

# Mining Anchor Text Trends for Retrieval

Na Dai and Brian D. Davison

Department of Computer Science & Engineering, Lehigh University, USA  
{nad207, davison}@cse.lehigh.edu

**Abstract.** Anchor text has been considered as a useful resource to complement the representation of target pages and is broadly used in web search. However, previous research only uses anchor text of a single snapshot to improve web search. Historical trends of anchor text importance have not been well modeled in anchor text weighting strategies. In this paper, we propose a novel temporal anchor text weighting method to incorporate the trends of anchor text creation over time, which combines historical weights of anchor text by propagating the anchor text weights among snapshots over the time axis. We evaluate our method on a real-world web crawl from the Stanford WebBase. Our results demonstrate that the proposed method can produce a significant improvement in ranking quality.

## 1 Introduction

When a web page designer creates a link to another page, she will typically highlight a portion of the text of the current page, and embed it within a reference to the target page. This text is called the anchor text of the link, and usually forms a succinct description of the target page so that the reader of the current page can decide whether or not to follow the hyperlink.

Links to a target page are ostensibly created by people other than the author of the target, and thus the anchor texts likely include summaries and alternative representations of the target page content. Because these anchor texts are typically short and descriptive, they are potentially similar to queries [7] and can reflect user's information needs. Hence, anchor text has been widely utilized as an important part in ranking functions in general for commercial search engines.

Anchor text can also be important to other tasks, such as query intent classification [15], query refinement [14], query translation [17] and so on. Mining anchor text can help to better understand queries and users' information needs and enrich the representation of the linked page content.

When used for retrieval, one anchor text might not be as useful as another, and so recent work [19, 7] has focused on how to determine the importance of anchor text for a given destination page. However, such work only considers one snapshot of the web graph (the current web), and so the influence from historical anchor text is "excluded".

More importantly, the creation of anchor text reflects how web content creators view the destination page. A historical trace of the variation in such viewpoints can help determine how to interpret the page. Consider a page which has 10 newly created in-links associated with a specific anchor text in the past 3 days. When compared with another page which only received ten in-links (with the same anchor text) within the past

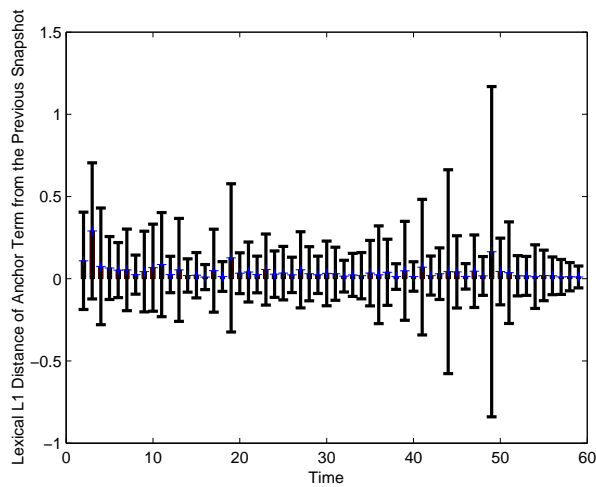
10 years, the importance of the anchor text on the former page should be emphasized, even if the absolute weights based on the current snapshot cannot differentiate them.

The primary goal of our work is to incorporate the trends of the creation of page in-links associated with anchor text into measuring the anchor text importance for representing page content in the retrieval task. We operate on the assumption that better anchor text representation of pages can improve retrieval quality. We incorporate the historical trends on anchor text by propagating the anchor text weights among historical and predicted future snapshots over the time axis. The significance of our work can be generalized onto other tasks, such as web page clustering and classification. It can also help to build time-sensitive document models. Furthermore, we propose a variety of ways to incorporate the trends from historical snapshots to better estimate the importance of anchor text in the current snapshot. Finally, we verify our models via empirical experiments, and our experiment shows the retrieval quality has significant improvement on a real-world web crawl from the Stanford WebBase.

**Related work.** Anchor text has been widely exploited in commercial search engines to improve web search. Brin and Page [3] recognized the importance of anchor text to be associated with the page to which a link points. Eiron and McCurley [8] investigated properties of anchor text in a large intranet and show its similarity with real user queries and consensus titles. It shed light on the evidence that better understanding of anchor text can help translate user queries into search results with high quality. Recent work by Fujii [9] investigated anchor text distributions on the web for query intent classification, which is used to construct an anchor text based retrieval model for web search. Anchor text can be modeled in different ways, incorporating diverse factors. Metzler et al. [19] focused on anchor text sparsity problem by incorporating anchor text associated with external links to enrich document representation. Dou et al. [7] took into consideration the relationship among anchor text which refers to the same destination page into anchor text based retrieval models to enhance the quality of search results.

Both Dou et al. [7] and Metzler et al. [19] used the retrieval model of BM25F [23], which is an extended BM25 model adaptive to fielded documents, to verify their models on anchor text empirically. The BM25F model can be further understood as smoothing destination page content by its surrogate anchor text, which is similar with the ideas of cluster-based smoothing and neighbor-based smoothing in language models. In this paper, we also use this model, but we aim to show the effectiveness of using historical information to model anchor text importance for retrieval.

