ is the set of all paths into node i and $|p|$ is the length of path p: this is because $(\mathbf{P}^t)_{ij}$ contains the sum of the branching contributions of all paths of length t from i to j, as one can easily show by induction on t (a path of length 0 and branching 1 is also included in the summation). This way of expressing the PageRank of a node is interesting, because it highlights the fact that the rank of a node is essentially obtained as a weighted sum of contributions coming from every path entering into the node, with weights that decay exponentially in the length of the path.

A natural generalization of this idea consists in taking into consideration a ranking $\mathbf{R}$ of the general form:

$$\mathbf{R} = \sum_{t=0}^{\infty}\text{damping}(t)\frac{1}{N}\mathbf{1}\cdot\mathbf{P}^t$$

or equivalently

$$R_i = \sum_{p\in\text{Path}(-,i)}\text{damping}(|p|)\frac{1}{N}\text{branching}(p) \tag{1}$$

where the damping function is a suitable choice of weights.

We call this form of ranking a *functional ranking* as it is parametrized by a damping function. This generalizes Lifantsev's [26] model in which the damping factor is a matrix of *voting trust* that is fixed during the computation, while in our case, this depends explicitly on the iterations. Our damping function could be even more general by using $\mathbf{D}(t)$, a damping matrix instead of $\text{damping}(t)\frac{1}{N}\mathbf{1}$; in this paper we analyze only the latter form. Fogaras [15] proposed using decreasing link weights depending on path lengths in the reverse link graph, and used exponentially decreasing weights as in PageRank for finding good Web browsing "starting points" in the Web graph. Another, yet unexplored, possible direction would be to consider damping functions that depend on other properties of the paths (e.g., whether the path passes through some node out of a certain set) rather than on their length.

As we have seen, generic PageRank is a functional ranking where the damping function

$$\text{damping}(t) = (1-\alpha)\alpha^t$$

decays exponentially fast (something similar was first considered in citation analysis back in 1953! [22]). The next section shows several functional rankings by describing their damping functions.

3. DAMPING FUNCTIONS

Formula (1) defines a form of ranking that is parametrized by a damping function; the latter describes how rapidly the importance of paths decays as the path length increases. A first, if only formal, problem is establishing which class of damping functions generates well-defined functional rankings.

THEOREM 1. *Every damping function such that the sum of dampings is 1 yields a well-defined normalized functional ranking.*

Proof. As shown in [10, Corollary 2.4], for every pair of nodes i and j, and for every length t

$$\sum_{p\in\text{Path}(i,j),|p|=t} \text{branching}(p) \leq 1.$$

In other words, the sum of branching contributions of all paths of a certain length between two specific nodes does never exceed 1. A more general property holds (the proof is an easy induction on the path length): for every node i and every length t

$$\sum_{p\in\text{Path}(i,-),|p|=t} \text{branching}(p) = 1.$$

As a consequence, to guarantee that the functional ranking is well-defined and normalized (i.e., that rank values sum to 1) we need:

$$\sum_{i=1}^{N} \sum_{p\in\text{Path}(-,i)} \text{damping}(|p|)\frac{1}{N}\text{branching}(p) = 1$$

or equivalently

$$\sum_{t=0}^{\infty} \text{damping}(t)\frac{1}{N} \sum_{p\in\text{Path}(-,-),|p|=t} \text{branching}(p) = 1\ .$$

As $\sum_{p\in\text{Path}(-,-),|p|=t}\text{branching}(p) = N$ the latter equality is equivalent to

$$\sum_{t=0}^{\infty} \text{damping}(t) = 1.$$

∎

Of course, not all choices are equivalent, so we have to find out which functions generate better rankings. Since a direct link should be more valuable as a source of evidence than a distant link, we focus on damping functions that are decreasing on t, the length of the paths.

Computation. For calculating functional rankings, we use the general algorithm shown in Figure 1; the next sections provide details on the initialization, stop condition and iteration steps for each calculation.

Require: N: number of nodes, **v**: preference vector
1: **for** i : 1 … N **do** {Initialization}
2: S[i] ← R[i] ← START
3: **end for**
4: **for** k : 1 … ∞ **do** {Iteration step}
5: **if** STOP **then**
6: **break**
7: **end if**
8: Aux ← **0**
9: **for** i : 1 … N **do** {Follow links in the graph}
10: **for all** j such that there is a link from i to j **do**
11: Aux[j] ← Aux[j] + R[i]/outdegree(i)
12: **end for**
13: **end for**
14: **for** i : 1 … N **do** {Add to ranking value}
15: R[i] ← Aux[i] × DAMP(k)
16: S[i] ← S[i] + R[i]
17: **end for**
18: **end for**
19: **return** S

Figure 1: Template algorithm for computing a functional damping. START, STOP and DAMP(k) differ for each functional ranking.

3.1 Exponential damping: PageRank

As we already noted, PageRank can be seen as a functional ranking where the damping function decays exponentially:

$$\text{damping}(t) = (1-\alpha)\alpha^t.$$

Since longer paths have less importance in the calculation of PageRank, it could be approximated by using only a few levels of links. In [11], it is shown that by using only the nodes at distance 1 from a target node (equivalent to linear damping with $L=2$), PageRank values can be approximated with 30% of average error. Using nodes at distance 2, the average error drops to 20% and at distance 3, to 10%. After that, there are no significant improvements by adding a few more levels, and the cost (the number of nodes to be explored) is much higher.

