Link Detection in XML Documents: What about repeated links?

Junte Zhang
University of Amsterdam
Amsterdam, the Netherlands
j.zhang@uva.nl

Jaap Kamps
University of Amsterdam
Amsterdam, the Netherlands
kamps@uva.nl

ABSTRACT

Link detection is a special case of focused retrieval where potential links between documents have to be detected automatically. The use case, as studied at INEX's Link the Wiki track, is that of a new, orphaned page (here, a structured XML document) for which we need to detect relevant incoming and outgoing links to other pages (here, the INEX Wikipedia collection). We focus on outgoing links and investigate link density, and especially repeated occurrences of links with the same anchor text and destination.

We provide an extensive analysis of link density and repetition, and look at parameters like the document's length, the distance between anchor text occurrences, and the frequency of the anchor text within an article. We also conduct experiments trying to determine what should be done with links that are repeated. We describe alternative approaches and compare them against two baselines: the first baseline is to link only once, and the second is to link all candidates. The performance is measured with precision and recall in terms of the total set of discovered links. Our main finding is that, although the overall impact of link repetition is modest, performance can increase by taking a informed approach to link repetition.

Categories and Subject Descriptors

H.3 [**Information Storage and Retrieval**]: H.3.1 Content Analysis and Indexing; H.3.3 Information Search and Retrieval; H.3.4 Systems and Software; H.3.7 Digital Libraries

General Terms

Measurement, Experimentation

Keywords

Wikipedia, Link Detection, Repeated Links, Evaluation

SIGIR 2008 Workshop on Focused Retrieval
July 24, 2008, Singapore

1. INTRODUCTION

Information Retrieval methods have been employed to automatically construct hypertext on the Web [1, 2, 6], as well for specifically discovering missing links in Wikipedia [4, 3]. These missing links are added manually by users, as well as automatically with scripts. We focus on automatic link-detection. The purpose of detecting missing links is to make navigation within and between pages easier.

To automatically detect whether two nodes, such as two XML files, are implicitly connected, it is necessary to search for some text segments that both nodes share, either explicitly or semantically. Often it is only one specific and extract string [1]. With whole documents, hyperlinks can be generated on the file level. With semi-structured documents, such as HTML or more structured documents in XML, one can *deeplink* by generating hyperlinks on the element level using the structure of the document.

In Wikipedia [14], excessive links make a Wikipedia article difficult to read. Good links in Wikipedia are relevant to the context. A whole document gets more context by adding more links, as extra information is added. However, there is the problem of *link density* in structured XML documents, such as the INEX Wikipedia collection consisting of 660,000 English Wikipedia articles in XML. On the one hand, we may decide to link only once per article to a given destination page. On the other hand, we may decide to link every time that the opportunity presents itself. The issue of link repetition is directly related to link density.

A rule of thumb used at Wikipedia is to "aim for a consistent link density" and "not to link eight words in one sentence and then none in the rest of the article."[1] To further quote the style guideline of Wikipedia:

> For general interest articles, where the links are of the "see also" or "for more information" type, it may be better to not link in the summary, deferring the link until the term is defined later in the article. Numerous links in the summary of an article may cause users to jump elsewhere rather than read the whole summary. For technical articles, where terms in the summary may be uncommon or unusual, and linking is necessary to facilitate understanding, it is permissible and may even be necessary to have a high link density in the introduction.

[1]Wikipedia:Only make links that are relevant to the context, `http://en.wikipedia.org/wiki/Wikipedia:Only_make_links_that_are_relevant_to_the_context`

Specifically, Wikipedia's manual style of links offers hints about repeated links.[2] The issue of overlinking is addressed in this quote:

> A link for any single term is excessively repeated in the same article, as in the example of overlinking that follows: "Excessive" is more than once for the same term, in a line or a paragraph, because in this case one or more duplicate links will almost certainly appear needlessly on the viewer's screen.

However, the inverse could also be true when one anchor term is not linked enough. Anchor terms that are more important should be linked more often. The same Wikipedia manual reads:

> Good places for link duplication are often the first time the term occurs in each article subsection. Thus, if an important technical term appears many times in a long article, but is only linked once at the very beginning of the article, it may actually be underlinked. Indeed, readers who jump directly to a subsection of interest must still be able to find a link.

It has already been pointed out in [7] that the amount of hypertext matters: if you give someone 'too much' hypertext, they will become lost; if you give them 'too little', they will not even be able to get started. The former is also called *overlinking*, while the latter is called *underlinking*. Both cases are seen as poor link structure. Furthermore, automatic or semi-automatic constructed hypertexts with information retrieval techniques can be difficult to use, causing user disorientation and cognitive overload [1].

These guidelines and manual are used for adding manual links in Wikipedia. This leads to our research questions:

- Does link repetition occur, and how often?
- How can we predict when to link in a XML document?
- Will link detection in XML documents improve by taking into account repetitions of links?