Previous work has been focused on improving retrieval quality by incorporating temporal factors. Acharya et al. [1] proposed a series of temporal signals concealed in search context, which benefit web search quality. Work in [16, 12, 5] incorporated the bias from the temporal distribution of (relevant) search results into language models. Berberich et al. [2] took into account the temporal factors into link analysis algorithms. However, none of them considers to use link creation rate reflected on the anchor text weights to improve ranking quality. Our work catch the evidence of link occurrence at different time points such that we can infer the creation rate of every node and associated anchor text for every link. This enables us to analyze the anchor text importance change over time, and further use it to model anchor text weights for retrieval.



*Fig. 1: Average and deviation of the lexical  $L_1$  distance of anchor text term distribution over time for each of the top 2000 search result of the query “paris hilton”. X-axis is the time axis from early to late (from Jan. 2001 to Dec. 2005 with the time unit being 1 month). Y-axis records the average and deviation of the lexical  $L_1$  distance of destination nodes’ anchor term distribution between two successive time points.*

The remainder of this paper is as follows. We introduce the temporal anchor text data used in this work. We then start to describe our methods in Section 3, which utilizes temporal anchor text to better evaluate the importance of anchor text for retrieval. The experiments in Section 4 and 5 show the effectiveness of our approaches. We conclude this work in Section 6.

## 2 Temporal Anchor Data

A destination page gets in-links from multiple source pages at different time points, each with distinct anchor text. We assign a timestamp to each pair of source and destination page, which represents the creation time of the associated link. Naturally, we consider the item  $\langle$ source page, destination page, anchor text, creation time $\rangle$  to be unique. If the anchor text on the link changes, we assume that the link associated with the old anchor text is removed and another link associated with the new anchor text is created.

Figure 1 demonstrates the variation of the similarity of anchor text terms from month to month over a five-year time period. Earlier months show somewhat larger changes, while the changes are more moderate in later time periods. This follows as many in-links were created during the time period from 2001 to 2002. However, we also found the change in 2004 has a larger deviation. We infer that in-links have sharp increase for some destination nodes, but not for others. To better understand the fine-grained variation of anchor text on links, we keep track of how the anchor text on each link change over time. The Jaccard coefficient of anchor terms on a specific link between two successive time points is  $0.9954 \pm 0.0514$  on average. Based on these ob-

servations, we believe that the anchor text on links are relatively stable. Most anchor text does not change from the time point when the associated link was created to the time point when it was removed. The change of aggregated effects from anchor texts on destination nodes can be potentially used to benefit web search. Motivated by these observations, we propose our temporal anchor text based retrieval method.

### 3 Temporal Anchor Text based Retrieval

In this section, we describe our proposed methods which incorporate historical trends of page in-links creation rate and smooth the anchor text weights for destination pages in anchor text based retrieval. Our method requires a web graph and the time point  $t_0$  on which it is crawled. Here, we define  $t_0$  to be the current time point, and assume the retrieval evaluation is based on the situation at  $t_0$ . We follow the approach proposed by Metzler et al. [19] to determine weights on anchor text at each time point. Metzler et al. aggregated a set of unique *anchor text lines* for each given destination page, and calculated weights on them individually. However, we add extra time points to the weights on anchor text lines. Such weights on anchor text lines represent their importance on a given destination page at a specific time point. The output of our method is a collection of anchor terms and the final smoothed weights on them for a destination page at time point  $t_0$ . Specifically, our approach can be divided into the following three steps:

- aggregate anchor text lines and calculate weights on them for destination pages at each time points before  $t_0$ ;
- analyze the trend and use it to predict the possible weights on anchor text lines at the time points after  $t_0$ ;
- propagate and diffuse the weights on anchor text lines through the time axis;

#### 3.1 Aggregate Historical Anchor Text

In order to better understand how to collect and weight the aggregated historical anchor text, we first describe how we weight the anchor text of the current snapshot. We use the methods in [19] to collect and weight anchor text for a specific web snapshot. Indeed, there are other ways to weight anchor text beyond the methods presented in [19]. However, [19] is the only one to deal with *anchor text sparsity* problem. Considering the fact that historical link information may sometimes be unavailable, to better enrich the anchor text representation becomes a good choice. We now briefly review the way of collecting and weighting anchor text in that work.

Given a URL  $u$ , all in-link pages  $P$  that are within the same site(domain) as  $u$  are collected as *internal links*. Those in-link pages  $A$  that are in different domains from  $u$  are defined as *external links*. The anchor text on the external links are called *original anchor text*. For internal links, we further collect the external links of these internal links. The anchor text on the new collected external links are known as *aggregated anchor text* of  $u$ . The original anchor text are weighted as follows:

$$wt(a, u) = \sum_{s \in S(u)} \frac{\delta(a, u, s)}{|anchors(u, s)|}$$

where  $S(u)$  is the set of external sites that links to  $u$ ,  $\delta(a, u, s)$  is 1 iff anchor text line  $a$  links to  $u$  from site  $s$ . The aggregated anchor text are weighted in multiple ways,

we choose two of them which are shown to have best performance in general in [19], defined as follows:

$$wt_{Min}(a, u) = \min_{u' \in N(u)} wt(a, u') \quad wt_{Max}(a, u) = \max_{u' \in N(u)} wt(a, u')$$

where  $N(u)$  is the set of internal in-links and  $wt(a, u')$  is the original weight of anchor text line  $a$  for URL  $u'$ .