Computation. Since PageRank is the principal eigenvector of the modified graph matrix, it can be easily approximated by the iterative Power Method algorithm, as suggested by Page *et al.* in their original paper [31]; this iterative algorithm gives good approximations (both in norm and with respect to the induced node order) in few iterations, even though convergence speed and numerical stability decay when α gets close to 1 [19, 18].

3.2 Linear damping

As an (extreme) alternative to PageRank, let us consider a simple damping function such as:

$$\text{damping}(t) = \begin{cases} \frac{2(L-t)}{L(L+1)} & t < L \\ 0 & t \geq L \end{cases}$$

that is, a damping function that decreases linearly with distance, and reaches zero at distance L. The trivial case $L=1$ gives a uniform ranking, and $L=2$ is ranking by indegree, as in the latter case all paths of length ≥ 2 are not considered.

From the definition,

$$\begin{aligned} \mathbf{R} &= \sum_{t=0}^{\infty}\text{damping}(t)\mathbf{v}\mathbf{P}^t = \sum_{t=0}^{L}\frac{2(L-t)}{L(L+1)}\mathbf{v}\mathbf{P}^t \\ &= \frac{2}{L(L+1)}\mathbf{v}\sum_{t=0}^{L-1}(L-t)\mathbf{P}^t \\ &= \frac{2}{L(L+1)}\mathbf{v}(L(\mathbf{I}-\mathbf{P})-\mathbf{P}(\mathbf{I}-\mathbf{P}^L))((\mathbf{I}-\mathbf{P})^2)^{-1}\ . \end{aligned}$$

provided that $(\mathbf{I}-\mathbf{P})^2$ is not singular.

An advantage of this type of ranking is that only the first few levels are considered, so the number of iterations is fixed. The rationale for this is that after a certain distance the information given by links can be disregarded.

Computation. For computing this functional ranking, we can define the following sequence:

$$\begin{aligned} \mathbf{R}^{(0)} &= \frac{2}{L+1}\mathbf{v} \\ \mathbf{R}^{(k+1)} &= \frac{(L-k-1)}{(L-k)}\mathbf{R}^k\mathbf{P}\ . \end{aligned}$$

The functional ranking with linear damping is $\sum_{k=0}^{L-1}\mathbf{R}^{(k)}$. For computing this ranking, the generic algorithm shown in Figure 1 can be used, with:

$$\begin{aligned} \text{START} &: 2v[i]/(L+1) \\ \text{STOP} &: k = L \\ \text{DAMP(k)} &: (\text{L}-\text{k})/(\text{L}-(\text{k}-1)) \end{aligned}$$

3.3 Quadratic hyperbolic damping: TotalRank

Recently, a ranking method called TotalRank [7] has been proposed. The method aims at eliminating the necessity for an arbitrary parameter by integrating PageRank over the entire range of α. If $\mathbf{r}(\alpha)$ is the vector of PageRank, then TotalRank is defined as:

$$\mathbf{T} = \int_0^1 \mathbf{r}(\alpha) d\alpha \, .$$

$\mathbf{T}$ can be written as:

$$\begin{aligned} \int_0^1 \mathbf{r}(\alpha) d\alpha &= \frac{1}{N} \sum_{t=0}^{\infty} \int_0^1 (1-\alpha)\alpha^t \mathbf{1} \cdot \mathbf{P}^t d\alpha \\ &= \frac{1}{N} \sum_{t=0}^{\infty} \frac{1}{(t+1)(t+2)} \mathbf{1} \cdot \mathbf{P}^t, \end{aligned}$$

where the first equality is obtained applying Theorem 1.27 of [33].

Provided that $\mathbf{P}$ is not singular and $\mathbf{P} \neq \mathbf{I}$, we can write TotalRank using the definition of the logarithm of a matrix:

$$\ln(\mathbf{I}-\mathbf{P}) = -\sum_{k=1}^{\infty} \frac{\mathbf{P}^k}{k}$$

$$\mathbf{T} = \mathbf{P}^{-1}(\mathbf{I} + (\mathbf{I}-\mathbf{P}^{-1})\ln(\mathbf{I}-\mathbf{P}))$$

TotalRank is a weighted sum of the scores associated with paths of varying lengths, in which the weights are hyperbolically decreasing on the lengths of the paths. In other words, TotalRank is a functional ranking with damping function:

$$\text{damping}(t) = \frac{1}{(t+1)(t+2)} = \frac{1}{t+1} - \frac{1}{t+2} \, ,$$

which is well defined since $\sum_{t=0}^{\infty} \text{damping}(t) = 1$.

Computation. It is known that the cost of calculating TotalRank is the same as the cost of calculating PageRank via the Power Method [8], even though some more iterations are required to obtain the same precision.

