

# Ranking Links on the Web: Search and Surf Engines

Jean-Louis Lassez<sup>1</sup>, Ryan Rossi<sup>1</sup> and Kumar Jeev<sup>2</sup>

<sup>1</sup> Coastal Carolina University, USA  
{jlassez, raross}@coastal.edu

<sup>2</sup> Johns Hopkins University, USA  
kjeev@cs.jhu.edu

**Abstract.** The main algorithms at the heart of search engines have focused on ranking and classifying sites. This is appropriate when we know what we are looking for and want it directly. Alternatively, we surf, in which case ranking and classifying links becomes the focus. We address this problem using a latent semantic analysis of the web. This technique allows us to rate, suppress or create links giving us a version of the web suitable for surfing. Furthermore, we show on benchmark examples that the performance of search algorithms such as PageRank is substantially improved as they work on an appropriately weighted graph.

**Keywords:** Search Engines, Surf Engines, Singular Value Decomposition, Heuristic Search, Intelligent Systems

## 1 Introduction

The ergodic theorem and/or its associated iterative construction of principal eigenvectors forms the backbone of the main search algorithms on the web. (PageRank[1], HITS[2], SALSA[3]). The standard proofs of the ergodic theorem rely on the Perron Frobenius theorem, which implies the use of a certain amount of mathematical machinery and restrictive hypotheses. A new fundamentally simpler proof of the ergodic theorem was derived in [4]. In a second section we will show how this proof can be used to clarify the role played by Markov models, the Perron Frobenius theorem and Kirchhoff's Matrix Tree theorem in the design of search engines. In a short third section we make a case that the ranking of links should play a major role in the design of surf engines. In the fourth section we first recall how singular value decomposition is used to extract latent semantic features[5, 6]. In the next three subsections we apply this technique to automatically rate and update links, leading to improved efficiency for search algorithms, we then generate surf sessions and extract meta sites and target sites. This construction of meta sites and targets can be used to generate hubs and authorities[2] and the bipartite graphs defined in SALSA[3] and Trawling[7].

## 2 A Symbolic View: From Kirchhoff to Google

In this section we review the results from [4] and see how they relate to PageRank, SALSA, HITS and other algorithms that form the core of search engines.

In the patent application for PageRank we find the statements: “the rank of a page can be interpreted as the probability that a surfer will be at a particular page after following a large number of forward links. The iteration circulates the probability through the linked nodes like energy flows through a circuit and accumulates in important places.” The first sentence shows that the web is considered as a Markov chain and that the ranking of sites is given as an application of the ergodic theorem[8], which indeed computes how frequently each site is visited by a surfer. The second sentence is related to Kirchhoff’s[9] current law.

The proof of the ergodic theorem is most frequently given as an application of the Perron Frobenius theorem, which essentially states that the probabilities of being at a particular site are given as the coefficients of the principal eigenvector of the stochastic matrix associated to the Markov chain, which is computed as

$$\lim_{n \rightarrow \infty} M^n e, \text{ where } e \text{ is the unit vector}$$

The implementation of the PageRank algorithm uses this construction (as a foundation, there is more to the PageRank algorithm and to the Google search engine), as well as SALSA. So we have two separate problems to consider. One is the use of the Markov Chain model for the web, and the other is the use of the Perron Frobenius theorem as a basis for an implementation. Indeed alternative constructions for computing the most frequently visited sites have been proposed for instance based on Gauss Seidel [10]. And if Kleinberg’s HITS algorithm is not based on the Markov Chain model or the ergodic theorem, it nevertheless makes systematic use of the Perron Frobenius theorem.

The ergodic theorem now plays a major role in computer science, but its complete proof is a challenge at least for undergraduate computer scientists. Indeed we have issues of convergence involving further theorems from analysis, computing eigenvalues which involves considerations of complex numbers, issues of uniqueness of solution which creates serious problems leading to restrictive hypotheses and further mathematical machinery [10].

