Using Neighbors to Date Web Documents

Sérgio Nunes[†]
sergio.nunes@fe.up.pt

Cristina Ribeiro[†,‡]
mcr@fe.up.pt

Gabriel David[†,‡]
gtd@fe.up.pt

[†]Faculdade de Engenharia da Universidade do Porto
[‡]INESC-Porto
Rua Dr. Roberto Frias, s/n
4200-465 Porto, Portugal

ABSTRACT

Time has been successfully used as a feature in web information retrieval tasks. In this context, estimating a document's inception date or last update date is a necessary task. Classic approaches have used HTTP header fields to estimate a document's last update time. The main problem with this approach is that it is applicable to a small part of web documents. In this work, we evaluate an alternative strategy based on a document's neighborhood. Using a random sample containing 10,000 URLs from the Yahoo! Directory, we study each document's links and media assets to determine its age. If we only consider isolated documents, we are able to date 52% of them. Including the document's neighborhood, we are able to estimate the date of more than 86% of the same sample. Also, we find that estimates differ significantly according to the type of neighbors used. The most reliable estimates are based on the document's media assets, while the worst estimates are based on incoming links. These results are experimentally evaluated with a real world application using different datasets.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval; H.3.5 [**Information Storage and Retrieval**]: Online Information Services—*Web-based services*; I.7.5 [**Document and Text Processing**]: Document Capture—*Document analysis*

General Terms

Algorithms, Experimentation, Measurement

Keywords

Web Information Retrieval, Web Dynamics, Link Analysis

1. INTRODUCTION

In information contexts, users consider time an important feature. Contrary to the number of incoming links or the term frequency of a web document, the age of a document is an easily perceptible characteristic. The concept of *new* and *old* is well understood by users. Currently, in state-of-the-art systems, time plays a small role in serving user's information needs. Web documents tend to be treated equally despite their age. However, as the web grows older, we speculate that time will play a significantly higher role in the overall web information context.

One way to explore this information would be to include temporal evidence in the ranking of web resources. A first step towards this *temporal-sensitive ranking* is the task of dating web resources. Classic approaches have used document-based features to achieve this goal. HTTP is a stateless request/response protocol. Clients submit standard requests to servers identifying specific resources. Servers reply to the request sending standard headers and, if available, the requested resource (body of the message). HTTP headers include a Last-Modified field representing the date and time at which the resource was last modified. This field has been used with success as an indicator of document's date in real world applications (see Section 2).

However, this approach has a significant drawback. A very high percentage of the response headers do not contain reliable information. Most servers do not return any information at all, while other always return the current date or simply a wrong value [10]. Previous studies have reported percentages of valid Last-Modified values ranging from 40% to 80% of the tested URLs. We elaborate on these statistics in Section 3.

In this work, we explore ways to improve these values by looking at web-based features, namely neighbors. The main rationale is that, by looking at web resources connected to the original document, more information can be gathered, improving the chances of retrieving valid HTTP values. In a nutshell, our hypotheses is that the dating of a single document can be improved if we look at its vicinity. This assertion has an implicit assumption - connected web resources tend to have similar update patterns. Figure 1 depicts the overall context.

Observing the web from the point of view of a single document, we consider three types of web resources: documents containing links to the selected document (incoming links), documents pointed to by the selected document (outgoing links) and, finally, the media assets associated with the document (e.g. images). For each of these sets, the HTTP headers are available. In the end, we expect to improve the percentage of valid Last-Modified fields retrieved. For instance, we expect to find more reliable information in the

WIDM'07, November 9, 2007, Lisboa, Portugal.

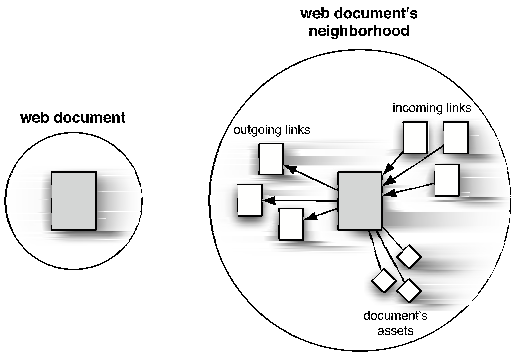

Figure 1: Using Neighbors to Extract Data.

header of static resources such as images than in the documents including them, mostly dynamic resources.