The issue link repetition and link density in automatic link detection, especially in XML documents like the INEX Wikipedia collection, is still a conundrum, which we try to address in this paper. The remainder of this paper is structured as follow: we start by embedding our work with related literature in Section 2, then we discuss our experimental setup in Section 3, the results are evaluated and presented in Section 4, and finally we conclude with our conclusion of our research question in Section 5.

2. RELATED WORK

2.1 World Wide Web

On the World Wide Web, automatic hyperlink tools are already available. In [1] an overview of different approaches is given of information retrieval techniques for the automatic construction of hypertext. The idea of global and local similarity is outlined, where the former is related to the whole document, and the latter to text segments in a document. There is a strict correlation between both, and the local similarity is more orientated towards precision to refine results later. Another distinction that was made was between first-order hypertexts which are added by the document author, and second-order hypertexts which are automatically added.

Entirely automatic methods for building second-order hypertext using Information Retrieval (IR) and inspired by relationship visualization techniques and graph simplification is presented in [2]. It is pointed out that document linking is based upon information retrieval similarity measures with adjustable levels of strictness. Using no significant natural language processing, and standard IR indexing techniques, inter-document links can be located and described. The idea of linking using structure is also addressed, like creating a link between parts of a document that are most similar.

A survey of the actual use of hyperlink analysis in web search engine ranking like Google's PageRank and other applications is given in [6]. Such other applications are crawling for high quality pages, search-by-example, computing the reputation of websites, finding "web communities", websites related to same or related topics, and web page categorization. The importance of link evidence for improving the ranking in ad-hoc retrieval using the INEX Wikipedia collection is shown in [11]. These research directions are related to the research in this paper, although our desired result is to improve the quality of automatic link detection for user navigation and serendipitous information seeking in the Wikipedia.

2.2 Link-detection in Wikipedia

For Wikipedia, automatic link construction tools are also available such as the link suggesting tool developed by [9]. In [4] a 2-step approach is presented to automatically discover missing hyperlinks in Wikipedia using the link structure. They compute a cluster a highly similar articles around a given article, and then they identify candidate links from those similar articles that might be missing on that given article. The clusters are computed by using co-citation, i.e. two articles are similar if they are co-cited by a third one.

Since 2007 is there a specific link-detection task at INEX called Link-the-Wiki (LTW). This task basically consists of two sub-tasks: detecting incoming links (from destination to source) and outgoing links (from source to destination).

An approach based on the content of an article is presented in [3], where the whole article is used as a query against the index using the Vector Space Model (VSM). The influence of setting different thresholds for the pool of related articles, and the effect of title matching was checked by measuring the performance using standard IR measures as Mean Average Precision (MAP).

In [5] the incoming links were detected by running a NEXI query [13] with the name nodes (titles) of the topics, or *//article[about(.,name)]*. For detecting outgoing links, all the titles in the collection were stored in an in-memory hash-table and looked up, where the window size varied from 8 words down to 1 word, and included stop words.

A different LTW approach is presented in [8], where the authors detect links by storing an anchor text a if it has a certain ratio and looking it up again during the link detection process. That ratio γ is the ratio between the number of articles that has a link from anchor a to a file d, and the number of files in which the anchor text a occurs only once.

[2]Wikipedia:Manual of Style (links), `http://en.wikipedia.org/wiki/Wikipedia:Manual_of_Style_(links)`

Using this ratio, highly relevant anchor texts can simply be looked up.

The approach adopted in [10] identified terms within the document that were over represented and from the top few generated queries of different lengths. Potentially relevant documents were identified by retrieving and ranking them in BM25/Okapi, where these terms were used as query.

Since the LTW track in 2007 only evaluated unique article-to-article links, the repetitions of links has not been taken into account by this previous and related research. So there was only focus on *what* to link, but not *when* and *how often*. We investigate mainly the latter two directions in this paper. The main challenge of automatic link detection is the actual detection of anchor terms, i.e. substrings that should be made clickable. Once these anchors have been found, IR technologies take over with finding and ranking the most plausible destinations.

3. EXPERIMENTAL SETUP

We start by analyzing the 90 existing topics (*qrels*) that were used at the INEX LTW track by looking at the types of links, repetition of links, and the article length. We continue by defining parameters that could possibly have impact on the detection of repeated links. Finally, we present our link detection approach, and the baselines that we used for our experiments.

3.1 Topics Analysis

3.1.1 Types of links

There are several types of links in the topics. These links have been implemented in the Wikipedia collection using XLink. An overview of the occurrence of these types of links in the un-orphaned (original) topics is presented in Table 1. The top 8 most frequent anchor terms on both the collection and file level are presented in Table 2 and 3.