3.4 General hyperbolic damping: HyperRank

TotalRank is part of a more general family of weighting schemes for paths of different lengths that can be approximated using:

$$\mathbf{s}(\beta) = \frac{1}{N\zeta(\beta)} \sum_{t=0}^{\infty} \frac{1}{(t+1)^{\beta}} \mathbf{1} \cdot \mathbf{P}^t \, .$$

Again, this way of ranking follows the general scheme, with damping function chosen as

$$\text{damping}(t) = \frac{1}{\zeta(\beta)(t+1)^{\beta}} \, .$$

Here, we are using Riemann's zeta function, $\zeta(\beta) = \sum_{t=1}^{\infty} t^{-\beta}$ for normalization, and we need $\beta > 1$ for it to converge. Note that when $\beta = 2$ we get weights similar to those of TotalRank, in which the t-th coefficient is $1/(t+1)(t+2)$ whereas here it is $1/\zeta(2)(t+1)^2$.

A meaningful choice for β should be done considering the distribution of paths of different lengths in a scale-free graph. A large α in PageRank, or a small β in HyperRank, means increasing the effect of longer paths in the score.

Computation. Let us define a vector sequence $\mathbf{R}^{(t)}$ as follows:

$$\begin{aligned} \mathbf{R}^{(0)} &= \frac{1}{N\zeta(\beta)} \\ \mathbf{R}^{(k+1)} &= \left(\frac{k+1}{k+2}\right)^{\beta} \mathbf{R}^{(k)} \mathbf{P} \, . \end{aligned}$$

It is easy to see that $\sum_{t=0}^{\infty} \mathbf{R}^{(k)} = \mathbf{s}(\beta)$, because $\mathbf{R}^{(k)} = 1/(N \cdot \zeta(\beta)(k+1)^{\beta}))\, \mathbf{1} \cdot \mathbf{P}^k$; this observation allows us to use the generic algorithm of Figure 1 with the following parameters:

$$\begin{aligned} \text{START} &: v[i]/\zeta(\beta) \\ \text{STOP} &: \text{convergence} \\ \text{DAMP(k)} &: (k/(k+1))^{\beta} \end{aligned}$$

Note that convergence speed is much slower than ordinary PageRank, especially when β is close to 1, the norm of the k-th summand being bound by $1/(1+1/k)^{\beta}$. Interestingly enough, though, convergence speed is reasonable if β is sufficiently large.

3.5 An empirical damping

An empirical damping function would consider how much the value of an endorsement decreases by following longer paths in the real web graph. This cannot be known exactly, but we can attempt to measure it indirectly. Pages that link to each other are more similar than pages chosen at random [13]; evidence from topical crawlers [34] shows that when doing breadth-first exploring, the topic "drifts" as the distance increases. On the same line of thought, we propose to use the decrease of text similarity as an approximation to an "empirical" damping function. In [29] it is shown that text similarity and link distance are anticorrelated up to 4-5 links.

To find out which is the correlation between link-distance and similarity, we performed the following experiment: we considered a web graph corresponding to a partial snapshot of the `.uk` domain with 18 million pages, and sampled 200 nodes at random. For each sampled node, we followed links backwards to obtain nodes at a minimum distance of 1, 2, 3, 4, or 5 links. Then, we sampled 12,000 pairs at each minimum distance at random, and computed their similarities with the original nodes. Similarity was measured using the normalization of TF.IDF [4], without stemming or stopword removal.

The resulting averages are shown in Figure 2, with standard deviation error bars. Text similarity clearly decreases with distance, and in some applications the empirical distribution of text similarity versus distance could be used as an "empirical" damping function. Different measures of text similarity can yield different distributions; for instance [36] uses the number of repeated words and phrases between pages and obtains a faster decrease in similarity. Our results show that a linear damping with $L = 8$ or $L = 9$ approximates better text similarity than an exponential damping as suggested in [29]; also, for different communities the link structure

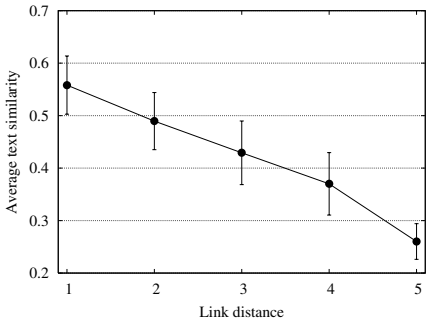

Figure 2: Link distance vs. average text similarity. A link distance of one means a direct link exists. Text similarity appears to decrease linearly in the first few levels of links.

could be different (e.g. academic vs. commercial Web subsets), so we should measure first which is the correlation of link distance to text similarity in the specific collection we want to rank.

4. COMPARING DAMPING FUNCTIONS

A comparison of the damping functions described in the previous section is shown in Figure 3: of course, hyperbolic damping functions decay asymptotically more slowly than exponential damping, but notice that for short paths the latter may dominate the former in many cases.

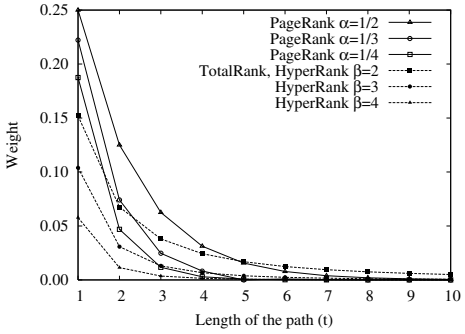

Figure 3: Weights given by the different damping functions, for some values of α and β.

