

Link-Contexts for Ranking

Jessica Gronski

University of California Santa Cruz
jgronski@soe.ucsc.edu

Abstract. Anchor text has been shown to be effective in ranking[6] and a variety of information retrieval tasks on web pages. Some authors have expanded on anchor text by using the words around the anchor tag, a *link-context*, but each with a different definition of link-context. This lack of consensus begs the question: What is a good link-context?

The two experiments in this paper address the question by comparing the results of using different link-contexts for the problem of ranking. Specifically, we concatenate the link-contexts of links pointing to a web page to create a *link-context document* used to rank that web page. By comparing the ranking order resulting from using different link-contexts, we found that smaller contexts are effective at ranking relevant urls highly.

1 Introduction

In their 2001 paper on anchor text, Craswell et al.[6] show that ranking based on anchor text is twice as effective as ranking based on document content. This finding spurred many researchers to leverage anchor text for various information retrieval tasks on web pages including ranking, summarizing, categorization, and clustering [12, 5, 4, 2, 3].

As descriptive words often appear around the hyperlink rather than between anchor tags, many researchers have expanded the words past anchor text to a ‘context’ around the link. These *link-contexts* have been defined as sentences [8], a windows [4], and the enclosing HTML DOM trees [12, 2] around the anchor text. Given these competing definitions of link-context, this paper investigates which definition performs best for the task of ranking.

Specifically, the two experiments in this paper explore how defining link-context as either windows, sentences, or HTML trees effect ranking quality. The link-contexts of links pointing to a single page are concatenated to create a *link-context document* for that page. Each link-context document is then ranked by the BM25 algorithm[16] and the order evaluated by various ranking metrics.

The results of the Syskill/Webert[14] experiment indicate that smaller link-contexts tend have better precision and NDCG[10] at low recall and thresholds while larger link-contexts have better precision and NDCG at higher recall and threshold levels. The results of the followup experiment on the AOL collection [13] indicate small link-contexts have the lower mean click-through url ranks which corroborates the earlier conclusion that smaller link-contexts are effective at low thresholds.

This paper proceeds by describing related works (§2), an overview of the experiments (§3), the experiment results (§4 and §5), and concludes with future work (§6).

2 Related Work

The inspiration for this paper was Craswell et al.[6] where link-context was defined as anchor text and their resulting documents were found to be more effective than page source in ranking documents. Prior to Craswell et al. a couple of papers highlight the effectiveness of anchor text as a feature for ranking linked web documents[3, 7]. Anchor text has been used for a variety of information retrieval tasks since McBrien’s WWW indexing worm[11], including topical web crawling [12, 5], and subject classification [4, 2].

The name “link-context” comes from Pant’s paper on topical web crawling but has appeared earlier in different incarnations[12]. Link-contexts have been defined as sentences containing the anchor tag[8], a fixed window before and after the anchor tag [4], the enclosing HTML DOM tree [2, 12, 5], and the enclosing ad-hoc tag (Paragraph, list item, table entry, title, or heading)[17].

Of the papers that use a link-context some justified their definition of what a link-context is. Chakrabarti et al. [4] justified using a window of 50 bytes for classifying pages by looking at the incidence rate of the word ‘yahoo’ within 100 bytes of hyperlinks to ‘http://www.yahoo.com’.

The context of an anchor is important to topical web crawlers because it determines which hyperlink to follow, hence a number of papers in the area experiment with different link-contexts. Notably, Chakrabarti et al. [5] trained a classifier to crawl websites for topic relevant links. Windows of up to five HTML DOM text elements around a hyperlink were part of the feature set, because larger windows only marginally increased their accuracy. Pant[12] experimented with defining link-context as windows of words or HTML DOM trees containing the hyperlink to decide which hyperlink to follow and found parent trees to be most effective.

The results of the Chakrabarti and Pant papers are not immediately applicable to the task of ranking for two reasons. First, the evaluation metrics for topical web crawling do not directly apply to the task of ranking web pages. Second, in topical web crawling one link-context is used per target page, whereas in ranking algorithms, as presented by Craswell, use many link-contexts to create a single document.