	All topics		Link in article	
Type	**Uniq**	**Total**	**1×**	**Max**
`<collectionlink>`	5,786	8,868	5,781	15
`<unknownlink>`	1,308	1,458	1,271	7
`<outsidelink>`	807	851	778	5
`<imagelink>`	197	212	197	15
`<languagelink>`	79	1,147	1,147	1
`<wikipedialink>`	59	60	58	1
`<weblink>`	27	28	26	2
Total	8,263	12,624	9,232	-

Table 1: Statistics of the types of Links in the 90 un-orphaned LTW articles on the file level

For example, if we regard all the links as one distribution, then the `<languagelink>` has 79 different types (appearing once), but the same types are used 1,147 times, of which the single link `<languagelink lang="de">` is used as often as 66 times, which means the same language links are reused in the articles. When we look at each file separately, then a language link appears only once in a file.

In the INEX Wikipedia collection, there three type of links which are used for link detection at the LTW task: `<collectionlink>`, `<wikipedialink>`, and `<unknownlink>`. The `<collectionlink>` comprises of the bulk of the links in the orphaned articles (70.0%). When looking on a global level

Freq.	Anchor term	Target file
51	"2004"	35524.xml
48	"United States"	31882.xml
40	"2005"	35984.xml
21	"France"	10581.xml
21	"United Kingdom"	31717.xml
18	"2003"	36163.xml
17	"2001"	34551.xml
16	"Japan"	15573.xml

Table 2: Frequency of top 8 anchor terms on collection level.

Freq.	Anchor term	In topic	Target file
15	"Florida"	150340.xml	10829.xml
12	"Miami Beach"	150340.xml	109449.xml
12	"2004"	1092923.xml	35524.xml
10	"California"	150340.xml	5407.xml
9	"2005"	1092923.xml	35984.xml
9	"USA"	150340.xml	31882.xml
9	"2004"	2542756.xml	35524.xml
8	"Long Beach"	150340.xml	94240.xml

Table 3: Frequency of top 8 anchor terms on file level.

at all orphaned articles, then there are 5,786 unique types of collection links, out of the total of 8,868. The number of collection links that only occurs once is 4,275, which is 73.9% of the different types of collection links, and 48.2% out of all collection links.

3.1.2 Link repetition

However, not every type of collection link appears once. The collection link to article 35524.xml is occurring most often on the collection level: 51 times, but it surprisingly does not to exist in the 2007 INEX collection that we used. We observe that many links that re-occur are named entities of years and geographical names like that of countries. Table 3 shows that over 3,000 of in total 8,868 links are links that are repeated. This is a substantial amount. On average, there are 98.5 *outgoing* collection links per topic, of which 64.3 per topic are unique, thus occurring once.

Many of the same types of links on a global level are reused in the files, such as links referring to years and dates which are almost always linked. Supporting evidence is given in Table 2. Moreover, 5,781 of the 8,868 collection links appear only once (65.2%) when looking on the file level, see Table 1, an outlier is the collection link 10829.xml (*"Florida"*), which is occurring 15 times in the topic 150340.xml (*"Miss Universe"*). The reason is that the topic *"Miss Universe"* has a very high link density, and the anchor term *"Florida"* occurs in total 15 times in the file, and thus is linked in all instances. A distribution plot is depicted for all link (re-)occurences in Figure 1. Most of the links occur only once, however, a substantial subset re-occurs.

3.1.3 Links in relation to article length

We observed that the link density in Wikipedia articles is mostly consistent and dependent on the length of an article. The length of an article is calculated by discarding all the XML structure, so we only obtain the cumulative length of all the text nodes in a file.

We found that there is a significant strong positive rela-

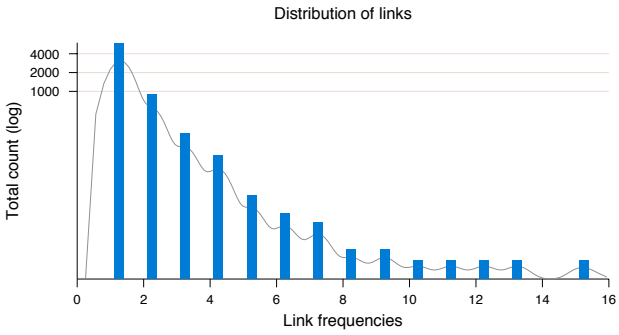


Figure 1: Distribution of all link frequencies

tionship between the length of a Wikipedia article (excluding structure) and the number of links appearing in that article (Pearson correlation coefficient $r = 0.78$ at $p < 0.01$, or Spearman's rho $= 0.85$ at $p < 0.01$), i.e. longer articles have more links than shorter articles, see Figure 2. Moreover, the average length of an anchor text is 12.3 characters, only 62 (0.7%) collection links are 3 characters or shorter.