In this section, we aim at analyzing how similar are these functional rankings, and how we could use one of the damping functions to approximate another with a suitable choice of parameters.

4.1 Approximating HyperRank with PageRank

Now we want to approximate the weights of:

$$\mathbf{s}(\beta) = \frac{1}{N\zeta(\beta)} \sum_{t=0}^{\infty} \frac{1}{(t+1)^{\beta}} \mathbf{P}^t$$

using the weights of:

$$\mathbf{r}(\alpha) = \frac{1-\alpha}{N} \sum_{t=0}^{\infty} \alpha^t \mathbf{P}^t \ ,$$

and we proceed again by considering paths up to a certain length:

$$\sum_{t=0}^{\ell} \left(\frac{1}{\zeta(\beta)(t+1)^{\beta}} - (1-\alpha)\alpha^t \ . \right)$$

The minimum can be zero, and it is attained at:

$$\alpha^*(\ell,\beta) = \sqrt[\ell]{1 - \frac{1}{\zeta(\beta)} \sum_{t=0}^{\ell} \frac{1}{(t+1)^{\beta}}} \ .$$

The α that minimizes the difference of weights for different values of β and of the maximum path lengths ℓ is shown in Figure 4. In the case of $\beta = 2$, for instance, for path lengths up to 10 to 20, the best α is between 0.75 and 0.85.

4.2 Approximating PageRank with LinearRank

For approximating the damping function of PageRank with the damping function of LinearRank, we consider the summation of the differences up to a certain path length. If $\ell \leq L$:

$$\sum_{t=0}^{\ell} \left((1-\alpha)\alpha^t - \frac{2(L-t)}{L(L+1)} \right)$$

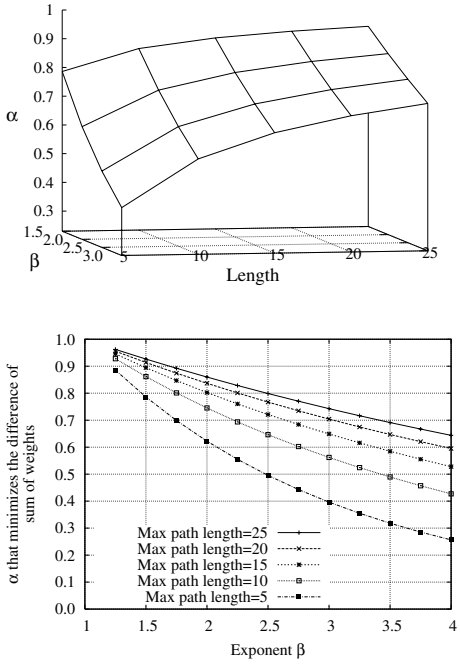

Figure 4: Best α for minimizing the difference of the sum of weights between PageRank and HyperRank, for various parameter combinations.

And if $\ell > L$:

$$\sum_{t=0}^{L-1} \left((1-\alpha)\alpha^t - \frac{2(L-t)}{L(L+1)} \right) + \sum_{t=L}^{\ell} (1-\alpha)\alpha^t$$

We will assume that $\ell \leq L$, so the evaluation of the difference between the two rankings is done in an area in which both rankings have non-zero values. The L that minimizes the difference for a given combination of α and ℓ is

$$\begin{aligned} L^*(\alpha,\ell) &= \ell + \frac{(2\ell+1)\alpha^{\ell+1} + 1 + \sqrt{(1+\alpha^{\ell+1})^2 + 4\ell(\ell+2)\alpha^{\ell+1}}}{2(1-\alpha^{\ell+1})} \\ &= \ell + 1 + O\left(\ell\alpha^{(\ell+1)/2}\right) \end{aligned}$$

and we have plotted it for different values of α and ℓ in Figure 5.

5. PARAMETERS FOR THE DAMPING FUNCTIONS

For our experiments, we used several snapshots from the Web, including the `.uk`, `.it` and `.eu.int` domains. For comparison, we also considered a synthetic scale-free network produced according to the evolving model described by Kumar *et al.* [24] (a combination of preferential attachment and random links) with the parameters suggested by Pandurangan *et al.* [32]. As far as the latter is concerned, in the generated graph the exponents for the power-law in the center part of the distributions are -2.1 for in-degree and PageRank, and -2.7 for out-degree; we generated a 100,000-nodes graph without disconnected nodes.

In this section, we study the behavior of the ranking functions for different values of their parameters.