Both original anchor text lines and external anchor text lines are used to enrich anchor text representation. We choose to use *combined* representation and *back off* representation to enrich destination pages' representation. Combined representation keeps the document structure and augments both original anchor text and aggregated anchor text, whereas back off representation exempts from the aggregated anchor text which have already appeared in original anchor text lines.

Once we have weights on anchor texts at the current time point  $t_0$ , we have actually known which links should contribute to anchor text weights. We keep track of these links by looking back to seek their creation time (see Section 3.4 for details). We define the difference of two successive time points as  $\Delta t$ , i.e.,  $\Delta t = t_i - t_{i-1}$ . We map each link onto the time axis according to its creation time. If link  $l$  is created before  $t_i$  but after  $t_{i-1}$ , then for any given time point  $t_j$  ( $0 > j \geq i$ ),  $l$  is included in the snapshot at  $t_j$ . Given any time point  $t_i$  ( $i < 0$ ), we calculate the weight  $w_i(a, u)$  of anchor text line  $a$  on the web page  $u$  based on all the links included at time point  $t_i$ .

### 3.2 Quantify Trends to Predict Future

Quantifying trends of weights on anchor text can help to predict how the weights change at future time points. Given a destination page, if the importance of its particular anchor text increases more greatly than its other anchor text, we may have higher confidence to believe such anchor text should be emphasized in some way since the trend shows it may get a higher weight in the near future. Here, we assume that: (1) newly created in-links prefer to associate with the anchor text used by recent in-links, according to their popularity, and (2) the weights on anchor text have reflected the number of page/site pointing to the target page, using that anchor text.

ARIMA (Auto-Regressive Integrated Moving Average) is a powerful way to predict time-series, but it is complex to use. Instead, we use the linear regression on moving average of order  $m$  to predict the value at the next time point. The reasons are as follows: (1) we observe that weights on anchor text have stable and monotone trends through time once the anchor text begins associating with the destination page; (2) we test the linear regression on weights (Max+combined) of individual anchor text line for a given destination page over time axis. The average mean square error (MSE) is only 0.0656. Based on these observations, we believe the linear model can well fit the trends of historical anchor text weights.

Given a URL  $u$  and one associated anchor text line  $a$ , we have a series of historical weights  $w_{-n}(a, u), w_{-n+1}(a, u), \dots, w_0(a, u)$ . We first use a sliding window with size  $2k+1$  ( $k > 0$ ) to smooth the time series. We calculate a moving average of order  $2k+1$  as the following sequence of arithmetic means:

$$\frac{\sum_{i=-n}^{-n+2k} w_i(a, u)}{2k+1}, \frac{\sum_{i=-n+1}^{-n+2k+1} w_i(a, u)}{2k+1}, \dots, \frac{\sum_{i=-2k}^0 w_i(a, u)}{2k+1}$$

By using the sequence calculated above, we next use linear regression to predict the possible average at time point  $(-k + 2)$ . The model assumes the moving average of order  $2k + 1$  has a linear relationship with the time points given a pair of anchor text and destination page, which is given by:

$$\bar{w}_i(a, u) = b + c \times i + \epsilon_i, i \geq (-k + 2) \quad (1)$$

See [22] for deep analysis of the non-stationary correlated errors  $\epsilon_i$ . We use existing evidence to estimate the parameters  $b$  and  $c$ . Once the weight  $\hat{w}_{-k+2}(a, u)$  is achieved,  $w_1(a, u)$  can be calculated by:

$$w_1(a, u) = \hat{w}_{-k+2}(a, u) \times (2k + 1) - \sum_{i=-2k+1}^0 w_i(a, u)$$

After we get the value of  $w_1(a, u)$ , we move the sliding window forward to calculate  $w_i(a, u)(i > 1)$ .

### 3.3 Diffuse Temporal Anchor Text Weights

Analyzing the trends of anchor text weights on a destination page allows us to predict the anchor text weights in the future. However, in order to better measure the importance of anchor text lines at  $t_0$ , we need to combine both the predicted future weights and the historical weights. As discussed in previous section, the predicted weights are extrapolated from historical trends, which help to differentiate two anchor text lines with the same weights at  $t_0$ . On the other hand, historical anchor text weights provide confirmation about what a destination page looks like. When we emphasize the predicted future weights, we actually give preference to the newly created destination pages, since the new pages tend to have higher anchor text creation rate, and the predicted anchor text weights are usually overemphasized. Whereas, when we combine some historical weights, we likely emphasize old pages which have stable anchor text distribution. By combining both the historical weights and predicted future weights, we can harmonize the influence from these two sides.