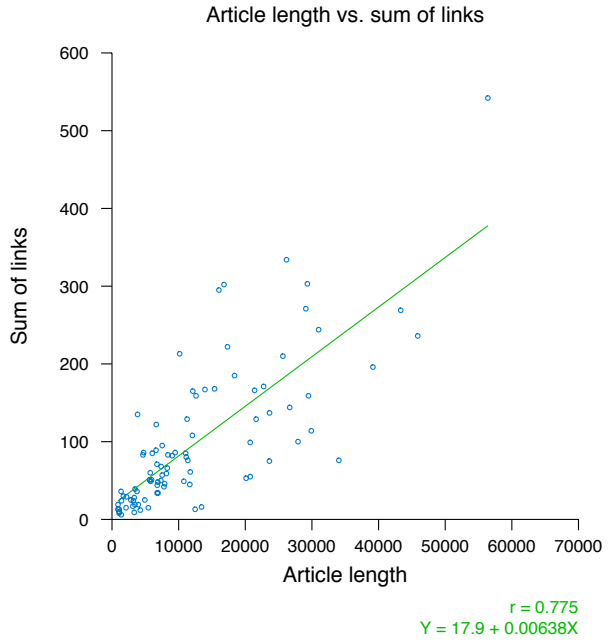


Figure 2: Strong positive correlation between article length and number of links.

The link density can be measured simply using a ratio,

$$\text{Link density ratio} = \frac{\text{total links}}{\text{article length}} \qquad (1)$$

A distribution of the occurrences of the different types of links is presented in Figure 1 and an overview of the length distribution of the articles is given in Figure 3. We see in Figure 4 that the majority of the topics have a similar link density.

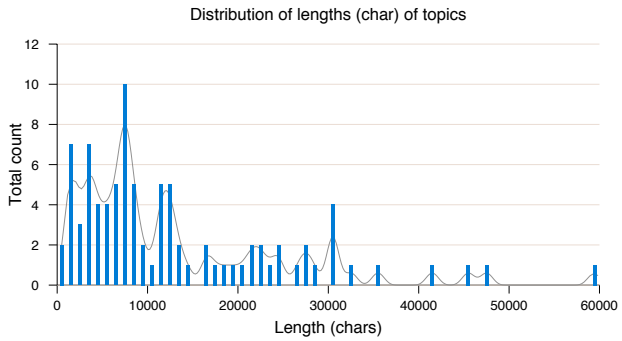


Figure 3: Distribution of article length of 90 topics

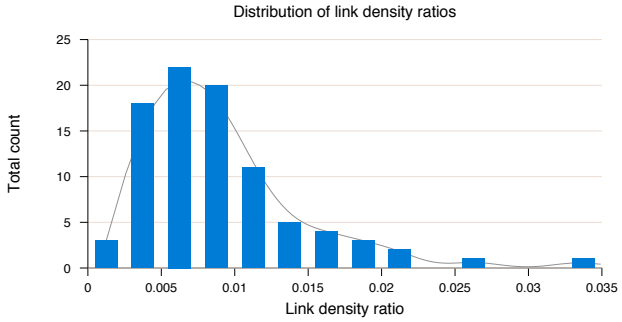


Figure 4: Distribution of link density ratios of 90 topics

3.1.4 Variables

In our experiments we check what the effect is of using 2 *dependent variables*, namely (1) "anchor distance" and (2) the number of "repeated candidate links", on the link detection performance. This performance is measured by comparing them against the ground truth of real links, which makes the number of real links our *independent variable*. Our definitions of the 2 dependent variables:

Anchor distance (AD) The distance between two anchor texts that refer to the same destination node, or A and A', can be calculated. We define it as the function, which we call the "Anchor Distance" AD, which is calculated as

$$AD(A, A') = rindex(A') - rindex(A), \qquad (2)$$

where *rindex* is a function that determines the index of the last occurrence of a letter or the substring in the whole file. A substring of a string $T = t_1 \dots t_n$ is a string $\hat{T}$ that is a subset of T, or $\hat{T} = t_{i+1} \dots t_{m+i}$ where $0 \leq i$ and $m + i \leq n$.

Repeated candidate links (RCL) A link that has been detected with our method is a link candidate, which does not necessarily have to be a real link. A repeated link candidate is a candidate link that occurs more than once in a topic.

The AD is directly related to the concept of link density, e.g. a greater AD means that the link density is less, while

a smaller AD show that the link density is greater. If a link occurs only once in an article, then it means that the distance is 0. Table 1 shows that the majority of a collection link occurs only once in an article, and that in one article the same collection link appears up to 15 times. Table 4 gives descriptive statistics over the 2 dependent variables over all detected candidate links that are repeated in the orphaned topics, and additional information about the article length is given. We did not choose to make the article length a third dependent variable, because it does not relate to individual candidate links, but only topics as a whole. Moreover, article length is implicitly connected with anchor distance and repeated candidate links. One candidate link is linked in the beginning and end of a file and has an extreme anchor distance.