In [11,12] it was shown that elimination theory, based on Tarski’s meta theorem could be used to derive strikingly simple proofs of important theorems whose known proofs were very involved. This technique was applied in [4] to the ergodic theorem. We informally present here the essential result that allows us to present the ergodic theorem with minimal mathematical machinery and no overly restrictive hypotheses.

Let  $G$  be a graph representing a Markov chain where the nodes  $s_i$  are called states (or sites in our application) and the edges represent links between states.

Consider the system of equations below. The  $x_i$  are the probabilities of being in state  $i$ , while  $p_{i,j}$  is the probability of moving from state  $i$  to state  $j$ . So if we are in state 2 with probability  $x_2$ , it is because we were previously in state 1 with probability  $x_1$  and we transitioned with probability  $p_{12}$ , or we were in state 4 with probability  $x_4$  and we transitioned to state 2 with probability  $p_{42}$ .

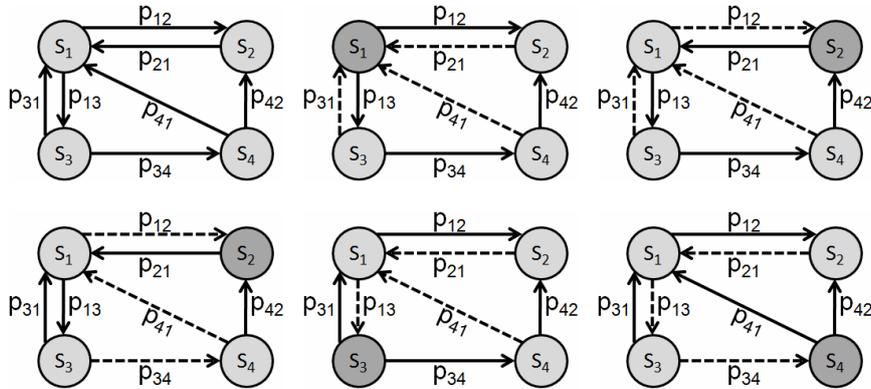
$$\begin{aligned}
p_{21}x_2 + p_{31}x_3 + p_{41}x_4 &= x_1 \\
p_{12}x_1 + p_{42}x_4 &= x_2 \\
p_{13}x_1 &= x_3 \\
p_{34}x_3 &= x_4 \\
\sum x_i &= 1
\end{aligned}$$

We solve this system by symbolic Gaussian elimination, using maple we find:

$$\begin{aligned}
x_1 &= p_{21}p_{34}p_{41} + p_{34}p_{42}p_{21} + p_{21}p_{31}p_{41} + p_{31}p_{42}p_{21} / \Sigma \\
x_2 &= p_{31}p_{41}p_{12} + p_{31}p_{42}p_{12} + p_{34}p_{41}p_{12} + p_{34}p_{42}p_{12} + p_{13}p_{34}p_{42} / \Sigma \\
x_3 &= p_{41}p_{21}p_{13} + p_{42}p_{21}p_{13} / \Sigma \\
x_4 &= p_{21}p_{13}p_{34} / \Sigma
\end{aligned}$$

$$\begin{aligned}
\Sigma &= p_{21}p_{34}p_{41} + p_{34}p_{42}p_{21} + p_{21}p_{31}p_{41} + p_{31}p_{42}p_{21} + p_{31}p_{41}p_{12} + p_{31}p_{42}p_{12} + \\
& p_{34}p_{41}p_{12} + p_{34}p_{42}p_{12} + p_{13}p_{34}p_{42} + p_{41}p_{21}p_{13} + p_{42}p_{21}p_{13} + p_{21}p_{13}p_{34}
\end{aligned}$$

A careful examination shows us that a monomial such as  $p_{42}p_{21}p_{13}$  represents a reverse weighted spanning tree with root at  $s_3$ . So we see clearly how to compute a general solution:  $x_i$  will be equal to the quotient of the sums of the monomials corresponding to the reverse weighted spanning trees rooted at  $s_i$  by the sum of the monomials corresponding to all reverse weighted spanning trees.

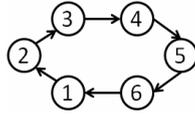


**Fig. 1.**

A simple induction will complete the proof whose details can be found in [4]. So we can present the ergodic theorem without any of the usual more complex machinery. We have the exact solution in a finite number of steps.