Specifically, we assume that the weights on an anchor text line at one time point can propagate through time axis to influence the weights of the same anchor text line at other time points for a given destination page. The intuition is that if an anchor text have a weight at a time point  $t_i$ , we hope it can influence the weights on the same anchor text at other time points by a decayed way which is proportional to a temporal distance. Thus, weights on two close time points would have more influence to each other than those on two far time points. Furthermore, we assume that the change ratio of the destination page content will also influence the weight propagation since huge change is likely to cause such propagation decayed more quickly. Given a time window, we calculate weights at the middle time point by aggregating the discount weights from all time points within it.

We now describe our method to propagate the weights formally. Let  $\gamma$  be the size of time window  $T$ , i.e., the number of time points within the time window. Let  $a$  be an anchor text line. Let  $u$  be a destination node, and  $u_i$  be a destination node at time point  $t_i$ .  $w_1(u, a), w_2(u, a), \dots, w_\gamma(u, a)$  are the weights of  $a$  on  $u$  at time points within the

time window  $T$ . The weights at time point  $t_{\frac{\gamma}{2}}$  after combining the propagated weights other time points within the time window is given by:

$$w'_{\frac{\gamma}{2}}(u, a) = \sum_{i=1}^{\gamma} f(u, \gamma, i) w_i(u, a) \quad (2)$$

where  $f(u, \gamma, i)$  is the kernel function which determines the way of combining weight  $w(u, a)$  at time point  $t_i$ .

Enlightened by previous work [6, 13, 21, 18] which used proximity-based methods, we use five modified kernel functions originated from Gaussian kernel (equation 3), Triangle kernel (equation 4), Cosine kernel (equation 5), Circle kernel (equation 6), and Rectangle kernel (equation 7), which are defined by:

$$f_1(u, \gamma, i) = \exp\left[-\frac{1}{2} \left(\frac{i - \frac{\gamma}{2}}{\gamma(1 + \overline{B}_u(i \leftrightarrow \frac{\gamma}{2}))}\right)^2\right] \quad (3)$$

$$f_2(u, \gamma, i) = 1 - \frac{|i - \frac{\gamma}{2}|}{\gamma(1 + \overline{B}_u(i \leftrightarrow \frac{\gamma}{2}))} \quad (4)$$

$$f_3(u, \gamma, i) = \frac{1}{2} \left[1 + \cos\left(\frac{\pi(i - \frac{\gamma}{2})}{\gamma(1 + \overline{B}_u(i \leftrightarrow \frac{\gamma}{2}))}\right)\right] \quad (5)$$

$$f_4(u, \gamma, i) = \sqrt{1 - \left(\frac{|i - \gamma/2|}{\gamma(1 + \overline{B}_u(i \leftrightarrow \frac{\gamma}{2}))}\right)^2} \quad (6)$$

$$f_5(u, \gamma, i) = 1 \quad (7)$$

where  $\overline{B}_u(i \leftrightarrow \frac{\gamma}{2})$  is the average similarity between the destination page  $u$ 's content at two successive time points within the scale  $[i, \gamma/2]$  if  $i < \gamma/2$  or  $[\gamma/2, i]$  if  $i \geq \gamma/2$ . Without loss of generality, we assume  $i < \gamma/2$ .  $\overline{B}_u(i \leftrightarrow \frac{\gamma}{2})$  is defined by:

$$\overline{B}_u(i \leftrightarrow \frac{\gamma}{2}) = \frac{1}{\frac{\gamma}{2} - i} \sum_{i'=i}^{\frac{\gamma}{2}-1} B_u(i', i' + 1) \quad (8)$$

We compare the similarity of two snapshots of page  $u$ 's content by comparing their associated language models via the Bhattacharyya correlation:

$$B_u(i', i' + 1) = \sum_{v \in V} \sqrt{P(w|\theta_{u_{i'}})P(w|\theta_{u_{i'+1}})} \quad (9)$$

This metric renders a similarity score between 0 and 1. Although this similarity is only based on  $P(w|\theta_u)$ , we can consider to combine other measures based on topic, times-tamp, or out-link overlap so that all these measures can influence the probability of propagating the anchor text importance through the time axis.

### 3.4 Implementation

One key problem for utilizing temporal anchor text is that it is difficult to keep track of the information about when a link was created. Given a link appearing in the current snapshot, we looked back to the archival copies of the source page via the Wayback Machine portal of Internet Archive [10]. We parsed these copies to get all out-links within this page, and checked whether the given link was still in the out-link collection and whether the anchor text associated with the given link had any change. If either the anchor text has changed or the link did not exist, we utilized the timestamp of the next latest copy to be the time when the given link was created.

## 4 Experiment Setup

### 4.1 Data set and Evaluation

Although many datasets, such as TREC .GOV collection [20], have been built for research purposes, they are usually small and biased, and cannot represent the characteristics of the real-world web graph. Hence, we choose to use a May 2005 crawl from the Stanford WebBase [4] as our dataset for ranking evaluation. This crawl has 58 million pages, and approximately 900 million links.