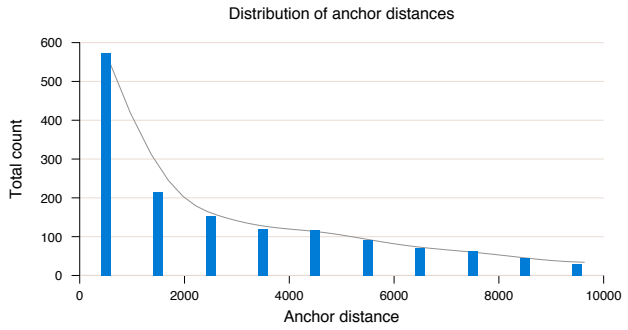


Figure 5: Distribution of anchor distances for an anchor a

	Anchor distance	*Article length*	*Link candidates*
Mean	9,382.97	13,454.58	10.76
Std.	11,418.94	12,191.25	29.71
Min	59	951	2
Med.	5,195	8,634	3
Max.	58,241	58,984	544

Table 4: Statistics of anchor distances (char), article lengths (char) and detected link candidates over the 90 orphaned topics.

3.2 Link Detection Method

3.2.1 Identification of related documents

We use the same method as outlined in [3], where we used the Vector Space Model (VSM) to retrieve related documents (articles) by using the whole article as a query to the index, where the index terms were stemmed using the Porter Stemmer, but no stopwords were removed. Our vector space model is the default similarity measure in Apache Lucene [12], i.e., for a collection D, document d, query q and query term t:

$$sim(q,d) = \sum_{t \in q} \frac{tf_{t,q} \cdot idf_t}{norm_q} \cdot \frac{tf_{t,d} \cdot idf_t}{norm_d} \cdot coord_{q,d} \cdot weight_t, \quad (3)$$

where

$$\begin{aligned} tf_{t,X} &= \sqrt{\text{freq}(t,X)} \\ idf_t &= 1 + \log \frac{|D|}{\text{freq}(t,D)} \\ norm_q &= \sqrt{\sum_{t \in q} tf_{t,q} \cdot {idf_t}^2} \\ norm_d &= \sqrt{|d|} \\ coord_{q,d} &= \frac{|q \cap d|}{|q|}. \end{aligned}$$

We also assume that articles that link to each other are somehow related textually. In [2] it is stated that the stronger the similarity, the better the quality of the relation is between two nodes. We adopt a *breadth m–depth n* technique for automatic text structuring for identifying candidate anchors and text node, i.e. a fixed n number of documents accepted in response to a query and a fixed m number of iterative searches. So the similarity on the document level and text segment level is used as evidence.

It is reported in [3] that when using the Vector Space Model, the best performance is achieved by retrieving the top 300 results. We use the same threshold in these experiments. This link detection experiment focuses only on outgoing links of the type collection `<collectionlink>`, which consists of the bulk of the links in the INEX Wikipedia collection. We do not allow a Wikipedia article to link to itself.

3.2.2 Identification of anchor texts

For our experiment, we only detect outgoing links by using the structure of the documents. An outgoing link is a link from an anchor text in the topic file to the *Best Entry Point* of existing related articles, which in our case was always the text-node of the */article[1]/name[1]* element. There is an outgoing link for topic t, when $S_{1 \cdots n} = T_{q \cdots r}$, where S is the title of a foster article, and T is a line in a orphan article. We assume that the first occurence of an anchor text is also a link. When there are multiple candidate anchors in a file, we learn and apply our link density parameters.

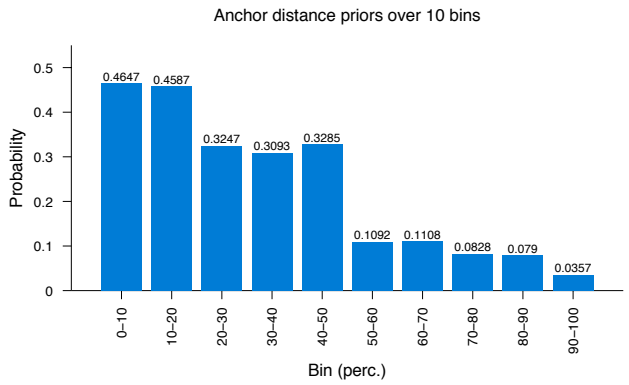
We extract for each topic the title enclosed with the `<name>` tag with a regular expression and store that in a hash-table for substring matching. We apply case-folding, and remove any existing disambiguation information put between brackets behind the title, e.g. *"What's Love Got to Do with It (film)"* becomes the substring *"What's Love Got to Do with It."* We only do exact string matching, and do no take into account linguistic features such as morphological variations between anchor terms or an other kind of normalization.