In the following section, we review work related to our study. Section 3 describes the process of building the initial dataset and presents high-level statistics. In Section 4, we describe and explore the idea of using a document's neighborhood to extract data. In Section 5, we setup and conduct several experimental evaluations, validating our findings using different datasets. Finally, Section 6 presents the conclusions.

2. BACKGROUND AND RELATED WORK

In recent years, we've seen a growth in research focused on the temporal/dynamic aspects of the web. First studies have characterized the evolution of web documents [2, 12, 16], mostly to improve search engine crawling [9]. More recent work explores temporal features to tackle tasks closer to the final user, specifically document ranking [6] and multi-document summarization [14]. Below is a brief survey of works that have either explored web document's dates or web document's neighbors.

Web document's dates have been successfully used by Amitay et al. [1] to produce temporal profiles of popular topics. By observing HTTP headers, particularly the Last-Modified field, significant events and trends are revealed. The Last-Modified field is used to approximate the age of the page's content. This work clearly shows that real life events can be exposed mainly due to what authors call *fossilized content*. A new tool, named *Timestamped Link Profile* (TLP), is used to characterize the distribution of timestamped links within a topical community. These profiles are built in three phases. First, a topical collection is assembled by issuing one or several queries to public search engines. These queries are manually selected to characterize the desired topic. Then, a second collection is built combining the links that point to the URLs in the first collection. Finally, the Last-Modified values of the pages in the second collection are gathered and a histogram is plotted. Despite the lack of the Last-Modified field in more than 50% of the URLs, the authors report several experiments where clear patterns are visible.

For web crawling, dating web resources is an extremely important task. Due to its increasing size, it is impossible to frequently crawl the entire web. Search engine designers have implemented strategies to cope with this problem. For example, frequently updated documents tend to be more frequently crawled [9]. Bar-Yossef et al [3] have explored a document's neighborhood to estimate its decay rate. Instead of limiting the observation to the number of dead links of a single document, the document's neighborhood is studied to develop a better estimative of the decay rate. A document's decay rate is not only related the number of dead links in it, but also to the number of dead links in documents at distance $>= 1$ from it.

Using a similar approach, neighborhoods have also been successfully used to estimate properties of individual documents. Sugiyama et al. [17] have developed a technique to refine the TF-IDF scheme for a target document by using the contents of its hyperlinked neighboring pages. Chen et al. [8] propose a method to estimate the PageRank of a particular web page by only looking at the local neighborhood of a node. The authors conduct several experiments and conclude that it is possible to produce a reasonable PageRank value by only looking at a moderate number of nodes. This method can be used to efficiently estimate individual PageRank values without having to perform a large-scale computation on the entire web graph.

In a distinct line of research, Wong et al [18] describe methods of temporal information extraction for document dating. Using natural language processing techniques, specifically information extraction methods, it is possible to identify words or expressions that convey temporal meaning (e.g. "today", "a long time ago"). Although we are not aware of any experimental work following this approach, this technique can also be used to date web documents.

3. DATASET CHARACTERIZATION

Our initial data contains 10,000 URLs from the Yahoo! Directory [1]. This sample was obtained using Yahoo!'s Random Link (YRL) service [2]. YRL service returns a random link from Yahoo!'s index via HTTP GET requests. Table 1 shows the distribution of top-level domains (TLD) in the sample, while Table 2 lists the top 5 domains obtained. In Table 3, depth is defined as the hierarchy of the URL within the domain. For example, the depth of a homepage is 0, while for a URL pointing to a document within the root folder depth equals 1. Although there is no public official information about the YRL service, we think that the sample is based on Yahoo! Directory and not on it's search engine index. The high percentage of homepages concurs with this hypotheses.