3 Experiment

The experiments reproduce the original Craswell et al.[6] experiment by ranking each document’s relevance to a query by both the page content and the content of the anchor text in hyperlinks pointing to the page. In addition, we define

several different link-contexts and compare the effectiveness of link-context documents in ranking web pages. The two experiments in this paper use the following framework and differ only in the source of the data and the evaluation metrics.

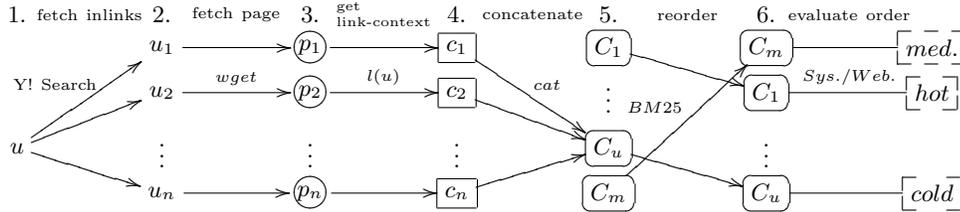


Fig. 1. Creating the link-context document

Briefly, as illustrated by fig. 1, for every url u in a given subject, the experiment found the link-context document defined by link-context l using the following approach:

1. Use the Yahoo! Search API[1] to find the urls of pages linking to u .
2. Fetch pages of in-linking urls from the internet.
3. Parse pages for the link-context l around the hyperlink to u ($l(u)$).
4. Concatenate link-contexts to create the link-context document.

Given all the link-context documents for urls in a subject the experiment then:

5. Use the BM25 algorithm to rank the link-context documents.
6. Evaluate the success of the ranking against the Syskill-Webert ranking of the pages or the AOL click-through data (fig. 1 illustrates the former).

3.1 Data

Syskill/Webert Experiment The data for the first experiment comes from the Syskill/Webert (SW) data[14] available through the UCI KDD archive [9] and was created in 1997 for the purpose of predicting user ratings of web pages with respect to a given topic. Each user was given an ‘index’ page with links to websites on a single subject. After visiting each link the user was intercepted with a screen asking the user to rate the visited page by adding it to a ‘hot’ list, a ‘lukewarm’ list (medium) or a ‘cold’ list[14]. While the user rating is not an explicit ranking order for the pages, it provides a partial order. One subject in the SW dataset, Bands, was dropped for lack of in-links to the SW pages.

The link-contexts for the SW pages were found by querying the Yahoo! Search API. While the Yahoo! Search API did not find any pages linking to a couple SW pages, 90% of pages were viable for this experiment. In total 253 SW pages were used in the project and 7,277 pages were used to construct in-link documents.

One potential criticism of the data used for this experiment is that the SW data was collected in 1997 while the in-link pages were collected in 2008. However, as there is no reason to believe that this time gap between data will effect any link-context unequally, the data still permits a relative comparison between link-contexts.

AOL User Data Experiment As the data in the first experiment was collected in 1997, we use a more current data source, the 2006 AOL User Session Collection[13], for the second experiment. The data contains user queries and, if a search result was chosen, the base domain of the clicked hyperlink and the rank of the result.

Of the queries in which a user clicked a url, over 90% had a rank of 11 or less and 97.5% had a rank of 50 or less, fig. 2. For this reason we choose the first 50 pages returned by the Yahoo! search API for a query as the candidate pages to rank using link contexts.