3.2.3 Priors

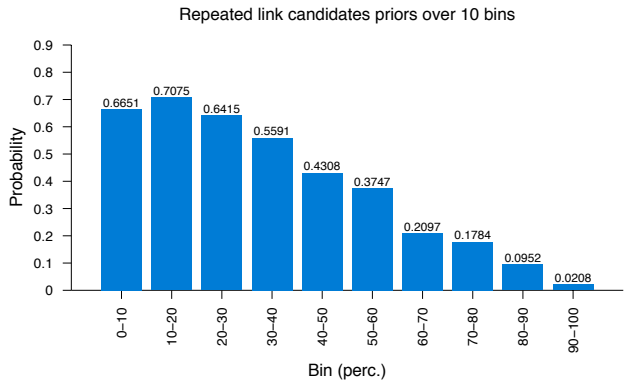
We compute and generate 2 prior plots, which are the anchor distance prior in Figure 6(a) and the repeated link candidates in Figure 6(b).[3] Each of these plots has 10 bins, where bin 1 consists of the bottom 10% of the anchor distances, bin 2 consists of the bottom 10-20 percent of the anchor distances, etc. The same is true for repeated link candidates, e.g. bin 10 in Figure 6(b) contains the top 10% of most frequent repeated link candidates.

We see that the probability that a link is repeated is higher when either the anchor distance is shorter, or the number of repeated candidate links is smaller. This is remarkable, and it may be due to an artifact of the topics, as we imagined that when the anchor distance becomes bigger, then it would become more probable that a link is repeated. We assumed

[3]We use here "prior probability" loosely. Since we are only interested in the shape of the distribution, we do not transform it into a probability distribution (which is in itself a simple mathematical exercise).



(a) Anchor distance priors over 10 bins



(b) Repeated link candidates priors over 10 bins

Figure 6: Priors partitioned in 10 bins.

the same for repeated candidate links. This requires more thorough analysis.

We improve our link detection approach as described in [3] by using the priors to make a Boolean choice: we either link a repeated candidate link, or not. We apply them on 'orphaned' topics, where all XML structure is removed, including any markup of the wikilinks. We do substring matching with the titles of the destination (target) articles to identify the anchor texts. We store these titles in an in-memory hash-table.

During the link detection procedure, we use the priors to make boolean choices on whether to link an anchor term that re-occurs in a topic given one of the 2 *dependent variables* as previously discussed, or P(repeated link | *dependent variable*).

3.3 Baselines and runs

We outlined in the introduction that there are 2 opposite approaches for dealing with link density: we can only link each anchor once, or we can always link a detected anchor. As we have pointed out before; both are not optimal. To compare our runs we use 2 baselines that are based on both polarities.

Baseline 1: "Link once" This baseline simulates the minimal link density. Each detected anchor is only made once a link.

Baseline 2: "Always linking" This baseline simulates the maximal link density. Each detected anchor is made a link.

Moreover, we have the following 4 runs which are in-between both baselines. These runs are based on the prior plots. Since we have 2 dependent variables, we have 2 variants for each run.

Run 1 We match the 2 bins with the highest priors.

Run 2 We match the 6 bins with the highest priors.

4. EVALUATION AND RESULTS

4.1 Evaluation

Our method is evaluated on 'cleaned' topics, where the collection link `<collectionlink>` markup has prior been removed. The original topics with markup are used as *qrels*. The official INEX Link-the-Wiki metrics only measure unique links between Wikipedia articles and do not take into account link density and the detection of repeated links. Using these official metrics, we reported at INEX in [3] a Mean Average Precision (MAP) value of 0.1825 and R-Prec value of 0.2233. It means that when the LTW task would also take into account link repetition and the issue of link density, we would have achieved higher scores.

Our evaluation is restricted to the number of links that are actually present in the un-orphaned pages, or A. Furthermore, our research is focused on investigating the link density in XML files, and not the accuracy of the actual detected links. Therefore to check this effect, we also only evaluate when we detected a link. Table 1 makes clear that most of the links in the topics appear once, so a minority of the links in the articles are actual repeated links.

We use the standard IR metrics for evaluating our methods. We use Precision P and Recall R. Precision is the number of detected true positive links tp divided by the sum of true positives and false positives fp, or all detected links D.

$$P = \frac{tp}{tp + fp} \tag{4}$$

where

$$fp(D, A) = \begin{cases} D - A & \text{if } D > A \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

Recall is the number of true positive links divided by the sum of true positives and false negatives fn.

$$R = \frac{tp}{tp + fn} \tag{6}$$

where

$$fn(D, A) = \begin{cases} A - D & \text{if } D < A \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

This evaluation means that when we underlink candidate anchor terms, we hurt the recall, but when we overlink a candidate term, then the precision drops. Finally, we use the weighted harmonic mean of precision and recall, the balanced F-score F

$$F = \frac{2 \cdot (P \cdot R)}{P + R} \tag{8}$$

4.2 Results

The results are shown in Table 5. When we focus on the performance of the baseline runs, we see that the baseline runs perform relatively well. The main reason is that most of the links in the topics occur once, but obviously this goes at the expensive of the recall. Baseline 2 outperforms baseline 1 because a very high recall is achieved. This causes a slight drop in the precision as compared with the 1st baseline.