This sample was later found to have 66 duplicates (0.66%). All of these duplicate URLs are from the Yahoo! realm, thus the data presented in Table 2 is biased towards the `yahoo.com` domain. This duplication has no impact in our analysis because it occurs in a very small percentage of URLs. It is important to note that this dataset cannot be seen as a random sample from the Yahoo! search engine. Obtaining random samples of URLs from the web is an active research problem [4].

The Last-Modified header field was available in 52% of the requests in our sample. Since many servers return the current date for every HTTP request, we excluded these values. The hourly distribution of Last-Modified values is presented in Figure 2. It is interesting to note that most

[1] `http://dir.yahoo.com`
[2] `http://random.yahoo.com/bin/ryl`

Table 1: TLD Distribution in Sample.

TLD	Count	Percentage
.com	6,012	60.12%
.org	920	9.20%
.uk	643	6.43%
.au	435	4.35%
.edu	430	4.30%
.net	389	3.89%
.us	230	2.30%
.ca	177	1.77%
.nz	110	1.10%
.sg	46	0.46%

Table 2: Top Domains in Sample.

Domain	Count	Percentage
yahoo.com	109	1.09%
geocities.com	103	1.03%
aol.com	37	0.37%
angelfire.com	30	0.30%
tripod.com	27	0.27%

updates occur within USA working hours (8 AM to 8 PM (EST)).

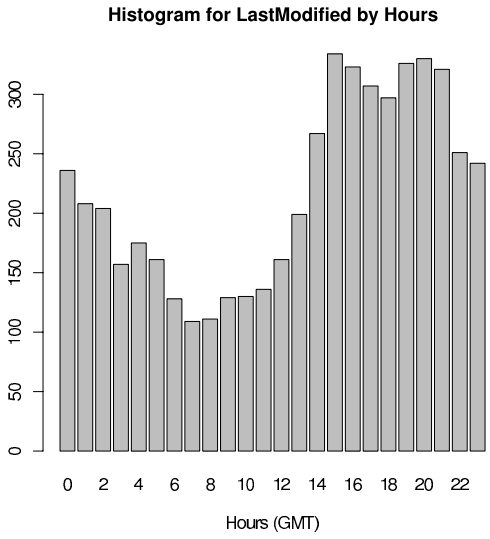

Figure 2: LastModified value by Hour.

The percentage of hosts returning Last-Modified dates has been measured in previous studies. These studies report very different results using very different samples. A summary is presented in Table 4. Gomes et al. [13] attribute their low score to the fact that "the lifetime of contents is getting shorter".

4. EXPLORING NEIGHBORS

As illustrated in Figure 1, we've defined three types of neighbors for a selected web document: incoming, outgoing and assets. A brief description of each one follows.

Table 3: Depth Distribution in Sample.

Depth	Count	Percentage
0 (homepage)	8439	84.39%
1	751	7.51%
2	428	4.28%
3	223	2.23%
4	97	0.97%

Table 4: Percentage of Retrieved Last-Modified in Previous Studies.

Study	Year	% Retrieved
Douglis [11]	1997	79%
Brewington [7]	1999	65%
Bent [5]	2003	56%
Gomes [13]	2006	36%
Our Sample	2007	52%

Incoming Neighbors Incoming neighbors is the set of web documents having a link to the selected document.

Outgoing Neighbors Outgoing neighbors is the set of web documents pointed to by the selected document.

Assets Assets are all the web resources, namely images, objects, CSS files or JavaScript files, that are pointed to by the selected document. More formally, the assets collection includes all URLs that are referenced in HTML `src` attributes.

To obtain these collections of URLs, we used the Perl programming language and the Yahoo! API. Perl was used to download each document and parse its contents. Both the links to other documents (out-links) and the links in HTML's `src` attributes (assets) were collected. The Yahoo! API [3] was used to retrieve the links pointing to each URL. This API does not return more than 1,000 links per request. In our sample, 10% of the tested URLs were above this limit. For these URLs, only the first 1,000 links returned were considered. It is worth noting that results from the Yahoo! API are different from those obtained using Yahoo!'s public web interface [15].