Also importantly we see that the only restrictive condition is that the denominator  $\Sigma$  is non null, that is we require that the graph admits at least one reverse spanning tree. That is an improvement on the use of Perron Frobenius which does not converge on a cycle as its largest eigenvalue has degree two. This causes restrictive conditions and makes the proof more complex, even though intuitively it is obvious that the solution

is a uniform value for all sites. While with our proof we see that obviously all spanning trees are isomorphic and therefore all probabilities are equal.

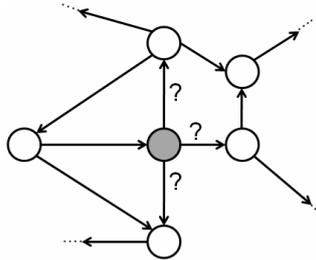


**Fig. 2.**

Furthermore, we see that if a graph has several “sinks” as described in the Google patent, then it cannot have a reverse spanning tree and the ergodic theorem will not apply. That is why in PageRank new links are added to get out of sinks, making the graph strongly connected.

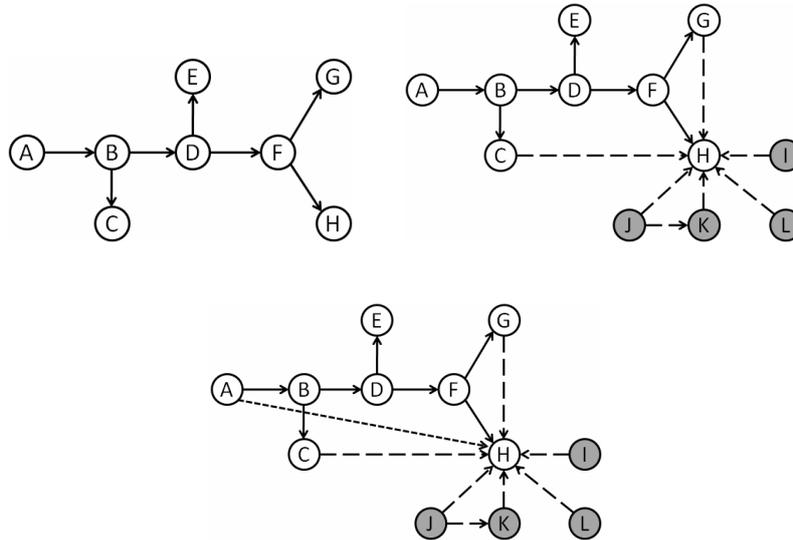
### 3 From Search Engines to Surf Engines

If at the theoretical level some algorithms assume that links are given some weights or probabilities, in practice they are given a uniform probability distribution. It is however clear that all links are not “equal” and that a weighting of these links should improve the performance of search engines. This is a local ranking as shown in figure 3, where the question is “where can I go from here?”



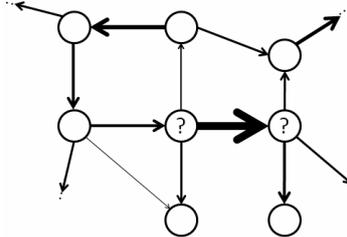
**Fig. 3.**

However if we consider the surfing aspect of search where we start a bit randomly and go window shopping on the sites, a ranking or rating of links should be of fundamental importance. This time the question is not “where can I go from here?” but rather “where *should* I go from here?” even if there is no link. This is particularly important for updating the graph, as illustrated in figure 4. The graph on the top-left represents the web at some point in time. Later new sites I,J,K,L are added with links to H and the sites C and G have updated their links adding one to H, as seen on the top-right graph. Now that H is clearly an important site, we would like to automatically update the graph, for instance by adding a link from A to H as shown in the graph at the bottom.



**Fig. 4.**

Finally an important question is raised when we rank the links globally, as opposed to locally. In figure 5 the importance of links is marked by the thickness of the arrow.



**Fig. 5.**

The question is: what is the nature of the sites which correspond to the most highly ranked links? We can expect them to be of some significance, and we will verify this at the end of the next section.