For ranking evaluation, 50 queries are selected from a set of consisting of those frequently used by previous researchers, ODP category names, and popular queries from Lycos and Google. For each query, we have judgment of 35 URLs on average. When human editors (members of our research lab) judge each pair of <query, URL>, they are asked to give a score based on how relevant the URL is to the given query. The rating results in the selection among excellent, good, not sure, bad, and worse. We use five-value scale which translates the ratings into the integers from 4 to 0. If the average score for this pair is more than 2.5, it is marked as relevant.

Based on the available relevance judgments, we evaluate the retrieval quality of our ranking algorithms over the Normalized Discounted Cumulative Gain (NDCG) [11] metric. NDCG credits ranking list with high precision at top ranks by weighting relevant documents according to their positions in the returned search results. Precision@10 is also used to show ranking performance, which calculates the average fraction of relevant URLs within the top ten results across all queries.

### 4.2 Ranking Function

Combining different fields of web pages has been shown highly effective for retrieval on the web in previous work [24]. BM25F is such a ranking model, which combines term frequencies in different fields linearly for BM25 score calculation. In this work, we test our anchor text weighting strategies by combining body text and anchor text in the BM25F model for retrieval. Suppose  $w_{body}(i, j)$  is the weight of term  $i$  for page  $j$  in the body field, i.e., the term frequency of term  $i$  in page  $j$ . Let  $w_{anchor}(i, j)$  be the weight of term  $i$  in the anchor text lines associated with page  $j$ , which is calculated by:

$$w_{anchor}(i, j) = \sum_{a \in A(j)} wt(a, j) \times tf_{anchor}(i, a)$$

where  $wt(a, j)$  is the weight on anchor text line  $a$  for the page  $j$ , and  $tf_{anchor}(i, a)$  is the term frequency of  $i$  in the anchor text line  $a$ .



Table 1: Performance comparison for different windows and different anchor text representations. The † and ‡ symbols demonstrate the performance has statistically significant improvement when compared with the baseline (Latest anchors) at the level of  $p < 0.1$  and  $p < 0.05$  by one-tailed student t test.

| Baseline               |         |         |         |         |                |         |         |         |         |
|------------------------|---------|---------|---------|---------|----------------|---------|---------|---------|---------|
|                        | P@10    | NDCG@3  | NDCG@5  | NDCG@10 |                | P@10    | NDCG@3  | NDCG@5  | NDCG@10 |
| No anchors             | 1.6150  | 0.1860  | 0.1830  | 0.1749  | Latest anchors | 1.6170  | 0.1899  | 0.1846  | 0.1781  |
| All historical anchors | 1.6596  | 0.2023  | 0.1901  | 0.1856  |                | -       |         |         |         |
| Backoff+Max            |         |         |         |         | Combined+Max   |         |         |         |         |
| Window                 | P@10    | NDCG@3  | NDCG@5  | NDCG@10 | Window         | P@10    | NDCG@3  | NDCG@5  | NDCG@10 |
| 1                      | 1.6383† | 0.2019‡ | 0.1911† | 0.1822† | 1              | 1.6383† | 0.2019‡ | 0.1889  | 0.1841‡ |
| 2                      | 1.6383† | 0.2064‡ | 0.1945† | 0.1858‡ | 2              | 1.6809‡ | 0.2064‡ | 0.1935‡ | 0.1889‡ |
| 4                      | 1.6809‡ | 0.2064‡ | 0.1945† | 0.1879‡ | 4              | 1.7234‡ | 0.2064‡ | 0.1951‡ | 0.1909‡ |
| 7                      | 1.7234‡ | 0.2076‡ | 0.1984‡ | 0.1915‡ | 7              | 1.7660‡ | 0.2094‡ | 0.1972‡ | 0.1944‡ |
| 12                     | 1.7234‡ | 0.2085‡ | 0.1990‡ | 0.1916‡ | 12             | 1.7660‡ | 0.2105‡ | 0.1980‡ | 0.1964‡ |
| 24                     | 1.7660‡ | 0.2086‡ | 0.2002‡ | 0.1950‡ | 24             | 1.8298‡ | 0.2129‡ | 0.2025‡ | 0.2003‡ |
| Backoff+Min            |         |         |         |         | Combined+Min   |         |         |         |         |
| Window                 | P@10    | NDCG@3  | NDCG@5  | NDCG@10 | Window         | P@10    | NDCG@3  | NDCG@5  | NDCG@10 |
| 1                      | 1.6170  | 0.1956† | 0.1901† | 0.1813† | 1              | 1.6170  | 0.1956† | 0.1875  | 0.1830‡ |
| 2                      | 1.6170  | 0.2024‡ | 0.1913† | 0.1829† | 2              | 1.6809‡ | 0.1994† | 0.1902† | 0.1875‡ |
| 4                      | 1.6596‡ | 0.2050‡ | 0.1921† | 0.1853‡ | 4              | 1.7234‡ | 0.2033‡ | 0.1941‡ | 0.1899‡ |
| 7                      | 1.7021‡ | 0.2063‡ | 0.1979‡ | 0.1892‡ | 7              | 1.7660‡ | 0.2081‡ | 0.1963‡ | 0.1937‡ |
| 12                     | 1.7234‡ | 0.2072‡ | 0.1975‡ | 0.1909‡ | 12             | 1.7660‡ | 0.2092‡ | 0.1980‡ | 0.1958‡ |
| 24                     | 1.7660‡ | 0.2073‡ | 0.1990‡ | 0.1943‡ | 24             | 1.8298‡ | 0.2115‡ | 0.2015‡ | 0.1996‡ |