For each of the three sets, an average Last-Modified value was calculated. As defined before, requests returning the current date were considered invalid and not included in the average. In the end, for each URL, we had its Last-Modified value, the average Last-Modified value of the incoming links, the average Last-Modified value of the outgoing links and the average Last-Modified value of its assets. In Table 5 the percentage of valid responses for each type of neighbor presented. As expected, the number of valid answers is higher for media assets (typically static files).

These results confirm that it is possible to use a document's vicinity to improve the percentage of valid answers. However, are these valid results useful? To answer this question, we observed the correlation between a document's Last-Modified value and the Last-Modified average from its neighbors. The strongest correlation occurs with a document's out-links (r=0.74), followed by media assets (r=0.6)

[3] `http://developer.yahoo.com/`

Table 5: Valid Last-Modified by Neighbor Type.

Source	% Valid
In-Links	47%
Out-Links	48%
Assets	83%
Neighbors Combined	94%
Original URLs (baseline)	53%

and finally in-links (r=0.28). Figures 3 and 4 represent the correlation plots for in-links and out-links respectively.

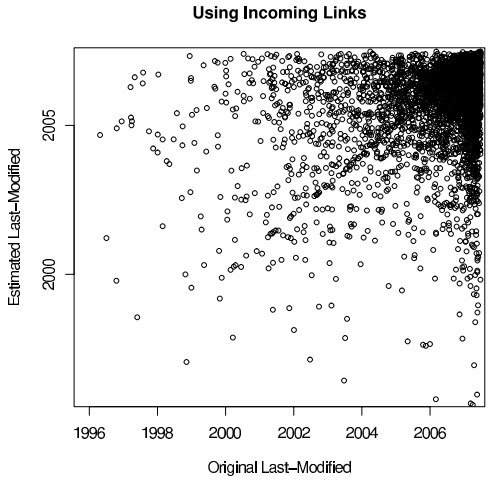

Figure 3: Correlation of Last-Modified values between a document and its incoming links.

To further investigate the nature of these correlations, we separated the out-links in two distinct sets, the outgoing links to other domains (out-out) and the outgoing links to the same domain (out-in). These two subsets exhibit different correlation values (see Table 6). It is possible to observe that the high correlation found with out-links is mostly attributable to outgoing links to the same domain, and likely to the same server.

Table 6: Statistics for outgoing links subsets.

Source	% Valid	Correlation (r)
Out-In Links	45%	0.82
Out-Out Links	38%	0.51

Finally, averaging all neighbors except incoming links, we get a correlation value of 0.73.

5. EXPERIMENTAL EVALUATION

To validate our findings we setup a real world experiment. We built a Timestamped Link Profile (TLP) (see Section 2) for a popular event, specifically the *FIFA World Cup 2006 in Germany*. Our first step was to manually select queries that return relevant results for this topic, namely: "Germany World Cup", "2006 FIFA World Cup" and "World Cup 2006". Then, we selected two sites that were references (i.e.

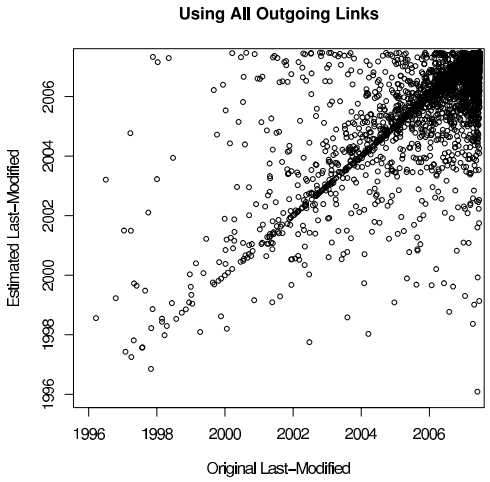

Figure 4: Correlation of Last-Modified values between a document and its outgoing links.

authorities) for this topic - `http://wm2006.deutschland.de` and `http://www.dfb-kulturstiftung.de`.