## 4 Latent Semantic Analysis of the Web

The seminal paper by Kleinberg[2] introduced the notion of Hubs and Authorities for sites on the web, and if it does not rely on the Markov Chain model, it is a remarkable application of the Perron Frobenius theorem, as it makes a double usage of its construction of principal eigenvectors.

We go one step further by considering the singular value decomposition which computes systematically all eigenvectors of  $MM^T$  and  $M^TM$ .

#### 4.1 Singular Value Decomposition

Let  $M \in \mathfrak{R}^{n \times m}$ , we decompose M into three matrices using Singular Value Decomposition:

$$M = U S V^T$$

where  $U \in \mathfrak{R}^{n \times n}$ ,  $S \in \mathfrak{R}^{n \times m}$  and  $V^T \in \mathfrak{R}^{m \times m}$ . The matrix S contains the singular values located in the  $[i, i]_{1, \dots, n}$  cells in decreasing order of magnitude and all other cells contain zero. The eigenvectors of  $MM^T$  make up the columns of U and the eigenvectors of  $M^T M$  make up the columns of V. The matrices U and V are orthogonal, unitary and span vector spaces of dimension n and m, respectively. The inverses of U and V are their transposes.

$$\begin{array}{ccc} \left[ \begin{array}{c|c|c|c} | & | & & | \\ d_1^f & d_2^f & \dots & d_k^f \\ | & | & & | \end{array} \right] & \left[ \begin{array}{cccc} s_1 & 0 & 0 & 0 \\ 0 & s_2 & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & s_k \end{array} \right] & \left[ \begin{array}{c} \text{---} d_1^c \text{---} \\ \text{---} d_2^c \text{---} \\ \vdots \\ \text{---} d_k^c \text{---} \end{array} \right] \\ U & S & V^T \end{array}$$

The columns of U are the *principal directions of the hubs* and the rows of  $V^T$  are the *principal directions of the authorities*. The principal directions are ordered according to the singular values and therefore according to the importance of their contribution to M.

The singular value decomposition is used by setting some singular values to zero, which implies that we approximate the matrix M by a matrix:

$$M_k = U_k S_k V_k^T$$

A fundamental theorem by Eckart and Young states that  $M_k$  is the closest rank-k least squares approximation of M [13]. This theorem can be used in two ways. To reduce noise by setting insignificant singular values to zero or by setting the majority of the singular values to zero and keeping only the few influential singular values in a manner similar to principal component analysis.

In latent semantic analysis we extract information about the relationships between sites as they change when we set all, but the most significant, singular values to zero. The singular values in S provide contribution scores for the principal directions in U and  $V^T$ .

We use the terminology “principal direction” for the following reason. In zoomed clusters [14] it was shown that (assuming unit vectors) the principal eigenvector is an “iterated centroid” that is a version of the notion of centroid, where outliers are given a decreasing weight. The iterative centroid is the reason Kleinberg’s HITS algorithm favors the most tightly knit communities.



$$C_\infty = \lim_{n \rightarrow \infty} (M^T M)^n e$$

The iterative centroid penalizes outliers and gives more weight or influence to the tightly knit community.

## 4.2 Automatic Rating and Updating of Links

The datasets we used in these experiments came from Panayiotis Tsaparas at the University of Toronto[15]. The queries we selected were: “Movies”, “Computational Geometry”, “Death Penalty”, “Abortion”, “Gun Control” and “Genetic.” We start with a set of web pages interconnected by links. The set of web pages can be represented as an adjacency matrix  $M$  of the hyperlink graph  $G$ , where  $M_{ij} = 1$  if there is a link from site  $i$  to site  $j$ , and  $M_{ij} = 0$  if there is no link between site  $i$  and  $j$ .