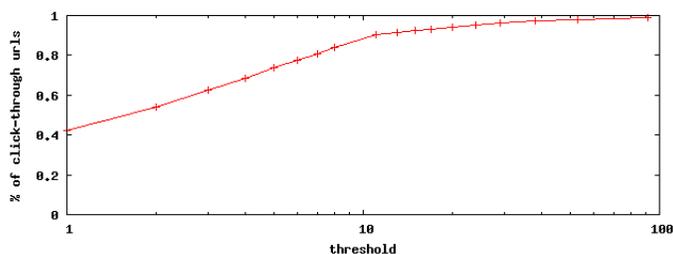


Fig. 2. Percent of click-through urls in the AOL User Data with AOL rank lower than the threshold

Of the 19,442,629 click-through queries in the AOL dataset, we randomly chose queries discarding those where no click-through event occurred or if the click-through url was not found in the first 50 results returned by the Yahoo! Search API. In total 100 click-through queries were collected and used as subjects for this experiment.

The link-context documents for these first 50 results are, as in the first experiment, constructed using the Yahoo! Search API. In total there are 100 queries, with 5,000 primary pages and 250,000 secondary pages (used to construct link-context documents).

3.2 Ranking Algorithm

The ranking algorithm used in the original Craswell et al.[6] paper is BM25[16] and to allow for direct comparison with the Craswell et al. paper, both experiments in this paper also use BM25 for ranking. To this end, the Xapian package[18] was used to index and query the documents generated by different context definitions.

3.3 Link-Context Definitions

Both the main and auxiliary experiments employ three kinds of link-contexts found in the literature: windows of words [4], HTML trees[2, 12], and sentence[8] link-contexts.

Window Contexts A window link-context grabs the n words before and after the hyperlink pointing to the target web page. Words are defined as whitespace-separated strings of letters in HTML DOM text elements. The value of n in this experiment ranges from one to five.

Tree Contexts The tree-based link-contexts are defined by the HTML DOM tree structure and all text elements found in the tree are included in the context. In this experiment the trees rooted at the parent, grandparent, and great-grandparent elements of the target anchor tag are candidate link-contexts. Not all HTML pages come well-formed but by using HTML Tidy[15], we extracted a reasonable approximation of the HTML tree intended.

Sentence Context We also consider a syntactic link-context, defined as the sentence in which the hyperlink appears. A sentence is defined as the words between punctuation marks (./?!). Though not all anchor tags appear within well-formed sentences, we accept paragraph tags, cell tags, and list tags as sentence boundaries if they appear closer than punctuation.

3.4 Performance Metrics

Syskill/Webert Experiment The ranking metrics used to measure success of different link-contexts in ranking web pages are: precision, recall, and NDCG[10]. For the purpose of precision and recall only ‘hot’ documents were considered relevant. For NDCG the rating of hot/medium/cold documents are treated as the rating 2/1/0, respectively. The performance of each link-context is graphed in a 11-point precision-recall graph and the NDCG is graphed for thresholds up to ten.

AOL Experiment The experiment using the AOL data has the same BM25 ranking algorithm and the same candidate link-contexts but uses the rank of the click-through result as the performance metric of the results. Specifically the mean rank of the click-through result over all queries is the metric for evaluating this experiment.

4 Results of Syskill/Webert Experiment

Initially we compared window, sentence and tree contexts separately and then compared the most viable contexts from initial experiments to draw overall conclusions. For brevity, the results of the intermediate results are published in

an extended paper[] and here we present the overall comparison and examine whether we reproduce the original Craswell et al. experiment.

This experiment was not a successful reproduction of the Craswell et al. experiment[6] in that the anchor text doesn't generally outperform page source. Except at recall levels of 10%, anchor text's precision worse than the page source, fig. 3(a). In the NDCG graph, fig. 3(b), anchor text performs worse than the page source after a threshold of two.