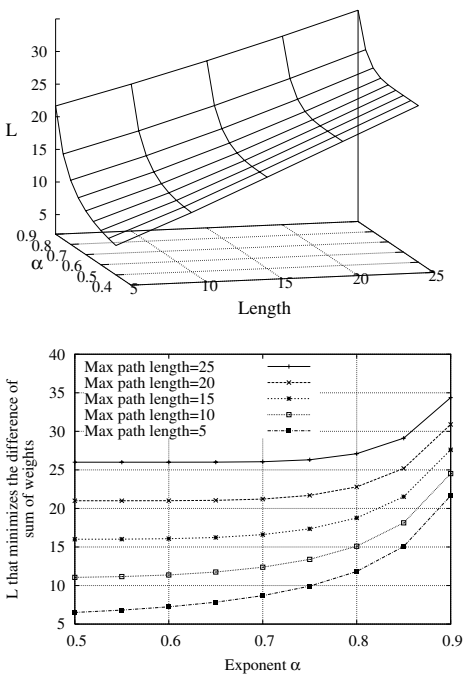

Figure 5: Best L for minimizing the difference of the sum of weights between LinearRank and PageRank, for various parameter combinations.

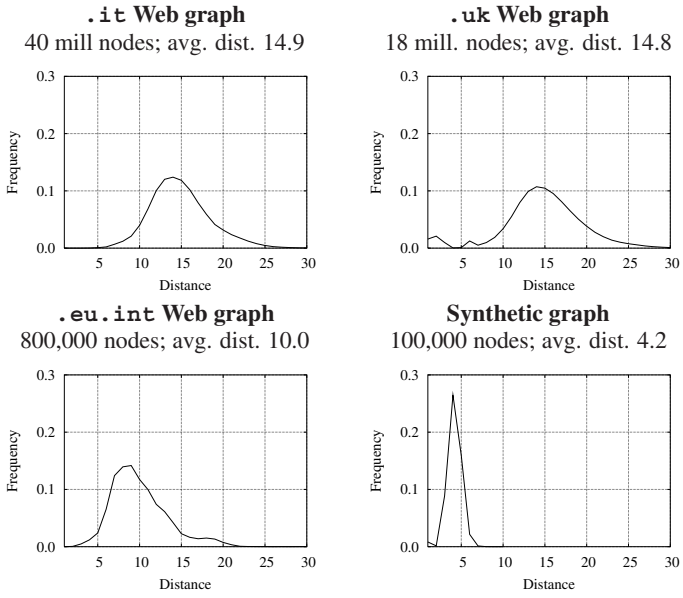

Figure 6: Distribution of the average number of nodes at a certain distance from a given node in three Web samples and a synthetic scale-free network.

5.1 Characteristic path lengths

In scale-free networks, the distances between pairs of nodes follow a Gaussian distribution [1] (the average is not given in their paper). Analytic estimations for the average distance of a graph of scale-free network of n nodes include: $O(\log(n))$ [35]; $O(\log(n)\ /\log(np))$ in sparse graphs with p links [12]; $1+\log(n/z_1)/\log(z_2/z_1)$ where z_1 is the average indegree, and z_2 is the average number of nodes at distance 2 [30]; and $O(\log(n)/\log(\log(n)))$ [9].

We did the following experiment: starting from a node picked at random, we followed the links backwards and counted the number of nodes at different distances. The average distances found, appear to be growing (sub)logarithmically with the size of the graph. Figure 6 shows the distribution obtained in each sample (the synthetic graph has less variance due to its small size). For this experiment, we are not counting the pages without in-links.

The act of linking a page represents human endorsement and should not be affected by the size of the graph. Neither the act of following a link, in terms of a random surfer, should be affected. However, an algorithm for *propagating* this endorsement through links for computing a ranking function needs to account for the typical distances involved; this need is typical in a situation where local properties have a global impact: for example, the addition of a single arc may reduce drastically the diameter of a graph. In most cases, researchers have used exponential damping with base 0.85 or 0.90 in graphs that are much smaller than the full Web (concept graphs, social networks, e-mail graphs, etc.), meaning that a potentially much larger fraction of the nodes contributed towards link ranking. We consider that in a smaller graph, the damping function should decay faster.

Let's suppose that for a graph with N_1 nodes it is found, by experimental or analytic means, that a good parameter for PageRank is α_1^*. Now, we would like to have a good parameter α_2^* for a graph with the same properties, except that the size of the new graph is $N_2 < N_1$. One possible approach, consistent with what we have done so far, is to consider that the sum of the weights up to the average path lengths of the graphs (L_1, L_2) have to be similar for both rankings to behave in a similar way. If we take this approach:

$$\begin{aligned} 1-(\alpha_1^*)^{L_1+1} &= 1-(\alpha_2^*)^{L_2+1} \\ \alpha_2^* &= (\alpha_1^*)^{\frac{L_1+1}{L_2+1}} \approx (\alpha_1^*)^{\frac{\log(N_1)}{\log(N_2)}} \end{aligned}$$

An example that can be used in practice is the following: let's consider a web graph with $N_1 = 11.5\times10^9$ pages (the size of the full Web estimated by [16]), and another graph with only $N_2 = 50\times10^6$ pages (the size of the Web of a large country); the second graph is roughly 3 orders of magnitude smaller.

If it is shown empirically that $\alpha_1^* = 0.85$ is a good value for the PageRank parameter for the whole Web, then $\alpha_2^* = 0.81$ should have a similar behavior in the 50-million page set, which is natural as the path lengths are shorter. If the subset of web pages were even smaller, for instance, $N_2 = 10^6$ pages (the size of the web of a large organization), then $\alpha_2^* = 0.76$, and for smaller graphs of $N_2 = 10^5$ nodes, $\alpha_2^* = 0.72$. We recommend using these values for graphs that are not comparable in size to the full Web graph.