In these experiments we first compute the SVD of the adjacency matrix  $M$  from a given query and set all but five singular values to zero. We then compute  $M_5$  a low rank approximation of  $M$ , to which corresponds a new graph  $G_5$ . To this new graph, we apply the inDegree and PageRank algorithms. We count the number of relevant sites among the first ten given by the algorithms. We then compare these results with the results on the original graph  $G$ , see table 1. There is clearly a significant improvement for both algorithms. This improvement is further illustrated in tables 2 and 3, where we show the top ten results for the “Movies” query in each situation. Similar results have been obtained with the other queries.

It is important to notice that we are not following a Markov model now because the matrix  $M_5$  is not a stochastic matrix, it even can have negative numbers. It does not correspond either to a Kirchhoff model as it does not fit a conservation system. However the values in  $M_5$  represent the larger coordinates of the eigenvectors with the larger eigenvalues, and as such representative of the main directions of the graph.

**Table 1.** Quantitative relevance results before and after LSA on inDegree and PageRank.

Queries	inDegree	LSA inDegree	PageRank	LSA PageRank
Movies	6	<b>10</b>	4	<b>10</b>
Computational Geometry	8	<b>10</b>	5	<b>10</b>
Death Penalty	9	<b>9</b>	6	<b>9</b>
Abortion	10	<b>10</b>	3	<b>10</b>
Gun Control	9	<b>10</b>	7	<b>10</b>
Genetic	9	<b>9</b>	6	<b>9</b>

**Table 2.** Ranking of sites from the “Movies” query using inDegree on the original graph compared with inDegree on the appropriately weighted graph in which LSA has been applied.

Rank	inDegree	LSA inDegree
1	Hollywood.com - Your entertainment	Hollywood.com - Your entertainment
2	Paramount Pictures	Film.com & Movie Reviews,
3	Film.com & Movie Reviews,	Paramount Pictures
4	Welcome to mylifesaver.com	Universal Studios
5	Disney.com -- Where the Magic	Disney.com -- Where the Magic
6	Universal Studios	Movies.com
7	My Excite Start Page	MGM - Home Page
8	Movies.com	All Movie Guide
9	Lycos	Boxoffice Magazine
10	Google	Batman Forever Movie

**Table 3.** Ranking of sites from the “Movies” query using PageRank on the original graph compared with PageRank on the appropriately weighted graph in which LSA has been applied.

Rank	PageRank	LSA PageRank
1	GuideLive: Movies in Dallas	Hollywood.com - Your entertainment
2	CitySearch.com	Film.com & Movie Reviews,
3	On Wisconsin	Paramount Pictures
4	CDOutpost.com	Universal Studios
5	Ebert & Roper and the Movies	Disney.com -- Where the Magic
6	Roger Ebert	Movies.com
7	Sofcom Motoring	MGM - Home Page
8	Hollywood.com - Your entertainment	All Movie Guide
9	The Knoxville News	Boxoffice Magazine
10	Excite@Home: Career Opp.	Gannett home page

### 4.3 Surf Sessions

In a surf session we chose a starting site at random and follow the links with the highest weight. To generate the weights we first compute the SVD of the adjacency matrix from the movies query and set all but two hundred singular values to zero. We then compute  $M_{200}$ . We selected the site “Math in the Movies” at random and followed the link with the highest value. In this example we can check that all visited sites are relevant, we had similar cases with other random start sites, where even if a visited site did not appear to be relevant the following ones were. Such surf sessions help us validate the SVD technique that we proposed, as well as being a starting point for the design of surf engines.

**Table 4.** Surf session starting with the site “Math in the Movies” from the query “Movies”

<b>Links</b>	<b>Surf Session</b>
Start	http://world.std.com/~reinhold/mathmovies.html Math in the Movies
2	http://www.pithemovie.com Pi The Movie: Enter
3	http://www.mrcranky.com Movies and more movies!
4	http://ign.com IGN
5	http://www.allmovie.com All Movie Guide
6	http://www.chireader.com/movies Reader Guide: Movies
End	http://www.amctv.com American Movie Classics

#### 4.4 Meta sites and Targets

Assuming that a hub and a corresponding authority are of excellent quality, we could expect that they are directly linked. This is why in SALSA, Trawling and others, the relationship between hubs and authorities is given as a bipartite graph. In our setting we can expect that the links with the highest value connect sites of particular interest. Indeed we see from the tables that the sites from which the links originate are sites about sites, or meta sites, while those pointed to by the links are targets. Such sites are related to Kleinberg’s Hubs and authorities, but they might be more specific and are computed differently, so we gave them different names to avoid confusion.