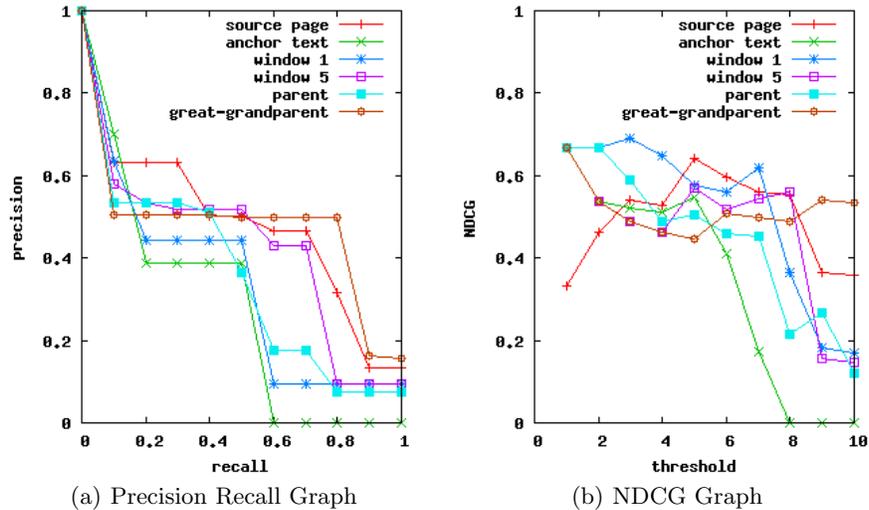


Fig. 3. Overall Comparison

The poor performance of anchor text at low levels may be due to the time difference between when the source pages were collected and when the anchor text was collected (§3.1) which may not only indicate a shift in meaning of the page but also could lead to fewer anchor texts pointing to the page if the page is no longer online.

For an overall comparison the results for the anchor text, page source, window of size 1, window of size 5, parent and great-grandparent link-contexts are included because they performed best in their class of link-context[]. Figure 3(a) and 3(b) highlight the general trend that small contexts, anchor text, windows of size 1, and parent link-contexts perform well at low thresholds and recall while large contexts tend to dominate at higher thresholds and recall.

The precision of the anchor text, page source, window of size 5, and great-grandparent link-contexts beat out the others at increasing levels of recall. The NDCG levels of window 1 link-context equal or exceed other link-contexts until a threshold of 5 when the source page exceeds it, (excepting at threshold 7 where window 1 is again the highest). Finally, at thresholds above nine the great-grandparent context's NDCG is greatest.

5 Results for AOL User Data Experiment

Since small contexts are good at pushing the most relevant pages to the top (high NDCG at low thresholds §4) and click-through pages are quite relevant, we expect small contexts to have the lowest mean click-through rank. Also because the data collection time and the validation data time are current, we expect to see better performance from link-contexts than the page source.

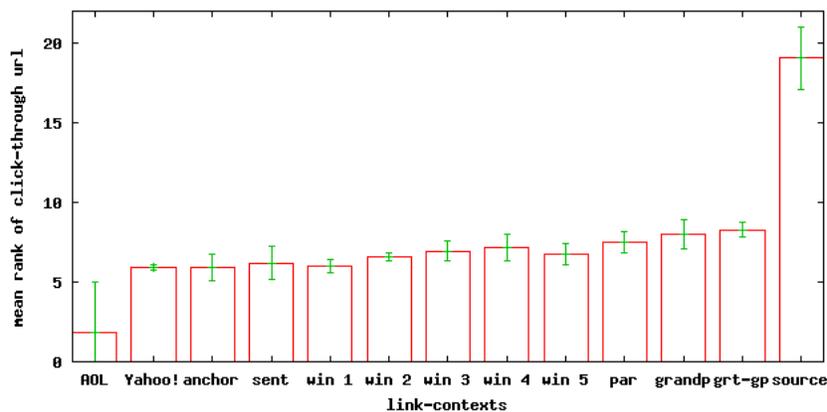


Fig. 4. Mean Rank of Click-through Url with (standard) error bars

The average rank of the click-through domain using the link-context documents and BM25 ranking is displayed in figure 4. Though all link-contexts are beaten by the full-scale search engines, small link-contexts such as windows of size 1 are highly competitive. The sentence and anchor link-contexts have mean ranks comparable to Yahoo!’s but with greater standard error. Contrary to our hypothesis the larger context windows of size 5 did better than those of size 3 and 4. Though all three contexts are within one another’s error range, this weakens our hypothesis and requires further investigation. Though the BM25 ranking combined with link-context documents perform worse than search engines it is remarkable how well it performs without a complicate feature set. Finally, the experiment does reproduce the results in the Craswell experiment as the source page performs worse than all link-context documents.