Using the Yahoo! API, we retrieved all incoming links to these domains (not only to the homepage), resulting in a total of 1,000 URLs after the elimination of duplicates. We then applied our techniques, as described before and using the same scripts, to retrieve Last-Modified values from HTTP headers. We were able to get 258 valid answers from this list. Despite this very low percentage, the TLP plot clearly revealed the expected pattern (see Figure 5). Following the steps described in Amitay et al. [1] we framed our plot to the period of interest, namely from the 1st of January of 2004 to the 1st of January of 2007.

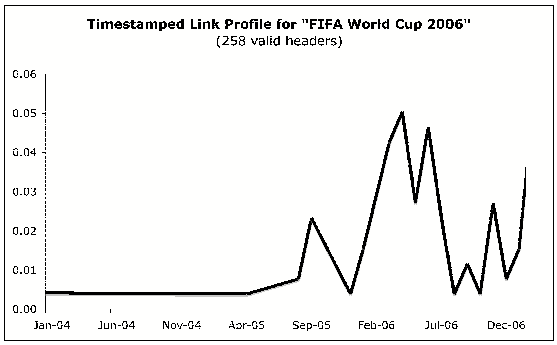

Figure 5: TLP for FIFA World Cup 2006 based on valid Last-Modified values.

To validate our approach, we then produced Last-Modified estimates based on media neighbors and outgoing links neighbors. Incoming links neighbors were excluded due to their low correlation values shown in the previous section. We then plotted another TLP based only on these estimates (see Figure 6). It is important to note that we've only used URLs that returned invalid HTTP headers. In other words,

this TLP is based solely on neighbors from URLs that did not returned a valid HTTP header.

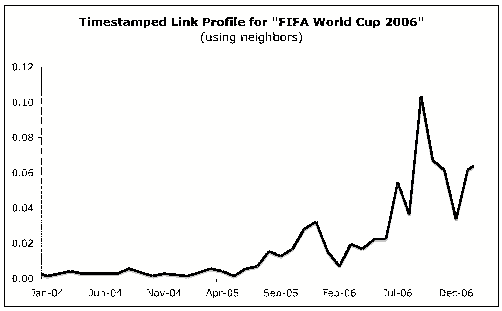

Figure 6: TLP for FIFA World Cup 2006 based on estimated Last-Modified values.

A visual comparison of these figures reveals a similar pattern. The time frame of the depicted event (i.e. "FIFA World Cup 2006") is clearly visible in both plots. There is a small shift to the right in the estimated plot that can be explained by the time needed to update the neighbors. We've conducted the same experiment using other topics and have achieved comparable results. We are confident to state that our neighbors-based approach returns valid results.

In a second experiment, we tried to explore a collection containing a very small number of URLs. In this case, we decided to build a TLP for a single URL. We choose `tsunamihelp.blogspot.com` because it is an regularly updated blog about events that have significant impact and discussion (i.e. tsunamis). First, we collected 100 incoming links using the Yahoo! API. Then, we tried to fetch the Last-Modified field from each URL in this collection. The small number of valid answers (19%) is plotted in Figure 7 for the period between January 2002 and December 2006. Clearly, no pattern emerged from this small set of results.

Using our proposed technique, we produced an estimate for all the 100 URL in the base collection. We were able to get a very high response rate of 99%. The resulting plot for the same period (Figure 8) is significantly different from the previous one. A peak erupts from the plot in September 2006, corresponding to a period after a significant earthquake followed by a tsunami in Indonesia [4]. Based only on a very small collection of web documents we were able to uncover information.

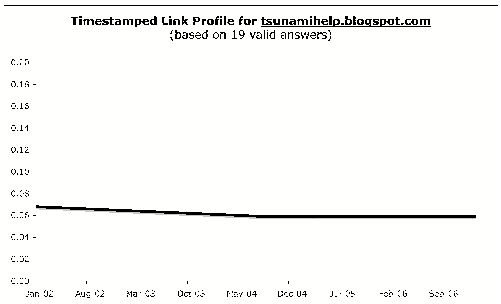

Figure 7: TLP for `tsunamihelp.blogspot.com` based on valid Last-Modified values.