When we look at our runs, we see that some of them achieve higher precisions. A higher precision is more valued in our evaluation, because it means that the links are properly placed in terms of frequency and density. Run 1 (RCL) performs best overall, which indicates that the actual number of repeated candidate links is related to detect whether a link should re-occur. We also improve the link detection performance by taking into account the anchor distance in Run 2 (AD).

Run	Precision	Recall	F-Score
Baseline 1	0.8459	0.8043	0.8206
Baseline 2	0.7526	0.9967	0.8053
Run 1 (AD)	0.7635	0.8750	0.7790
Run 2 (AD)	0.8517	0.8126	0.8279
Run 1 (RCL)	0.8445	0.8286	0.8295
Run 2 (RCL)	0.8517	0.8126	0.8279

Table 5: Overal results for link detection.

In Table 6 we only present the top 8 detected links sorted by anchor distance with the 2nd baseline run. This table illustrates an interesting finding, which is the clear relation between link detection and the topicality of documents. All the detected candidate links are related to the topic "communism". We also see that we obviously overlink overwhelmingly. Two out of the 8 detected link candidates are actually false links when using the un-orphaned topics as 'ground truth'. However, when looking at the anchor terms, both of them seem very plausible links. It makes clear that to really determine whether a link is needed, user assessments are required.

AD	#Real	RCL	Topic	Anchor
58,241	2	15	15641.xml	russia
58,057	0	5	15641.xml	communist party
57,912	3	6	15641.xml	joseph stalin
57,143	2	3	15641.xml	leon trotsky
56,862	2	3	15641.xml	stalinism
56,212	2	9	15641.xml	moscow
55,840	0	5	15641.xml	cult of personality
51,018	2	34	15641.xml	soviet union

Table 6: Zooming on 8 results with longest anchor distance (AD) from baseline 2, where article length is 58,984.

4.3 Discussion

A limitation of our experimental setup is that we do not take into account the variable of "Intuitiveness." According to Wikipedia's guidelines, piped links should be kept as intuitive as possible. Piped links should not made "easter egg" links, which require the reader to follow them before understanding what's going on.

Our link detection method does not deal with violations of these guidelines as we do exact substring matching, and such violations are count as true positives in the automatic evaluation, and subsequently these hurt the performance of our method. Example 1 is used by Wikipedia as an instance of such a link.

(1) ... and by mid-century the puns and sexual humor were (with only a few [[**Thomas Bowdler | exceptions**]]) back in to stay.

The reference to *"Thomas Bowdler"* is not seen, unless the reader clicks on it or hovers over the link. If there are cases of such references, then the article should be explicitly linked by using a "see also" as in Example 2, or rephrased as in Example 3. In any case, the exact anchor terms should be made clear explicitly. Using Wikipedia's guidelines, noise in the data, such as variants of the same terms should not occur often, but dealing with such noise will certainly improve the accuracy of link detection.

(2) ... and by mid-century the puns and sexual humor were (with only a few exceptions; see [[**Thomas Bowdler**]]) back in to stay.

(3) ... and by mid-century the puns and sexual humor were back in to stay, [[**Thomas Bowdler**]] being an exception.

Another limitation of our study is that we did not properly deal with overlapping anchors, which should be avoided or parsed correctly. In the topic *"Educational progressivism"* (10005.xml) we identify 2 links in the same substring *"education reform"*, namely (1) *"education"* and (2) *"education reform"*. Example 4 shows these link candidate instances in simplified XML form. A solution may be to always select the longest substring.

(4) `<link> <link>` education `</link>` reform `</link>`

Finally, related to Example 4 is the issue of proper segmentation of the anchor terms. If we only match substrings that are separated with non-word boundaries, then we will not find anchor terms like *"Yahoo!"*. That is why we do plain substring matching, where the trade-off is generating too many candidate anchor terms (and thus links).

5. CONCLUSIONS AND FUTURE WORK

In this paper we described our work on predicting link density in XML documents for automatic link-detection. We raised 3 questions, and we address them here.

- Does link repetition occur, and how often?

In our analysis we showed that the same links do re-occur in the same documents. A substantial subset of the number of links are actual repeated links.

- How can we predict when to link in a XML document?

The main challenge in link detection is the detection of anchor terms. To find relevant anchor terms, we first cluster related documents using the Vector Space Model. Using the structure of XML documents, we extracted relevant substrings in the `<name>` nodes of Wikipedia articles. We assumed that a link should be created at the first instance.

To predict when a repeated link candidate should be actually made a link, we conducted a study with 2 variables. There variables are the distance between 2 of the same anchor terms, and the total number of possible link candidates in a file. We showed that both variables matter in dealing with repeated links.

