

Representing and Quantifying Rank - Change for the Web Graph

Akrivi Vlachou¹, Michalis Vazirgiannis^{1,2}, and Klaus Berberich³

¹Department of Informatics, Univ. of Economics and Business, Athens, Greece

²Gemo, INRIA, Paris France

³Max-Planck-Institut für Informatik, Saarbrücken, Germany
avlachou@aueb.gr, mvazirg@aueb.gr, kberberi@mpi-sb.mpg.de

Abstract. The web graph is an immensely dynamic structure continuously changing with regards to pages and links. One of the grand challenges in the last years is efficient web search inherently involving the issue of page ranking. In this paper we address the issue of representing and quantifying the web graph evolution. We study the rank position of a web page among different snapshots of the web graph and propose similarity measures of the rank change rate that are independent of the graph size. We define the *rank change rate (racer)* quantifying the web graph evolution. Thereafter, we examine different ways to aggregate the rank change rates and to identify highly dynamic web pages. In our experimental evaluation we study the dynamic of the web pages especially for pages that have a high rank position.

Key words: Pagerank, Web Graph, Web Evolution

1 Introduction

The web is a dynamic structure that is constantly changing. The evolution of the web graph is mainly caused by the changes in graph structure and in the web pages content. Every day it increases both in terms of new pages and new links that interconnect them. One of the biggest challenges is that of searching these vast amounts of data. The research area of web search inherently involves the issue of page ranking. Thus, we claim that the changes in the graph structure is of higher importance as those predominantly cause the changes in authority score and therefore of the web page ranking.

In this paper we address the issue of representing and quantifying the web graph evolution. Since a dominant issue is the ranking of pages we define the *rank change rate (racer)* quantifying the web graph evolution. We represent the evolution of a web page through a sequence of *racer* values. Our approach of *web graph representation* through the *racer* values enables (i) a concise representation of the web graph - in terms of keeping only the changes among snapshots, (ii) the possibility of as-of queries for the past via piecewise linear approximations, (iii) the identification of trends (i.e. in authority), and (iv) predictions based on

the evolution patterns extracted, for example by building a Markov model on the *racer* sequences and making predictions for future ranking values.

Thereafter, we pose the problem of finding highly dynamic pages, as those that have high *racer* values over a large period of time. In particular, we are interested in finding representative web pages that allows us to determine structural parts of the graph that change fast or set of web pages that are semantical related. Toward this goal we first propose some measurements for calculating aggregate trends in the graph and thereafter, we propose two methods to identify highly dynamic pages. To summarize, the key contributions of this paper are:

- We propose a rank normalization method and present *racer*, a formula that calculates the ranking change rate of the web pages. We generate *normalized racer* sequences which are used in order to describe the evolution of the web.
- We discuss the problem of finding the dynamic parts of the graph as those that change fast. We pose the problem of estimating the dynamism of a set of web pages over the time and propose methods to determine highly dynamic web pages.
- We present initial experiments of the proposed measurements and estimate the expressiveness of our proposed method to describe the evolution of the web graph.

2 Related Work

One of the major research areas that emerged is that of web search. The ranking of the web pages has become the most important and fundamental process in this context. Recently, new algorithms have been developed to benefit from structural information about the web. Broder et. al. [1] have shown how to design an efficient algorithmic partitioning method for certain eigenvector computations which is the key to some of the most successful search algorithms. Pagerank [2] is one of the most important algorithms used for ranking web search results and has received significant attention in the related research. The continuous crawling of the web is almost impossible due to its dynamic nature and there are several approaches to capture the dynamic of the web graph. Yang et. al. [3] propose a method called Predictive Ranking, aiming at estimating the web structure, based on the intuition that the crawling and consequently the ranking results are inaccurate (due to inadequate data and dangling pages). Another approach aiming at approximating Pagerank values without the need to perform all the computations over the entire graph is that of Chien et. al. [4]. The authors propose an efficient algorithm to incrementally compute good approximations of Pagerank, based on the evolution of the web graph's link structure. Given a set of link changes, they identify a small portion of the web graph in the vicinity of these changes, and model the rest of the web as a single node in this small graph. They subsequently compute a version of Pagerank on this small graph and transfer these results to the original graph. This approach, however, requires the continuous monitoring of the web graph in order to track any link modifications. Numerical properties of PageRank have been studied in [5]. It was observed that

the probability values produced by the PageRank algorithm decay according to a power law, and they incorporate this into a model of how the web evolves. Finally, in [6] the authors study the way web changes putting emphasis on the content, leaving out the structural change of the web which reflects essentially the evolution of pages importance. In this paper we address the aforementioned issues from a different scope.