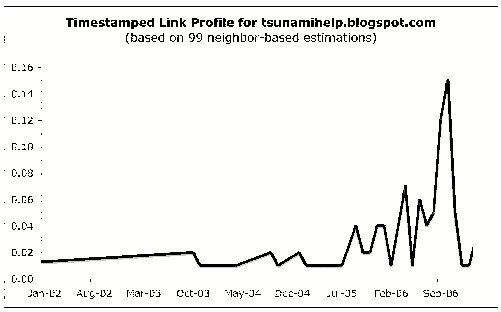

Figure 8: TLP for `tsunamihelp.blogspot.com` based on estimated Last-Modified values.

6. CONCLUSIONS

Dating information resources is a valuable task. There are multiple use cases where this is important, such as finding up-to-date information, organizing web resources, understanding the flow of conversations and crawler scheduling. For web information resources, estimates of the last update date have typically relied on the observation of HTTP headers, on period content analysis looking for content change, or on natural language processing. In this work we explore and evaluate a complementary technique to date web documents, specifically HTML documents.

[4] `http://wikipedia.org/wiki/July_2006_Java_earthquake`

The proposed technique complements the traditional approach based on HTTP headers by also looking at a document's vicinity. The HTML structure of a document is parsed to derive its vicinity. Then, the set of neighbor resources is analyzed and an average update date is calculated. This average is then used to date web document for which HTTP headers are not available. In the end, we are able to improve the percentage of web documents dated using HTTP headers. While alternative approaches try to detect changes by periodically inspecting documents, this technique only requires a snapshot of the web, no historical information is needed.

In our sample of 10,000 URLs, we were able to retrieve valid HTTP headers in 53% of the requests. As expected, observing neighbors, we were able to retrieve valid headers from 86% of the URLs. There were differences between the various types of neighbors; while for media assets we had 83% valid answers, for incoming links we only had 47% answers. Observing static web resources (e.g. images) it is possible to achieve a higher percentage of valid answers. We found that there is a strong correlation between a document's Last-Modified field and the average Last-Modified value from its neighbors (r=0.73).

To validate these findings, we've conducted several experiments using real-word data. We've built Timestamped Link Profiles for several topics, following the original idea by Amitay et al. [1]. For each topic we produced two TLPs, one based on real data and the other based on estimates from media and outgoing links neighbors. The results confirmed our expectations, neighbor-based estimates are valid and accurate. Despite these positive results, it is important to note that estimates should be interpreted with critical judgement. During the experimental validation phase we observed occasional "noise" in the data caused by a non-homogeneous vicinity. For instance, while producing the TLP for one of the topics, we noticed that a large number of neighbors were from a specific web server with badly configured headers. This situation caused a significant shift in the final estimated TLP. Removing these pages from the neighbor's set was sufficient to significantly improve the final estimates.

Overall, our results show that it is possible to use a document's vicinity to estimate its most recent update date. There are multiple possibilities to complement or further advance this study. First, to validate these results in broader contexts, similar studies should be conducted in collections with different characteristics. These collections might include more dynamic documents (e.g. blogs), or include a higher percentage of internal documents versus home pages (e.g. random crawl). Another research path is to study the quality of estimates provided by the different neighbors. There are sites known to be better at providing accurate information. A final suggestion for future work is the exploration of other features in neighboring documents. As shown in this paper, the Last-Modified value from neighboring documents exhibits good correlation values. Which other low-level features evidence a correlation between documents and their neighbors?

7. ACKNOWLEDGMENTS

Sérgio Nunes was financially supported by the Fundação para a Ciência e a Tecnologia (FCT) and Fundo Social Europeu (FSE - III Quadro Comunitário de Apoio), under grant SFRH/BD/31043/2006.

8. REFERENCES

[1] E. Amitay, D. Carmel, M. Herscovici, R. Lempel, and A. Soffer. Trend detection through temporal link analysis. *J. Am. Soc. Inf. Sci. Technol.*, 55(14):1270–1281, December 2004.