5.2 Experimental comparison

In this section, we present experimental results about the similarity between the ranking orders induced by some of the functional rankings discussed in the previous sections. To perform the experiments, we used data from the U.K. Web graph. To compare ranking orders, we used Kendall's τ: a correlation measure related to the number of inversions in the rank order of one variable when the data is ordered according to the other variable ($\tau = 1$ means perfect agreement, $\tau = -1$ means reverse ordering).

We tested the correlation of in-degree with these damping functions. In general the correlation drops as more levels of links are considered (we omit the details here for lack of space, they will appear in the full version of this paper). Figure 7 (a) shows how

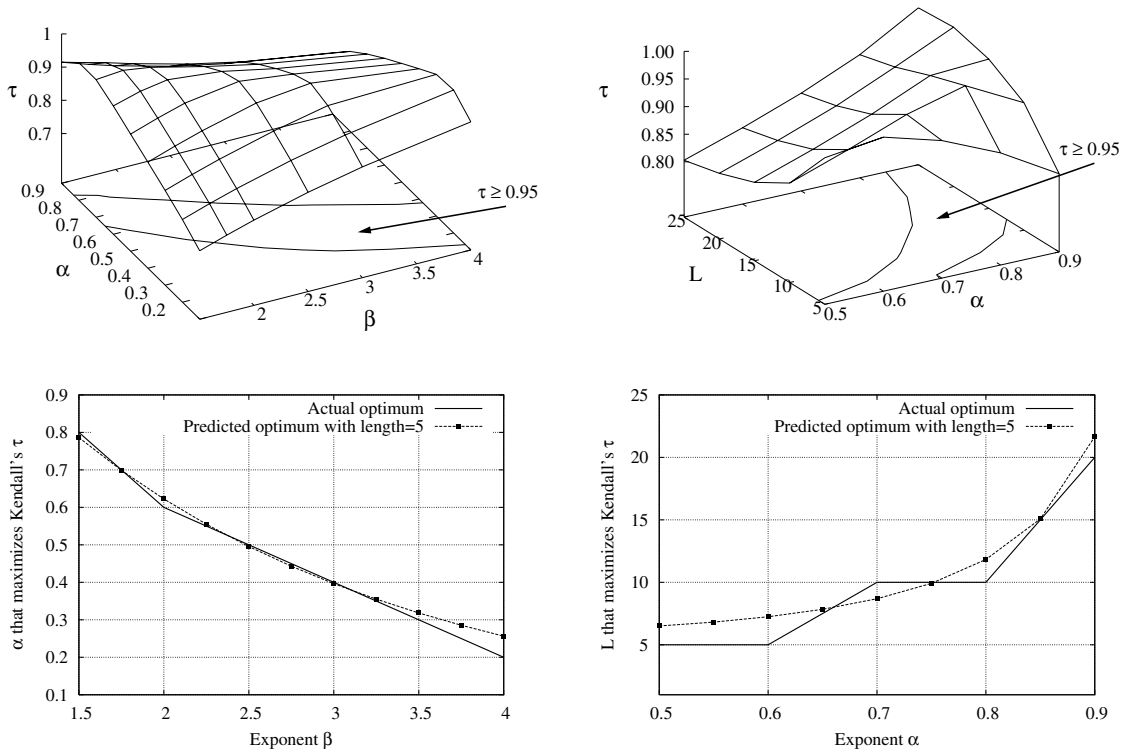

Figure 7: Comparison (using Kendall's τ) between ranking orders in the U.K. web graph, with various damping parameters. There is a region of the parameter space in which the ranking orders are very close. The predicted combination of parameters that yields the minimum difference with $\ell = 5$, is very close to the actual optimum in both cases.

PageRank compares with HyperRank for various pairs of α and β. In the limit $\alpha, \beta \to 1$ both rankings are equivalent, and they remain similar in a large region of the parameter space. We can see that the rankings obtained with HyperRank and PageRank can be almost equivalent (Kendall's $\tau \geq 0.95$). Moreover, the analysis shown in section 4.1 considering only paths of lengths less than 5, provides a very good approximation for the optimal combination of parameters. This means that in fact, the difference in the damping functions in the first few levels is crucial.

The exponents β required for giving a good approximation of PageRank are small when $\alpha \geq 0.7$, limiting the practical applicability of HyperRank, as it does not converge more quickly than PageRank. As far as LinearRank and PageRank with $\alpha = 0.8 \ldots 0.9$ are concerned, paths of roughly 10 to 20 links should be considered to obtain rankings that are almost equivalent, as shown in Figure 7 (b).

The predicted optimum given in section 4.2 with $\ell = 5$ (i.e., considering only the summation of the differences between both damping functions up to paths of length 5) is very close to what was obtained in practice. For $\alpha = 0.8$, calculating LinearRank with $L = 10$ (which means the same number of iterations) gives $\tau \geq 0.98$; for $\alpha = 0.9$, calculating LinearRank with $L = 15$ also gives $\tau \geq 0.98$. In both cases, the ranking order of PageRank is approximated by the ranking order of LinearRank with very high precision.