Recent work of Dill et al. [7] provides some explanation for this self-similar behavior, i.e. that many properties of the web graph are reflected in smaller snapshots of the Web. This provides the basis for our experiments, in which we derive an understanding of the rank change rate of the Web by studying only a small subgraph of the Web graph.

3 Rank Change Rate (*racer*)

In this Section we present our proposed framework for representing the web graph evolution. We build a model which is based on the previously calculated ranking values of the web pages, for example based on the Pagerank's authority scores. We first compute, for each page and for different graph snapshots, the *rank change rate* (*racer*) using normalized ranks. Let G_{t_i} be the snapshot of the web graph at the timestamp t_i and let $n_{t_i} = |G_{t_i}|$ the number of nodes of the graph at time t_i . Assuming two snapshots of the web graph G_{t_i} and G_{t_j} and times t_i, t_j respectively with $t_i < t_j$ and $n_{t_i} \ll n_{t_j}$. For simplicity let us assume that G_{t_i} is a subset of G_{t_j} , i.e. no nodes were removed from G_{t_i} . Let p be a web page of the web graph G_{t_i} then we define $rank(p, t_i)$ as a function providing the ranking of the pages according to some criterion.

We now discuss the need for normalization of the page ranking across graph snapshots. Lets assume that there is a page p that belongs to the web graph G_{t_1} and G_{t_2} and $rank(p, t_1) = rank(p, t_2)$, then apparently the same page is much more important in the second case. For instance, assume $rank(p, t_1) = rank(p, t_2) = 5$ and $n_{t_1} = 100$, $n_{t_2} = 1000$. One would claim that the first event - page occupies the 5th out of 100 pages - is less important than the second - page occupies the (again) 5th out of 1000 pages. Thus we motivate the definition of the normalized rank - *nrank* - of a page in a ranked list. We impose that the *nrank* of all pages in a ranked list sum up to 1¹. Thus the *nrank* of a page p that occupies position $rank(p, t_1)$ in a list of $n_{t_1} \gg 1$ items is

$$nrank(p, t_i) = \frac{2 * rank(p, t_i)}{n_{t_i}^2} \quad (1)$$

Afterwards, we define rank change rate (*racer*) using the normalized ranks (*nrank*) as

$$racer(p, t_i, t_j) = \frac{nrank(p, t_i) - nrank(p, t_j)}{nrank(p, t_i)} = 1 - \frac{rank(p, t_j)}{rank(p, t_i)} * \left(\frac{n_{t_i}}{n_{t_j}}\right)^2 \quad (2)$$

¹ We omit proofs due to space limitations

Since we are interested to represent the dynamic of the web through more than one racer values, the values of different snapshots must be comparable. Thus, we define the *normalized rank change rate* (*nracer*). In order to make the racer values compatible across different graph snapshots we have to divide the racer values with its value range. To determine the racer value range we define the maximum value (*max*) as the racer value when a page goes from bottom $rank(p, t_1) = n_{t_1}$ to top $rank(p, t_2) = 1$ and the minimum value (*min*) when a page goes from top $rank(p, t_1) = 1$ to bottom $rank(p, t_2) = n_{t_2}$. Therefore normalized rank change rate (*nracer*) for page p between graph snapshots G_{t_i} and G_{t_j} is given by

$$nracer(p, t_i, t_j) = \frac{racer(p, t_i, t_j)}{max - min} \quad (3)$$

Notice that we do not use footrule or Kendal's Tau distance to identify change because they are not sensitive to the rate of change and the relevant importance of change. For example the top page falling to the 10th place and the 990th page falling to the 1000th place will be considered as of equal importance events with the foot rule distance. In Kendal's Tau case reverse pairs' ranks are equally important regardless to the magnitude of the rank disagreement.

