

Rank Synopses for Efficient Time Travel on the Web Graph

Klaus Berberich
Max-Planck Institute for
Informatics
Saarbrücken, Germany
kberberi@mpi-inf.mpg.de

Srikanta Bedathur
Max-Planck Institute for
Informatics
Saarbrücken, Germany
bedathur@mpi-inf.mpg.de

Gerhard Weikum
Max-Planck Institute for
Informatics
Saarbrücken, Germany
weikum@mpi-inf.mpg.de

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

General Terms: Algorithms, Measurement

Keywords: Web Dynamics, Web Archive Search, Web Graph, PageRank

1. INTRODUCTION

The World Wide Web is increasingly becoming the key source of information pertaining not only to business and entertainment but also to a spectrum of sciences, culture, and politics. However, the Web has an even greater source of information within it – *evolutionary history* of its structure and content. It not only captures the evolution of digital content but embodies the near-term history of our society, economy, and science. Although efforts such as the Internet Archive [1] are archiving a large fraction of the Web, there is a serious lack of tools that are designed for the effective search over these Web archives.

Time travel queries are aimed at supporting the evolutionary (temporal) analysis over Web archives extending the power of Web search-engines. Specifically, a time travel query \mathcal{Q} is defined as a pair $\langle Q_{ir}, Q_{tc} \rangle$, where Q_{ir} is the IR-style keyword query and Q_{tc} is the target temporal context. For example, consider the following time travel query which asks for pages concerning Olympics Games 2004, $\mathcal{Q} = \langle Q_{ir} : \{ \text{“Olympic”, “Games”} \}, Q_{tc} : 15/July/2004 \rangle$. It is required that the Q_{ir} be evaluated and ranked based on the state of the archived collection as of the time instance Q_{tc} .

Effective results for such time travel queries consist of a list of pages that are ranked based on a combination of their content relevance with regard to the query terms and a query-independent measure reflecting their *authority*. Due to the high dynamics of the Web, current authority scores do not accurately reflect historical authority of Web pages. In this work, we therefore focus on reconstructing historical *PageRank* scores, a popular authority measure. The reconstructed scores can then be combined with traditional measures of content relevance such as tf-idf or OKAPI BM25 to obtain the final scores that determine the ranking of Web pages.

We first introduce a novel normalization scheme for PageRank scores that enables their comparison across instances of the Web graph at different times. Building on a time-series representation of these normalized scores, we propose a compact *Rank Synopses* structure that allows efficient reconstruction of historical PageRank scores on Web archives.

2. PAGERANK SCORE NORMALIZATION

PageRank is a well known link-based ranking technique, widely adopted both in practice and research. Given a directed graph $G(V, E)$ representing the link graph of the Web, the following for-



Node	PageRank (non-normalized)		PageRank (normalized)	
	A	B	A	B
White	0.2920	0.2186	1.7391	1.7391
Grey	0.4160	0.3115	2.4781	2.4781
Black	–	0.1257	–	1.0000

Figure 1: Sensitivity of PageRank Values ($\epsilon = 0.15$)

mula gives the PageRank $r(v)$ of a node v :

$$r(v) = (1 - \epsilon) \left(\sum_{(u,v) \in E} \frac{r(u)}{\text{out}(u)} \right) + \frac{\epsilon}{|V|} \quad (1)$$

with $\text{out}(u)$ denoting the out-degree of node u and ϵ being the probability of making a random jump (aka. damping factor).

As a consequence of its probabilistic foundation and the fact that each node is guaranteed to be visited, PageRank scores are generally *not comparable across different graphs* as the following example demonstrates. Consider the gray node in the two graphs shown in Figure 1. Intuitively, importance of neither the gray node nor the white nodes should decrease through the addition of the two black nodes, since none of these nodes are “affected” by the graph change. The PageRank scores, however, as given in the corresponding table in Figure 1 convey a decrease in the importance of the gray node and the white nodes, thus contradicting intuition. These decreases are due to the random jump inherent to PageRank that guarantees the additional black nodes to have non-zero visiting probability.

Referring to Equation 1, we can see that the PageRank score of any node in the graph is lower bounded by $r_{low} = \frac{\epsilon}{|V|}$, which is the score assigned to a node without incoming edges. However, this definition does not account for dangling nodes (i.e., nodes without any outgoing edges) – which are shown to form a significant portion of the Web graph crawled by search engines [4]. These pages are treated by making a random jump whenever the random walk enters a dangling page. Under this model, with $D \subseteq V$ denoting the set of dangling nodes, PageRank scores are lower bound by:

$$r_{low} = \frac{1}{|V|} (\epsilon + (1 - \epsilon) \sum_{d \in D} r(d))$$

which is again the score assigned to a node without incoming edges. We use this refined lower bound for normalizing the PageRank scores – for a node v its normalized PageRank score is defined as

$$\hat{r}(v) = \frac{r(v)}{r_{low}} .$$

The proposed normalization eliminates the dependence on the size of the graph with very little additional computational cost. For the earlier example, the normalized PageRank scores of the gray and the white nodes do not change as can be seen from the table in Figure 1. Further details of the normalization technique have been omitted here due to space limitations.