The aggregated term weights on  $i$  is a linear combination of weights  $i$  on anchor text and page body, which is given by:

$$w(i, j) = (1 - \alpha) \times w_{anchor}(i, j) + \alpha \times w_{body}(i, j)$$

where  $\alpha$  is a combination parameter, which controls the balance between term weights on anchor text and page body used in BM25F ranking function. The document length is calculated by the same method.

## 5 Experimental Results

In this section, we describe the results of the experimental ranking evaluation. We start by showing how much the proposed ranking algorithms significantly improve the retrieval quality. We then render some deeper analysis about the characteristics of this ranking algorithms with respect to the improvement of ranking quality.

### 5.1 Performance Comparison

The first experiment shows the effectiveness of enlarging the window for propagating historical weights on anchor text lines over multiple aggregation functions and anchor text representation. The results are reported in Table 1. The baseline is the best performance by using the methods in [19] (on the latest snapshots). Since we remove all inlinks if there is no historical snapshots for the inlinked nodes in the Internet Archive, the improvement of using anchor text versus without using anchor text is not obvious. The performance of almost all combinations of window sizes, aggregation functions and document representation over all the metrics outperform the baseline significantly. Furthermore, the performance of all combinations of aggregation functions consistently increases with the window size, which indicates the temporal inlinks, especially long

term historical inlink context is a good resource to reflect the link evolution that can be utilized in improving the ranking quality in terms of document relevance. Furthermore, the combined aggregation functions outperform the Backoff approaches, which suggests that the benefits from the “confirmation” influence brought by duplicate anchor text lines outweigh the noise they introduced.

## 5.2 Analysis

We now describe the effectiveness of kernel functions used in propagating anchor text line weights. The results are rendered in Table 2. The performance of Rectangle kernel is arguable the best in general among all combinations of aggregation functions. Gaussian and Circle kernels show comparable performance, which outperform Triangle and Cosine kernels. This observation demonstrates that search results benefit from emphasizing both historical and predicted future anchor weights without deemphasizing the influence of time points far away from the current point. We infer that the ranking quality will benefit from long-term temporal information rather than short-term one since long-term information tends to express more stable trends.

Table 2: Performance comparison for different kernels for propagating temporal anchor line weights when the window size is 12. The kernel 1, 2, 3, 4, and 5 represent Gaussian kernel, Triangle kernel, Cosine kernel, Circle kernel, and Rectangle kernel respectively.

| Baseline    |               |               |               |               |                |               |               |               |               |
|-------------|---------------|---------------|---------------|---------------|----------------|---------------|---------------|---------------|---------------|
|             | P@10          | NDCG@3        | NDCG@5        | NDCG@10       |                | P@10          | NDCG@3        | NDCG@5        | NDCG@10       |
| No anchors  | 1.6150        | 0.1860        | 0.1830        | 0.1749        | Latest anchors | 1.6170        | 0.1899        | 0.1846        | 0.1781        |
| Backoff+Max |               |               |               |               | Combined+Max   |               |               |               |               |
| Kernel      | P@10          | NDCG@3        | NDCG@5        | NDCG@10       | Kernel         | P@10          | NDCG@3        | NDCG@5        | NDCG@10       |
| 1           | 1.7022        | <b>0.2085</b> | 0.1962        | 0.1897        | 1              | 1.7457        | <b>0.2105</b> | <b>0.1980</b> | 0.1940        |
| 2           | 1.7021        | 0.2044        | 0.1955        | 0.1900        | 2              | 1.7447        | 0.2063        | 0.1955        | 0.1931        |
| 3           | 1.7020        | 0.2044        | 0.1955        | 0.1900        | 3              | 1.7447        | 0.2063        | 0.1955        | 0.1931        |
| 4           | <b>1.7234</b> | 0.2063        | 0.1985        | 0.1899        | 4              | <b>1.7660</b> | 0.2057        | 0.1966        | 0.1930        |
| 5           | 1.7023        | 0.2050        | <b>0.1990</b> | <b>0.1916</b> | 5              | <b>1.7660</b> | 0.2068        | 0.1977        | <b>0.1964</b> |
| Backoff+Min |               |               |               |               | Combined+Min   |               |               |               |               |
| Kernel      | P@10          | NDCG@3        | NDCG@5        | NDCG@10       | Kernel         | P@10          | NDCG@3        | NDCG@5        | NDCG@10       |
| 1           | 1.7021        | <b>0.2072</b> | 0.1953        | 0.1890        | 1              | 1.7447        | 0.2086        | <b>0.1980</b> | 0.1933        |
| 2           | 1.7019        | 0.2030        | 0.1950        | 0.1889        | 2              | 1.7438        | 0.2050        | 0.1946        | 0.1920        |
| 3           | 1.7019        | 0.2030        | 0.1950        | 0.1889        | 3              | 1.7438        | 0.2050        | 0.1946        | 0.1920        |
| 4           | <b>1.7234</b> | 0.2050        | 0.1955        | 0.1891        | 4              | <b>1.7660</b> | <b>0.2092</b> | 0.1957        | 0.1924        |
| 5           | 1.7021        | 0.2037        | <b>0.1975</b> | <b>0.1909</b> | 5              | <b>1.7660</b> | 0.2055        | 0.1967        | <b>0.1958</b> |