[2] R. A. Baeza-Yates, F. Saint-Jean, and C. Castillo. Web structure, dynamics and page quality. In *SPIRE 2002: Proceedings of the 9th International Symposium on String Processing and Information Retrieval*, pages 117–130, London, UK, 2002. Springer-Verlag.

[3] Z. Bar-Yossef, A. Z. Broder, R. Kumar, and A. Tomkins. Sic transit gloria telae: towards an understanding of the web's decay. In *WWW '04: Proceedings of the 13th international conference on World Wide Web*, pages 328–337, New York, NY, USA, 2004. ACM Press.

[4] Z. Bar-Yossef and M. Gurevich. Random sampling from a search engine's index. In *WWW '06: Proceedings of the 15th international conference on World Wide Web*, pages 367–376, New York, NY, USA, 2006. ACM Press.

[5] L. Bent, M. Rabinovich, G. M. Voelker, and Z. Xiao. Characterization of a large web site population with implications for content delivery. In *WWW '04: Proceedings of the 13th international conference on World Wide Web*, pages 522–533, New York, NY, USA, 2004. ACM Press.

[6] K. Berberich, M. Vazirgiannis, and G. Weikum. T-rank: Time-aware authority ranking. In *Algorithms and Models for the Web-graph : Third International Workshop, WAW 2004*, volume 3243 of *Lecture Notes in Computer Science*, pages 131–142, Berlin, Germany, January 2004. Springer.

[7] B. E. Brewington and G. Cybenko. How dynamic is the web? *Computer Networks*, 33(1-6):257–276, June 2000.

[8] Y.-Y. Chen, Q. Gan, and T. Suel. Local methods for estimating pagerank values. In *CIKM '04: Proceedings of the thirteenth ACM conference on Information and knowledge management*, pages 381–389, New York, NY, USA, 2004. ACM Press.

[9] J. Cho and H. Garcia-Molina. The evolution of the web and implications for an incremental crawler. In *VLDB '00: Proceedings of the 26th International Conference on Very Large Data Bases*, pages 200–209, San Francisco, CA, USA, 2000. Morgan Kaufmann Publishers Inc.

[10] L. Clausen. Concerning etags and datestamps. In J. Masanès and A. Rauber, editors, *4th International Web Archiving Workshop (IWAW'04)*, 2004.

[11] F. Douglis, A. Feldmann, and B. Krishnamurthy. Rate of change and other metrics: a live study of the world wide web. In *USENIX Symposium on Internet Technologies and Systems*, December 1997.

[12] D. Fetterly, M. Manasse, M. Najork, and J. L. Wiener. A large-scale study of the evolution of web pages. *Softw. Pract. Exper.*, 34(2):213–237, February 2004.

[13] D. Gomes and M. J. Silva. Modelling information persistence on the web. In *ICWE '06: Proceedings of*

the 6th international conference on Web engineering, pages 193–200, New York, NY, USA, 2006. ACM Press.

[14] A. Jatowt and M. Ishizuka. Temporal multi-page summarization. *Web Intelli. and Agent Sys.*, 4(2):163–180, April 2006.

[15] F. Mccown and M. L. Nelson. Agreeing to disagree: search engines and their public interfaces. In *JCDL '07: Proceedings of the 2007 conference on Digital libraries*, pages 309–318, New York, NY, USA, 2007. ACM Press.

[16] A. Ntoulas, J. Cho, and C. Olston. What's new on the web?: the evolution of the web from a search engine perspective. In *WWW '04: Proceedings of the 13th international conference on World Wide Web*, pages 1–12, New York, NY, USA, 2004. ACM Press.

[17] K. Sugiyama, K. Hatano, M. Yoshikawa, and S. Uemura. Refinement of tf-idf schemes for web pages using their hyperlinked neighboring pages. In *HYPERTEXT '03: Proceedings of the fourteenth ACM conference on Hypertext and hypermedia*, pages 198–207, New York, NY, USA, 2003. ACM Press.

[18] K.-F. Wong, Y. Xia, W. Li, and C. Yuan. An overview of temporal information extraction. *International Journal of Computer Processing of Oriental Languages*, 18(2):137–152, 2005.