**Table 5.** Meta sites and Targets for the query “Computational Geometry”

<b>Link Weight</b>	<b>Meta sites</b>	<b>Targets</b>
<b>1.92</b>	"All" Engineering Resources on the Internet	Geometry in Action
<b>1.85</b>	Computational Geometry Pages: What's ancient?	Geometry in Action
<b>1.60</b>	Computational Geometry on the WWW	Geometry in Action
<b>1.53</b>	"All" Engineering Resources on the Internet	Directory of Computational Geometry Software
<b>1.47</b>	Computational Geometry Links	Geometry in Action
<b>1.46</b>	Computational Geometry Pages: What's ancient?	Directory of Computational Geometry Software
<b>1.36</b>	Computational Geometry Web Directories	Geometry in Action
<b>1.35</b>	"All" Engineering Resources on the Internet	The former CGAL home page
<b>1.30</b>	Computational Geometry Pages: What's ancient?	The former CGAL home page

## 5 Conclusion

We have shown that the ranking of links, using singular value decomposition, can have a beneficial effect on the ranking of sites, the discovery of meta sites, and can serve as a basis for the design of surf engines.

**Acknowledgments.** Parts of this paper were presented in Sapporo, Marseille and Paris, we thank Nicholas Spryratos, Yusuru Tanaka, Mohand Said Hacid, Michele Sebag for their comments and interest.

## References

1. Page, L., Brin, S., Motwani, R., Winograd, T.: The PageRank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project, (1998)
2. Kleinberg, J.: Authoritative sources in a hyperlinked environment. In Proceedings of the 9th ACM-SIAM, SODA, (1998)
3. Lempel, R., Moran, S.: The stochastic approach for link-structure analysis (SALSA) and the TKC effect. In 9th International WWW Conference, (2000)
4. Jeev, K., Lassez, J-L.: Symbolic Stochastic Systems. AMCS, 321-328 (2004)
5. Berry, M.W., Browne, M.: Understanding Search Engines: Mathematical Modeling and Text Retrieval. SIAM, Philadelphia (2005)
6. Deerwester, S., Dumais, S., Landauer, T.K., Furnas, G., Harshman, R.: Indexing by latent semantic analysis. J. Amer. Soc. Info. Sci. 41, 391-407 (1990)
7. Kumar, S.R., Raghavan, P., Rajagopalan, S., Tomkins, A.: Trawling emerging cybercommunities automatically. In 8th International WWW Conference, (1999)
8. Марков, А.А.: Распространение закона больших чисел на величины, зависящие друг от друга. Известия Физико-математического общества при Казанском университете, 2-я серия, том 15, ст. 135-156 (1906)
9. Kirchhoff, G.: Über die Auflösung der Gleichungen, auf welche man bei der untersuchung der linearen verteilung galvanischer Ströme geführt wird. Ann. Phys. Chem. 72, 497-508 (1847)
10. Langville, A., Meyer, C.: Google's PageRank and Beyond: The science of search engine rankings. Princeton University Press, Princeton, New Jersey (2006)
11. Chandru, V., Lassez, J-L.: Qualitative Theorem Proving in Linear Constraints. Verification: Theory and Practice. 395-406 (2003)
12. Lassez, J-L.: From LP to LP: Programming with Constraints. In: Theoretical Aspects of Computer Software. LNCS, vol. 526, pp. 420-446. Springer, Heidelberg (1991)
13. Eckart, C., Young, G.: The approximation of one matrix by another of lower rank. Psychometrika. 1, 211-218 (1936)
14. Lassez, J-L., Karadeniz, T., Mukkamala, S.: Zoomed Clusters. ICONIP, 824-830 (2006)
15. Tsaparas, P.: Using non-linear dynamical systems for web searching and ranking. Principles of Database Systems, 59-69 (2004)