4 Identifying highly dynamic web pages.

As initial experiments show, the vast majority of the pages remain stale across graph snapshots, while significant changes in ranking are observable only for a small fraction of pages. An important problem is to identify the structural subsets of the graphs that present a high degree of dynamism, i.e. high change rate values. Beyond structural parts of the web, there is also the need for finding the sets of web pages that are semantical related, i.e. for example based on a query term, that are highly dynamic. In a first step, the issue is the identification of highly dynamic web pages which can be used as representative web pages. In a second step, these web pages may be used to determine structural parts of the web graph or to retrieve semantical related web pages and examine their dynamism. Here we will deal with the first issue leaving the second as an open research topic.

We first define some measurements to quantify the dynamism of a set of web pages. Thereafter, we discuss about ranking aggregation which provides us the web pages with the highest rank change values over a large time period. Finally, we define the Pareto-optimal web pages that are highly dynamic and even they have not necessary the highest rank change values over a large time period, they are useful as representative web pages to determine highly dynamic sub-graphs.

4.1 Rank aggregation measurements.

By studying the values of *nracer* we observe the trend and dynamic of a particular page over time. To capture the dynamism of a set of web pages, we have

to define the rank change value between two snapshots for a set of web pages. Notice, that even if *nracer* values are calculated based on the entire graph, the proposed aggregations may consider only a subset of the graph, for example the higher ranked pages or the pages corresponding to a query result set.

A straightforward measurement of the rank change value between two snapshots for a set of web pages is to consider footrule distance between the *nrank* values.

$$fracer(G_{t_i}, G_{t_j}) = \sum_{\forall p \in G_{t_i}} |nrank(p, t_i) - nrank(p, t_j)| \quad (4)$$

To capture the dynamics of a graph over time, but also the trend, of this graph we define the aggregation of the *nracer* rank over all pages in the transition between G_{t_i} and G_{t_j} as a measurement of change among the graph snapshots:

$$aracer(G_{t_i}, G_{t_j}) = \frac{\sum_{\forall p \in G_{t_i}} nracer(p, t_i, t_j)}{n_{t_i}} \quad (5)$$

While *aracer* provides knowledge about the trend a set of web pages has, we are also interested in the dynamism of the graph in total. Thus we define *sracer* that aggregates the absolute values of the rank change rate.

$$sracer(G_{t_i}, G_{t_j}) = \frac{\sum_{\forall p \in G_{t_i}} |nracer(p, t_i, t_j)|}{n_{t_i}} \quad (6)$$

In our experimental evaluation we study the dynamics of the web using all the above mentioned measurements.

4.2 Aggregate ranking.

Aggregate ranking provide us the most dynamic web pages over a large time period that consists of more than two snapshots. Assuming a set of consecutive graph snapshots G_{t_i} , then for each pair $(G_{t_i}, G_{t_{i+1}})$ we can define a list NR_i containing the pages p in G_{t_i} and $G_{t_{i+1}}$ ranked in descending $|nracer|$ score order. Apparently the top pages in NR_i represent the most dynamic ones with regards to rank change rate while the last ones in this list are the stales ones. The objective is to aggregate the NR_i lists into a sorted list NR that best represents the pages ranked in descending order of *racer* values over the entire time period. This problem has been extensively worked out in the past [8], [9]. We use a straightforward adaption of [8] and we consider all list equal-weighted. Missing values are handled by giving them the minimum value, i.e. zero.

Based on the globally sorted list NR there are two ways to choose the most dynamic web pages: a. either to choose the k^{th} web pages with the highest position in NR , where k is a fixed parameter, b. or to choose all web pages that their aggregated score is larger than a threshold value t , which also is a fixed value. Afterwards, this set of pages can be used for defining highly dynamic structural parts of the web by exploiting their neighborhood.

4.3 Pareto-optimal web pages.

While the aggregate of the lists NR_i returns a globally sorted list, we are interested to find a set of highly dynamic web pages, as representative web pages. These web pages can be used to determine highly dynamic sub-graphs, based on the terms they contain or based on their locality.