3. RANK SYNOPSES

At each observation of an evolving Web graph, G , one can compute PageRank scores for all nodes in the graph. For a given time series of such PageRank scores of a Web page, $\Theta = \langle (t_0, r_0), \dots, (t_n, r_n) \rangle$, a *rank synopsis* is a piecewise linear approximation given by,

$$\Phi = \langle ([s_0, e_0], \Phi_0), \dots, ([s_m, e_m], \Phi_m) \rangle .$$

Elements $([s_i, e_i], \Phi_i)$ of Φ contain a set of parameters Φ_i of the linear function that is used to approximate the time series on the time interval $[s_i, e_i]$ and are referred to as *segments* in the remainder. The segments cover the whole time period of the time series, i.e.,

$$s_0 = t_0 \wedge s_m = t_n \wedge \forall_{1 \leq i \leq m} s_i \leq e_i$$

and time intervals of subsequent segments have overlapping right and left boundaries, i.e.,

$$\forall_{1 \leq i < m} e_i = s_{i+1} .$$

Our goal is to construct a rank synopsis having a *minimum number of linear segments* while retaining a *guarantee on the approximation error* per observation. This approximation error per segment is defined as the maximal relative error made on an observation within the segment, i.e.,

$$error([s_i, e_i], \Phi_i) = \max_{t_i \in [s_i, e_i]} \left| 1 - \frac{\Phi_i(t_i)}{r_i} \right|$$

A tunable parameter θ is used as a threshold for the approximation error thus controlling the quality of the synopses fit.

An optimal rank synopsis can be computed using a dynamic programming algorithm having overall $O(n^4)$ time complexity, while a close-to-optimal rank synopsis can be generated using a greedy heuristic that reduces the time complexity to $O(n^2)$ [5]. Furthermore, close-to-optimal rank synopses can be maintained incrementally as new observations of the evolving Web graph become available.

4. EXPERIMENTAL EVALUATION

Although we used a variety of datasets for our analysis, in this paper we report results over the evolving graph obtained through the revision history of the English Wikipedia encyclopedia [2]. This dataset contains the editing history of Wikipedia spanning the time window from January 2001 to December 2005 (the time of our download). From this rich dataset we extracted a graph whose nodes correspond to articles and edges correspond to their interconnecting hyperlinks. This graph has 1,618,650 nodes and 58,845,136 edges in total. We took 60 monthly snapshots of this graph and using the popular value $\epsilon = 0.15$ as our random jump probability, we precomputed PageRank scores for each month.

Kendall's τ is used in our experiments to compare rankings. We employ the implementation provided by Boldi et al. [3] to compute Kendall's τ values reported in the experimental results. As per the definition that they have used, these scores are in the range $[-1, 1]$, with 1 (−1) indicating a perfect agreement (disagreement) of the two compared permutations.

The main utility of the rank synopses is to reconstruct the PageRank score for a given time in the past. Hence, it is important that the interpolation accuracy of the synopses be of high quality. To this end, we computed close-to-optimal rank synopses using entries for each alternate month from precomputed PageRank rankings, and interpolated the scores for left-out observation times. We report the obtained accuracy against the achieved storage compression ratio (i.e., the ratio between the amount of storage consumed by the rank synopses and the amount of storage consumed by the original rankings). Table 1 summarizes the results for different values of θ .

θ	Accuracy	Compression Ratio	Storage (in MB)
1%	0.78	0.69	108.30
2.5%	0.76	0.67	103.97
5%	0.73	0.51	78.95
10%	0.69	0.37	57.68
25%	0.61	0.25	38.59
50%	0.54	0.20	30.85

Table 1: Accuracy vs. storage on Wikipedia

We also conducted a scalability experiment to evaluate the storage advantage gained by rank synopses over storing original rankings for an increasing number of observations of the evolving graph. On the Wikipedia dataset we compute rank synopses taking only the first five, first ten, etc. observations as an input for the rank synopses computation. The amounts of storage required by the rank synopses for various values of θ and the original rankings are plotted in Figure 2.

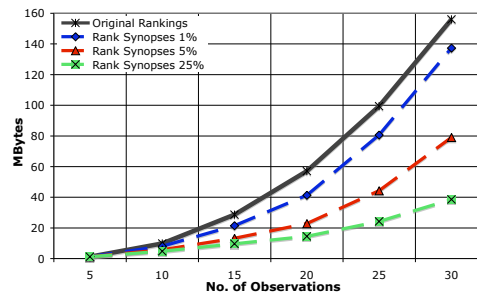


Figure 2: Scaling behavior of rank synopses on Wikipedia

The linear rank synopses, as can be seen from Figure 2, require consistently less storage than the original rankings. Apart from that, the required storage for the linear rank synopses grows modestly for all threshold values as we increase the number of precomputed rankings that are taken as an input. Thus, as we increase the number of observations from 5 to 30, the storage required by the rank synopses for the threshold value $\theta = 25\%$, for instance, increases only by a factor of 33, which is significantly less than the factor of 130 observed for the original rankings.

5. REFERENCES

- [1] Internet Archive. <http://www.archive.org>.
- [2] Wikipedia. <http://www.wikipedia.org>.
- [3] P. Boldi et al. Do your worst to make the best: Paradoxical effects in pagerank incremental computations. *WAW '04*.
- [4] N. Eiron et al. Ranking the Web Frontier. *WWW '04*.
- [5] E. J. Keogh et al. An online algorithm for segmenting time series. *ICDM '01*.