Next, we investigate the relationship between the average age of search results and the relative improvement of ranking quality in Table 3. We bucketize the queries according to the average age of their top 2000 search results. Bucket 0 is the one, in which query results have the short average age, and bucket 3 is the one in which query results have the longest average age. From Table 3, query results with longer ages benefit more by propagating anchor text weights from past time points, whereas the query results with shorter ages have better improvements by propagating predicted weights from future time points over all window sizes. By combining the weights on both past time points and future time points, the relative improvement is greater than only combining weights in one direction for most buckets in different window sizes.

Table 3: Performance comparison for queries bucketized by the average age of search results. The weighting strategy is Combined+Max. P: Propagating weights on anchor text lines from past time points; F: Propagating predicted weights on anchor text lines from future points; T: Propagating weights on anchor text lines from both sides.

| Window | Time | Bucket 0 | Bucket 1 | Bucket 2 | Bucket 3 | Window | Time | Bucket 0 | Bucket 1 | Bucket 2 | Bucket 3 |
|--------|------|----------|----------|----------|----------|--------|------|----------|----------|----------|----------|
| 1      | P    | 4.00%    | 2.72%    | 2.05%    | 2.92%    | 7      | P    | 5.92%    | 7.11%    | 2.88%    | 4.58%    |
|        | F    | 4.00%    | 2.72%    | 2.05%    | 2.92%    |        | F    | 7.24%    | 6.46%    | 3.04%    | 4.07%    |
|        | T    | 7.79%    | 2.67%    | 2.20%    | 3.54%    |        | T    | 8.93%    | 7.18%    | 3.14%    | 4.27%    |
| 2      | P    | 7.79%    | 2.75%    | 2.20%    | 3.54%    | 12     | P    | 6.11%    | 4.78%    | 3.01%    | 4.64%    |
|        | F    | 7.79%    | 2.67%    | 2.20%    | 3.54%    |        | F    | 8.08%    | 7.02%    | 3.03%    | 4.27%    |
|        | T    | 7.01%    | 6.44%    | 2.70%    | 4.27%    |        | T    | 12.04%   | 6.18%    | 2.69%    | 4.27%    |
| 4      | P    | 7.12%    | 6.32%    | 2.70%    | 4.27%    | 24     | P    | 5.83%    | 2.88%    | 0.85%    | 5.05%    |
|        | F    | 8.82%    | 6.44%    | 2.63%    | 4.03%    |        | F    | 12.76%   | 7.46%    | 2.75%    | 4.27%    |
|        | T    | 8.80%    | 7.12%    | 2.83%    | 4.27%    |        | T    | 11.04%   | 5.30%    | 2.15%    | 3.92%    |

## 6 Conclusion

The dynamic page in-links and associated anchor text reflect how other pages view destination page changes over time. However, the ever-changing weights on anchor text, as an indicator of the change of anchor text importance, is seldom used for web search, partly because such information is not available. In this paper, we utilize the historical archival copies of web pages provided by the Internet Archive (a public resource) to investigate the way to benefit web search. We propose new methods to quantify anchor text importance, which is motivated by differentiating pages with different in-link creation rate over time and different historical in-link context. Evaluation experiments on one crawl of Stanford WebBase show the ranking performance of our proposed methods has more than 10% improvement over a baseline without considering historical information.

From this work, we realize that the current archival web pages only cover a small portion of the historical web, which causes a large amount of missing anchors (only 2.57% anchors have archival copies in our data set) and thus limits the application of the proposed method. Furthermore, the crawling policies used to collect these archival web page copies might not accurately records the trace of web activities. However, as an initial work, our results unraveled that with enough historical information for pages on the web, we can potentially give more accurate estimation about anchor text importance and page in-link importance to improve web search. As future work, we hope to find a way to mitigate the gap caused by the anchors having archival copies and those without in searching process, such that the method can be completely applicable.

### Acknowledgments

This work was supported in part by a grant from the National Science Foundation under award IIS-0803605 and an equipment grant from Sun Microsystems.