An appropriate set of web pages is the set of the Pareto-optimal pages [10] or skyline set [11]. This operator filters out a set of interesting web pages from a potentially large set. A web page is interesting if it is not dominated by any other, i.e. is not worse than any other in all lists. In our case, a web page over a time period is considered as highly dynamic if there does not exist any other web page that has a higher *rank change rate* in all lists NR_i . Even though Pareto-optimal web pages have not necessarily the highest *rank change rate* values over a large time period, they are useful as representative web pages to determine highly dynamic sub-graphs. For example consider a web page p that has a high *nracer* value in every second list $NR_{2*i}, \forall i$ and an extremely low *nracer* in $NR_{2*i+1}, \forall i$. It is obvious that web page p is highly dynamic over some time periods, but it is quite impossible to maintain a high rank position in the globally sorted list NR . The definition of the Pareto-optimal web pages ensure us that web pages with behavior like page p will be returned as highly dynamic web pages.

Notice, that in contrast to the aggregate ranking, a parameter is not required to choose the highest dynamic pages. In our future work we plan to explore their locality to identify structural parts of the web graph that are highly dynamic.

5 Experimental Evaluation

In order to evaluate the effectiveness of the proposed approach, we performed initial experiments on real data sets. The dataset is a subset of the Internet Archive obtained from its European branch² that contains weekly crawls of eleven U.K. governmental web sites. We extracted the evolution of the corresponding part of the web graph from this dataset, yielding a total of 560,496 distinct nodes and 4,913,060 edges corresponding to web pages and interconnecting hyperlinks. Pagerank was computed on monthly snapshots of this graph, resulting in a total of 24 pre-computed rankings. All the experiments were run on low-end commodity hardware (a 3 – GHz Pentium PC with 1GB of memory and a local IDE disk, under Windows XP).

In Figure 1 we illustrate the distribution frequency of the *nracer* values with regards to the temporal distance between graph snapshots. The plot shows the number of web pages with the particular *nracer* value in logarithmic scale. From this graph we conclude: a. for consecutive graph snapshots, the vast majority of the pages (80 – 90%) improve their ranking but only marginally and b. when the temporal distance increases, the *nracer* values follow the same distribution but the peak is shifted to the right, conveying thus that web pages ranking ameliorates with time.

² An extended version of the dataset (with regard to the number of crawls) is accessible online at <http://www.europarchive.org/ukgov.php>

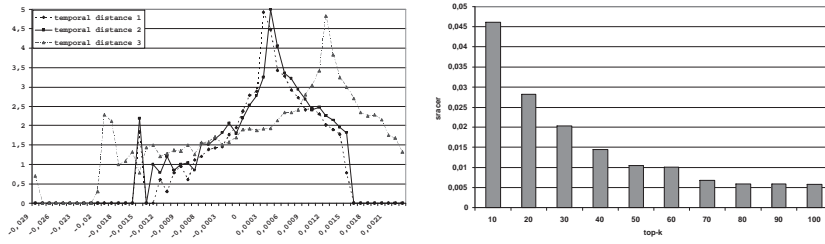


Fig. 1. a. The (log) frequency distribution of *nracer* b. *sracer* vs. size of the graph

In the next set of experiments we aim to study the dynamics of the web pages in the higher rank positions. In the first experiment we consider only the subset of the web pages that contains the k pages with the highest rank. In the left chart in Figure 1 we illustrate the *sracer* values with regard to the number of pages that we consider in each subset. As expected the *sracer* values decrease as the k increases since the percentage of pages that are removed from the set within two timestamps is higher when the set is small. This indicates that even the pages that are high ranked change within two timestamps and verifies our assumption that the web graph is a highly dynamic structure. Thereafter, we focus on the subgraph for $k = 100$ pages.