Precision Finally, we focused our attention on the LinearRank ranking that uses linear damping, to see if LinearRank with a small number of iterations can provide a ranking that is competitive with PageRank. With this aim in mind, we used the WebTREC Gov2 collection (available from the University of Glasgow for research purposes, see `http://ir.dcs.gla.ac.uk/test_collections/`). This collection consists of about 25 million documents obtained in 2004 from the `.gov` domain. We picked at random 50 tasks and manually created keyword queries for this evaluation, following the policy used in the standard *ad hoc* TREC tasks. We then used the Managing Gigabytes for Java (`mg4j`) framework to select from the collection 1000 pages matching each query according and re-ordered the query results according to the scores resulting from different link-based ranking strategies.

On this graph, the PageRank calculation took 39 iterations to converge on the L1-norm of the difference between two iterations to less than 10^{-6}. We computed the standard precision and recall measures [4] and averaged them across all queries. Precision at result number *N* is the fraction of correct results in the first *N* results returned by the system; the "correct" results in our case are taken from the quality assessments included in this reference collection. This is shown in Figure 8.

Of course using link ranking improves the precision over no ranking at all, and PageRank and LinearRank behave very similarly. For instance, if we compare the PageRank (that requires 39 iterations) with LinearRank at distance 5 (that requires 5 iterations only) we observe that the precision of the first element is 8% better for PageRank, of the first five elements is 17% better for PageRank, but for the first ten elements is 2% better for LinearRank. From that point over, both rankings are roughly equivalent.

This means that LinearRank at distance 5 can provide a level of precision for information retrieval tasks that is quite similar to that of PageRank. This is applicable in contexts where link-based

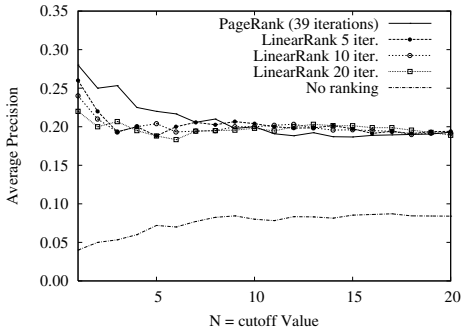

Figure 8: Evaluation of the precision of LinearRank and PageRank in the WebTREC Gov2 collection.

ranking cannot be computed in advance, but a computation at query time is necessary. For instance, this occurs if we need to analyze links over a sub-graph that is generated at query time.

6. CONCLUSIONS

In this paper we have defined a broad class of link-based ranking algorithms based on the contribution of damping factors along all the different paths reaching a page. We found that functional rankings using different damping functions can provide similar orderings, if the parameters are chosen carefully. LinearRank can be used for calculating a ranking that is as good as PageRank, but with a fixed, and smaller, number of iterations. Also, the parameters for the damping functions depend on the characteristic path lengths in the graph, which are known to grow sub-logarithmically on the size of the graph.

More work is needed to find other damping functions that compute rankings similar to PageRank but are easier and faster to compute. We use a global ranking similarity, but another measure could be the ranking similarity in the top 20 results of real queries. In this setting our results can change, so future work will include this variation. Our results show that the exponential damping used by PageRank is not that special.

Because of their high cost, link-based ranking methods that involve iterative calculations at query time are probably not used by large-scale search engines at this moment, but the functional ranking with linear damping we have presented can provide a good approximation with few iterations. Moreover, the approach we have presented could be also applied to multivalued ranking functions such as HITS [23] and topic-sensitive PageRank [20] to obtain, for instance, a method for approximating the hubs and authority scores using less iterations and a linear damping function.

Our approach also helps to understand how easy or difficult it is to collude many pages to modify the ranking of a given page. Clearly there are many different factors: path lengths, damping function, branching degrees, and number of colluded pages. The graph structure of the collusion will affect those factors and we plan to analyze them. In particular, under the assumption that is easier to "spam" closer links, PageRank damping is more affected by collusion than the rest of the damping functions presented here. In [5] a truncated exponential damping, combined with other link-analysis techniques, is used for spam detection.

Acknowledgements: we would like to thank Dániel Fogaras for a valuable discussion about TotalRank that motivated part of this research. The authors also thank the support from ICREA and the Cátedra Telefónica at Universitat Pompeu Fabra.

7. REFERENCES

[1] R. Albert, H. Jeong, and A. L. Barabási. Diameter of the World Wide Web. *Nature*, 401:130–131, 1999.

[2] A. Arasu, J. Cho, H. Garcia-Molina, A. Paepcke, and S. Raghavan. Searching the Web. *ACM TOIT*, 1(1):2–43, 2001.

[3] R. Baeza-Yates and E. Davis. Web page ranking using link attributes. In *Alt. track papers & posters, WWW Conf.*, pp. 328–329, New York, NY, USA, 2004.

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

[5] L. Becchetti, C. Castillo, D. Donato, S. Leonardi and R. Baeza-Yates. Using rank propagation and probabilistic counting for link-based spam detection. Tech. report DELIS-TR-0341, Dynamically-Evolving Large-Scale Inf. Sys. 2006.