### References

1. A. Acharya, M. Cutts, J. Dean, P. Haahr, M. Henzinger, U. Hoelzle, S. Lawrence, K. Pflieger, O. Sercinoglu, and S. Tong. Information retrieval based on historical data. US Patent 7346839, March 2008.
2. K. Berberich, M. Vazirgiannis, and G. Weikum. Time-aware authority ranking. *Algorithms and Models for the Web-Graph*, 3243/2004(3):32, 2004.

3. S. Brin and L. Page. The anatomy of a large-scale hypertextual Web search engine. In *Proc. of the 7th Int'l World Wide Web Conf.*, Apr. 1998.
4. J. Cho, H. Garcia-Molina, T. Haveliwala, W. Lam, A. Paepcke, S. Raghavan, and G. Wesley. Stanford WebBase components and applications. *ACM Trans. on Internet Technology*, 6(2):153–186, 2006.
5. W. Dakka, L. Gravano, and P. G. Ipeirotis. Answering general time sensitive queries. In *Proc. of the 17th ACM Conf. on Information and Knowledge Management (CIKM)*, pages 1437–1438, New York, NY, 2008. ACM.
6. O. de Kretser and A. Moffat. Effective document presentation with a locality-based similarity heuristic. In *Proc. 22nd Annual Int'l ACM SIGIR Conf. on Research and Dev. in Information Retrieval*, July 1999.
7. Z. Dou, R. Song, J.-Y. Nie, and J.-R. Wen. Using anchor texts with their hyperlink structure for web search. In *Proc. 32nd Annual Int'l ACM SIGIR Conf. on Research and Dev. in Information Retrieval*, July 2009.
8. N. Eiron and K. S. McCurley. Analysis of anchor text for web search. In *Proc. 26th Annual Intl ACM SIGIR Conf. on Research and Dev. in Information Retrieval*, July 2003.
9. A. Fujii. Modeling anchor text and classifying queries to enhance web document retrieval. In *Proc. of the 17th Int'l World Wide Web Conf.*, Apr 2008.
10. Internet Archive. The Internet Archive. <http://www.archive.org/>, 2009.
11. K. Jarvelin and J. Kekalainen. IR evaluation methods for retrieving highly relevant documents. In *Proc. of the 23rd Annual Int'l ACM SIGIR Conf. on Research and Development in Information Retrieval*, pages 41–48, July 2000.
12. R. Jones and F. Diaz. Temporal profiles of queries. *ACM Trans. Inf. Syst.*, 25(3):14, 2007.
13. K. Kise, M. Junker, A. Dengel, and K. Matsumoto. Passage retrieval based on density distributions of terms and its applications to document retrieval and question answering. In *Reading and Learning*, volume 2956 of *LNCS*, Berlin/Heidelberg, Apr. 2004. Springer.
14. R. Kraft and J. Zien. Mining anchor text for query refinement. In *Proc. of the 13th Int'l World Wide Web Conf.*, pages 666–674, New York, NY, USA, 2004. ACM.
15. U. Lee, Z. Liu, and J. Cho. Automatic identification of user goals in web search. In *Proc. of the 14th Int'l World Wide Web Conf.*, pages 391–400, New York, NY, 2005. ACM Press.
16. X. Li and W. B. Croft. Time-based language models. In *Proc. of the 20th Int'l Conf. on Information and Knowledge Management*, pages 469–475, New York, NY, USA, 2003. ACM.
17. W.-H. Lu, L.-F. Chien, and H.-J. Lee. Anchor text mining for translation of web queries: A transitive translation approach. *ACM Trans. Inf. Syst.*, 22(2):242–269, 2004.
18. Y. Lv and C. Zhai. Positional language models for information retrieval. In *Proc. 32nd Annual Int'l ACM SIGIR Conf. on Research and Dev. in Information Retrieval*, July 2009.
19. D. Metzler, J. Novak, H. Cui, and S. Reddy. Building enriched document representations using aggregated anchor text. In *Proc. 32nd Annual Int'l ACM SIGIR Conf. on Research and Dev. in Information Retrieval*, July 2009.
20. NIST. Text REtrieval Conference (TREC) home page. <http://trec.nist.gov/>, 2008.
21. D. Petkova and W. B. Croft. Proximity-based document representation for named entity retrieval. In *Proc. of the 16th ACM Conf. on information and knowledge management*, pages 731–740, New York, NY, USA, 2007. ACM.
22. S. S. Rao. On multiple regression models with nonstationary correlated errors. *Biometrika*, 91(3):645–659, 2004.
23. S. Robertson, H. Zaragoza, and M. Taylor. Simple bm25 extension to multiple weighted fields. In *Proc. of the 13th ACM Int'l Conf. on Information and Knowledge Management (CIKM)*, pages 42–49, New York, NY, USA, 2004. ACM.
24. H. Zaragoza, N. Craswell, M. Taylor, S. Saria, and S. Robertson. Microsoft Cambridge at trec 13: Web and hard tracks. In *TREC '13: Proceedings of the thirteenth Text REtrieval Conference*, 2004.