- Will link detection in XML documents improve by taking into account repetitions of links?

We gave an outline of our approaches for automatic link-detection by taking into account some structure in the XML documents and context with link density analysis. We compared our runs with 2 baselines. The results are evaluated with precision, recall and their weighted harmonic means. Links are repeated in XML documents like Wikipedia articles, and there is also a user need for repeated links. Our preliminary experiments showed that when we take into account repeated links in the 'ground truth', we can achieve better link detection performance compared to the baselines of 'linking once' and 'link always'.

We presented our preliminary work on this subject. For future work, we would like to conduct more studies with different samples of documents, and test it more thoroughly. We also would like to apply and test our approaches with users on real-world problems with other XML datasets, such as linking archival finding aids, where we presented our conceptual framework in [15]. Moreover, we will further improve our method by making more use of the context of the anchors of the hyperlinks. Detecting variants of the same candidate anchor terms will also be investigated.

6. ACKNOWLEDGMENTS

Junte Zhang was supported by the Netherlands Organization for Scientific Research NWO under grant # 639.072.-601. Jaap Kamps was supported by the NWO grants # 612.-066.513, 639.072.601, and 640.001.501, and by the E.U.'s 6th FP for RTD project MultiMATCH contract IST-033104.

7. REFERENCES

[1] M. Agosti, F. Crestani, and M. Melucci. On the use of information retrieval techniques for the automatic construction of hypertext. *Information Processing and Management*, 33:133–144, 1997.

[2] J. Allan. Building hypertext using information retrieval. *Information Processing and Management*, 33:145–159, 1997.

[3] K. N. Fachry, J. Kamps, M. Koolen, and J. Zhang. Using and detecting links in wikipedia. In N. Fuhr, M. Lalmas, A. Trotman, and J. Kamps, editors, *Focused access to XML documents: 6th International Workshop of the Initiative for the Evaluation of XML Retrieval (INEX 2007)*, Lecture Notes in Computer Science. Springer, 2008.

[4] S. Fissaha Adafre and M. de Rijke. Discovering missing links in wikipedia. In *LinkKDD '05: Proceedings of the 3rd international workshop on Link discovery*, pages 90–97. ACM Press, New York NY, USA, 2005.

[5] S. Geva. GPX@INEX2007: Ad-hoc Queries and Automated Link Discovery in the Wikipedia. In N. Fuhr, M. Lalmas, and A. Trotman, editors, *Pre-Proceedings of INEX 2007*, pages 403–409, 2007.

[6] M. Henzinger. Hyperlink analysis on the world wide web. In *HYPERTEXT '05: Proceedings of the sixteenth ACM conference on Hypertext and hypermedia*, pages 1–3, New York, NY, USA, 2005. ACM.

[7] K. Instone. Too much hypertext or too little? *SIGWEB Newsl.*, 3(3):22, 1994.

[8] K. Y. Itakura and C. L. A. Clarke. University of Waterloo at INEX2007: Ad Hoc and Link-the-Wiki Tracks. In N. Fuhr, M. Lalmas, and A. Trotman, editors, *Pre-Proceedings of INEX 2007*, pages 380–387, 2007.

[9] N. Jenkins. Can we link it, 2008. `http://en.wikipedia.org/wiki/User:Nickj/Can_We_Link_It`.

[10] D. Jenkinson and A. Trotman. Wikipedia Ad hoc Passage Retrieval and Wikipedia Document Linking. In N. Fuhr, M. Lalmas, and A. Trotman, editors, *Pre-Proceedings of INEX 2007*, pages 365–379, 2007.

[11] J. Kamps and M. Koolen. The importance of link evidence in Wikipedia. In C. Macdonald, I. Ounis, V. Plachouras, I. Rutven, and R. W. White, editors, *Advances in Information Retrieval: 30th European Conference on IR Research (ECIR 2008)*, volume 4956 of *Lecture Notes in Computer Science*, pages 270–282. Springer Verlag, Heidelberg, 2008.

[12] Lucene. The Lucene search engine, 2007. `http://lucene.apache.org/`.

[13] A. Trotman and B. Sigurbjörnsson. Narrowed Extended XPath I (NEXI). In N. Fuhr, M. Lalmas, S. Malik, and Z. Szlávik, editors, *INEX*, volume 3493 of *Lecture Notes in Computer Science*, pages 16–40. Springer, 2004.

[14] Wikipedia. The free encyclopedia, 2006. `http://en.wikipedia.org/`.

[15] J. Zhang, K. N. Fachry, and J. Kamps. Automatic link-detection in encoded archival descriptions. In L. L. Opas-Hänninen, M. Jokelainen, I. Juuso, and T. Seppänen, editors, *Digital Humanities 2008, Conference Abstracts*, pages 226–228, Oulu, Finland, 2008. The University of Oulu.