[6] K. Bharat and M. R. Henzinger. Improved algorithms for topic distillation in a hyperlinked environment. In *Proc. of ACM SIGIR*, pp. 104–111, Melbourne, Australia, 1998. ACM Press, New York.

[7] P. Boldi. TotalRank: ranking without damping. In *Poster Proc. of WWW Conf.*, pp. 898–899, Chiba, Japan, 2005. ACM Press.

[8] P. Boldi, M. Santini, and S. Vigna. PageRank as a function of the damping factor. In *Proc. of WWW Conf.*, pp. 557–566, Chiba, Japan, 2005. ACM Press.

[9] B. Bollobás and O. Riordan. The diameter of a scale-free random graph. *Combinatorica*, 24(1):5–34, 2004.

[10] M. Brinkmeier. PageRank revisited. *ACM TOIT*, 6(3):257–279, 2006.

[11] Y.-Y. Chen, Q. Gan, and T. Suel. Local methods for estimating PageRank values. In *Proc. of CIKM*, pp. 381–389, New York, USA, 2004. ACM Press.

[12] F. Chung and L. Lu. The diameter of random sparse graphs. *Adv. Appl. Math.*, 26:257–279, 2001.

[13] B. D. Davison. Topical locality in the Web. In *Proc. of ACM SIGIR*, pp. 272–279, Athens, Greece, 2000. ACM Press.

[14] N. Eiron, K. S. Mccurley, and J. A. Tomlin. Ranking The web frontier. In *Proc. of WWW Conf.*, pp. 309–318, New York, USA, 2004. ACM Press.

[15] D. Fogaras. Where to start browsing the Web? In *IICS*, vol. 2877 of *Springer LNCS*, pp. 65–79, Leipzig, Germany, 2003.

[16] A. Gulli and A. Signorini. The indexable Web is more than 11.5 billion pages. In *Poster Proc. of WWW Conf.*, pp. 902–903, Chiba, Japan, 2005. ACM Press.

[17] S. W. Haas and E. S. Grams. Page and link classifications: connecting diverse resources. In *DL '98: Proc. of ACM Conf. on Digital libraries*, pp. 99–107, New York, NY, USA, 1998. ACM Press.

[18] T. Haveliwala and S. Kamvar. The condition number of the PageRank problem. Tech. Report 36, Stanford University, 2003.

[19] T. Haveliwala and S. Kamvar. The second eigenvalue of the Google matrix. Tech. Report 20, Stanford University, 2003.

[20] T. H. Haveliwala. Topic-sensitive PageRank. In *Proc. of WWW Conf.*, pp. 517–526, Honolulu, Hawaii, USA, 2002. ACM Press.

[21] W.-K. Joo and S. H. Myaeng. Improving retrieval effectiveness with hyperlink information. In *Proc. of IRAL*, Singapore, 1998.

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

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

[24] R. Kumar, P. Raghavan, S. Rajagopalan, D. Sivakumar, A. Tomkins, and E. Upfal. Stochastic models for the Web graph. In *Proc. of FOCS*, pp. 57–65, Redondo Beach, USA, 2000. IEEE CS Press.

[25] Y. Li. Toward a qualitative search engine. *IEEE Internet Comp.*, July 1998.

[26] M. Lifantsev. Voting model for ranking Web pages. In *Proc. of the Int. Conf. on Internet Computing*, pp. 143–148, Las Vegas, Nevada, USA, 2000.

[27] T.-Y. Liu and W.-Y. Ma. Webpage importance analysis using conditional Markov random walk. In *Proc. of IEEE/WIC/ACM WI Conf.*, Compiegne, France, 2005. ACM Press.

[28] M. Marchiori. The quest for correct information of the Web: hyper search engines. In *Proc. WWW Conf.*, Santa Clara, USA, 1997.

[29] F. Menczer. Lexical and semantic clustering by Web links. *Journal of ASIST*, 55(14):1261–1269, 2004.

[30] M. E. Newman, S. H. Strogatz, and D. J. Watts. Random graphs with arbitrary degree distributions and their applications. *Phys. Rev. E Stat. Nonlin. Soft Matter Phys.*, 64(2), 2001.

[31] L. Page, S. Brin, R. Motwani, and T. Winograd. The PageRank citation ranking: bringing order to the Web. Tech. report, Stanford University, 1998.

[32] G. Pandurangan, P. Raghavan, and E. Upfal. Using Pagerank to characterize Web structure. In *Proc. of COCOON*, vol. 2387 of *LNCS*, pp. 330–390, Singapore, 2002. Springer.

[33] W. Rudin. *Real and Complex Analysis*. McGraw-Hill, May 1986.

[34] P. Srinivasan, G. Pant, and F. Menczer. A general evaluation framework for topical crawlers. *Information Retrieval*, 8(3):417–447, 2005.

[35] D. J. Watts and S. H. Strogatz. Collective dynamics of small-world networks. *Nature*, 393(6684):440–442, June 1998.

[36] F. Wu, B. A. Huberman, L. A. Adamic, and J. R. Tyler. Information flow in social groups. *Physica A: Statistical and Theoretical Physics*, 337(1-2):327–335, June 2004.