6 Conclusion and Future Work

This pilot study is a step toward answering the question of which link-context to use. The lesson of the first experiment is that if one cares about the first pages returned then a small context is preferable and if overall quality of results is more important than a larger one is better. The second experiment corroborates the

first by showing that the small link contexts have lower mean click-through urls ranks. Particularly, using link-contexts defined by windows of size 1 was effective in bringing click-through urls to the top of the list. While these link-contexts are not alone better than established search engines, they may be highly effective features in a search algorithm with many features.

Future work includes using a linear regression to explore the merit of using a linear combination link-contexts. Also, because of the effect of size on ranking it may be interesting to train the algorithm at different thresholds. Finally, in future experiments increasing the variety of link-context definitions, for example taking larger windows or the whole page as a link context, may prove fruitful.

References

1. Yahoo! search web services. <http://developer.yahoo.com/search/>.
2. G. Attardi, A. Gulli, and F. Sebastiani. Automatic web page categorization by link and context analysis. In C. Hutchison and G. Lanzarone, editors, *THAI-99*, pages 105–119, Varese, IT, 1999.
3. S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. *Computer Networks and ISDN Systems*, 30(1-7):107–117, 1998.
4. S. Chakrabarti, B. Dom, D. Gibson, J. Kleinberg, P. Raghavan, and S. Rajagopalan. Automatic resource list compilation by analyzing hyperlink structure and associated text. In *WWW7*, 1998.
5. S. Chakrabarti, K. Punera, and M. Subramanyam. Accelerated focused crawling through online relevance feedback. In *WWW2002*. ACM, May 2002.
6. N. Craswell, D. Hawking, and S. Robertson. Effective site finding using link anchor information. In *SIGIR '01*, pages 250–257, New York, NY, USA, 2001. ACM Press.
7. M. Cutler, H. Deng, S. S. Maniccam, and W. Meng. A new study on using html structures to improve retrieval. In *Tools with Artificial Intelligence*, pages 406–409, 1999.
8. J. Delort, B. B. Meunier, and M. Rifqi. Enhanced web document summarization using hyperlinks, 2003.
9. S. Hettich and S. D. Bay. The UCI KDD archive. <http://kdd.ics.uci.edu>, 1999.
10. K. Järvelin and J. Kekäläinen. Cumulated gain-based evaluation of IR techniques. *ACM Trans. Inf. Syst.*, 20(4):422–446, October 2002.
11. O. A. McBryan. GENVL and WWW: Tools for taming the web. In O. Nierstarsz, editor, *WWW1*, CERN, Geneva, 1994.
12. G. Pant. Deriving link-context from html tag tree. In *DMKD '03*, pages 49–55, New York, NY, USA, 2003. ACM.
13. G. Pass, A. Chowdhury, and C. Torgeson. A picture of search. In *InfoScale '06*, New York, NY, USA, 2006. ACM Press.
14. M. J. Pazzani, J. Muramatsu, and D. Billsus. Syskill webert: Identifying interesting web sites. In *AAAI/IAAI, Vol. 1*, pages 54–61, 1996.
15. D. Raggett. Clean up your web pages with hp's html tidy. *Comput. Netw. ISDN Syst.*, 30(1-7):730–732, 1998.
16. S. E. Robertson, S. Walker, M. Hancock-Beaulieu, A. Gull, and M. Lau. Okapi at trec. In *Text Retrieval Conference*, pages 21–30, 1992.
17. S. Slattery and M. Craven. Combining statistical and relational methods for learning in hypertext domains. In D. Page, editor, *ILP-98*, number 1446, pages 38–52, Madison, US, 1998. Springer Verlag, Heidelberg, DE.

18. Xapian. Xapian. <http://www.xapian.org/>.