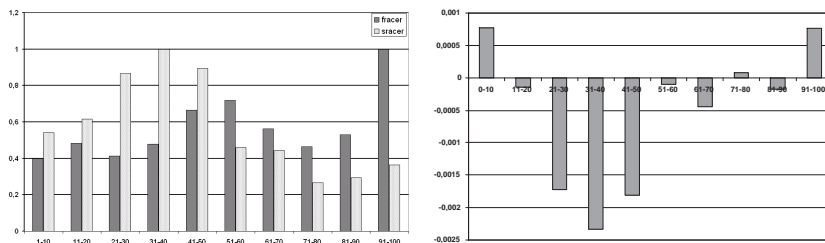


Fig. 2. a. *sracer* values vs. ranking position b. *aracer* values vs. ranking position

In the left chart in Figure 2, again we plot the *sracer* values but this time with regard to the position of the web page in the top – 100 rank. We observe that the pages at the positions 30 – 40 out of 100 have the highest *sracer* values. The right chart in Figure 2 presents the aggregate rank changes *aracer* vs the ranking position. The values of *aracer* illustrate the trend of the pages. We observe that the low and high ranked pages have positive *aracer* values while the pages that are middle-ranked have negative *aracer* values.

6 Conclusions and Future Work

The dynamics of the web provide a fascinating domain of study for researchers from academic and commercial fields. Searching in the web inherently involves the ranking issue. Assuming Pagerank as the ranking algorithm, and considering the dynamics of the web, in this paper we address the issue of representing and quantifying the web graph evolution. Thus, we define *rank change rate* (*racer*). We pose the problem of finding highly dynamic web pages and propose two appropriate methods. We conducted initial experiment with real web data evolving over time. The results are encouraging towards achieving an efficient representation of the web graph evolution. As future work we plan to achieve predictions of the ranking position, i.e. what will be the rank at time t . For this purpose we will capitalize on Markov models that are inherently suited for predictions. Furthermore, another interesting issue is to measure change topic-wise, i.e. for different queries or topics the *rank change rate* can be calculated and compared with the *rank change rate* over the whole web graph.

References

1. Broder, A., Kumar, R., Maghoul, F., Raghavan, P., Rajagopalan, S., Stata, R., Tomkins, A., Wiener, J.: Graph structure in the web. *Comput. Networks* **33**(1-6) (2000) 309–320
2. Brin, S., Page, L.: The anatomy of a large-scale hypertextual Web search engine. *Computer Networks and ISDN Systems* **30**(1–7) (1998) 107–117
3. Yang, H., King, I., Lyu, M.R.: Predictive ranking: a novel page ranking approach by estimating the web structure. In: *Proc. of the 14th Int. Conf. on World Wide Web*, New York, NY, USA, ACM Press (2005) 944–945
4. Chen, Y.Y., Gan, Q., Suel, T.: Local methods for estimating pagerank values. In: *Proc. of the 13th Int. Conf. on Information and Knowledge Management*, New York, NY, USA, ACM Press (2004) 381–389
5. Pandurangan, G., Raghavan, P., Upfal, E.: Using pagerank to characterize web structure. In: *Proc. of the 8th Annual Inter. Conf. on Computing and Combinatorics*, London, UK, Springer-Verlag (2002) 330–339
6. Fetterly, D., Manasse, M., Najork, M., Wiener, J.: A large-scale study of the evolution of web pages. In: *Proc. of the 12th Int. Conf. on World Wide Web*, New York, NY, USA, ACM Press (2003) 669–678
7. Dill, S., Kumar, R., McCurley, K.S., Rajagopalan, S., Sivakumar, D., Tomkins, A.: Self-similarity in the web. In: *Proc. of the 27th Int. Conf. on Very Large Data Bases*, San Francisco, CA, USA, Morgan Kaufmann Publishers Inc. (2001) 69–78
8. Fagin, R., Lotem, A., Naor, M.: Optimal aggregation algorithms for middleware. In: *Proc. of the 12th Symposium on Principles of Database Systems*, New York, NY, USA, ACM Press (2001) 102–113
9. Dwork, C., Kumar, R., Naor, M., Sivakumar, D.: Rank aggregation methods for the web. In: *Proc. of the 10th Int. Conf. on World Wide Web*, New York, NY, USA, ACM Press (2001) 613–622
10. Kung, H.T., Luccio, F., Preparata, F.P.: On finding the maxima of a set of vectors. *J. ACM* **22**(4) (1975) 469–476
11. Borzsonyi, S., Kossmann, D., Stocker, K.: The skyline operator. In: *IEEE Conf. on Data Engineering*, Heidelberg, Germany (2001) 421–430