# Semantic web access prediction using WordNet

Lenka Hapalova (supervised by Ivan Jelinek)

hapall11@fel.cvut.cz (jelinek@fel.cvut.cz),
Czech Technical University in Prague, Faculty of Electrical Engineering - Dpt. of Computer Science and Engineering, Karlovo nám. 13, 121 35 Prague 2, CZ

**Abstract.** The user observed latency of retrieving Web documents is one of limiting factors while using the Internet as an information data source. Prefetching became important technique to reduce the average Web access latency. Existing prefetching methods are based predominantly on URL graphs. They use the graphical nature of HTTP links to determine the possible paths through a hypertext system. Although the URL graph-based approaches are effective in the prefetching of frequently accessed documents, few of them can pre-fetch those URLs that are rarely visited. In our paper we aim to propose a new prefetching algorithm that would increase the efficiency of Web prefetching and that will embody the new demands for Web personalisation and Web search assistance. The aim of the research is to design a system for web page prefetching. The system should use user's link path history in combination with the semantic path history. To enable this, semantically annotated web pages are necessary. We cannot rely on the web documents' creators thus one part of the work must be the design and implementation of simple annotator based on WordNet just for purposes of our research.

**Keywords:** Web access latency, prefetching, semantic Web, Web access prediction, personalisation, Markov models

## 1 Introduction

Due to the rapid development of the Internet usage and the exponential growth of online information, the Internet has become one of the most important information sources. The usage of World Wide Web (WWW) as a data source has increased as it provides quick and easy access to a tremendous variety of information in remote locations. The wide range of sources' locations is the benefit as well as the drawback of the WWW. Users often suffer from long delay time when they access Web pages – so-called Web access latency. With the rapid growth of Web services on the Internet, users are experiencing access delays more and more often.

Document pre-fetching is an effective tool to improve the access to the World Wide Web. In comparison with caching, pre-fetching aims to pre-retrieve Web documents (more generally Web objects) to the client side even before they are

actually requested. The efficiency of this is mainly limited by the accuracy of Web page access prediction. The accuracy affects the performance of prefetching in two ways: Firstly, evidently bad guess does not reduce the latency. Secondly, bad guess means extra bandwidth burden that subsequently means even longer delays in Web documents transfer.

Knowing the user's browsing history provides us with extra information like the type of the user or his/her preferences. This information about the user can help to improve prediction accuracy in prefetching process. Other demands rise up from the tremendous variety and amount of data presented on the Internet. For users it is demanding to find relevant data. Building user profile can also assist user's navigation to facilitate retrieval of demanded information.

This motivates our research, where we suggest a scheme for reducing the latency perceived by users by predicting and pre-fetching files that are likely to be requested soon, while the user is browsing through the currently displayed page.

## 2 Proposal

The main idea of our proposal works on the presumption that history based pre-fetching does not need to use just the link path history, but can also use a semantic path history. Let's say that a user is searching for features of last automobile X model. The process of information retrieval usually starts by entering a keyword into a search engine. The search engine offers some result links based on the entered keyword and the user starts to evaluate them. The user selects a page from the result list and opens it. In that moment, history based methods for pre-fetching still do not have enough information to predict next step from the current page (there are just two pages in the user's history and so there may be plenty of profiles matching that path). The help in this case may be the keywords extracted from the page.

Probably, there are users searching the same thing, but did not start at the same point - the same page. But at certain point of their path they visited our user's current page. Catching the keywords of visited pages to the link path we can find other users' profiles that were after the same thing but did not follow our user's link path up to now. These profiles can be selected for the web access prediction for current user.

### 2.1 Semantic description of web page

Notice that the Web HTML format was designed merely for document presentation. A challenge is to automatically extract semantic knowledge of HTML documents and construct adaptive semantic nets between Web documents online. Semantics extraction is a key to Web search engines, as well. Unfortunately, current semantics extraction technology is far away from maturity for a general-purpose semantic pre-fetching system. With limited space in this article, we outline the basic idea of annotating the documents with their semantic description.

The approach comes out from the idea presented by [1] who observed that client surfing is often guided by some keywords in anchor text of Web objects. Anchor text refers to the text that surrounds hyperlink definitions (hrefs) in Web pages. They refer

to this phenomenon as *semantic locality*. The authors observed that the anchor text usually gives a truth picture of the linked Web document and used that as the semantic descriptor of it. As well as the authors we intend to use keywords in anchor text of Web objects for web page description. For further processing and, hopefully, with no loss in precision we take into account just nouns that can be found in WordNet lexicon.

As one web page can be, and usually is, linked from many documents there can be found many different keywords while browsing the web. The keywords can be synonyms or can have different meanings and altogether creates the semantic description of the document.

To distinguish different importance of different keywords we establish a weight on keywords. The weight, in general, represents the number of occurrences of the keyword and also the occurrences of the keyword's hypernyms/hyponyms in sense of WordNet's definition. The final algorithm generates the database of Web pages and their semantic description based on the set of weighted concepts (nouns in anchor texts) found in WordNet. The database can be built using crawler as well as using server logs.

### 2.2 Prefetching

The prediction of user's next page will be performed based on the algorithm [see **Alg. 1**]. In general, this algorithm uses current user's browsing history and based on Markov models predicts next page. To predict pages in case when Markov model does not provide enough information, it tries to find the next page based on semantic similarity of user's current page and pages linked to it.

The algorithm assume, that there is available $k$-th order Markov model and that the user has passed an ordered sequence of pages $P_n = (p_0, p_1, \ldots, p_n)$, where, $n < k$. There is also a table $T$ of links and their semantic descriptions as created in previous section: table of pairs $T = \{(p_i, C_{pi})\}$, $C_{pi}$ is the set of weighted concepts describing page $p_i$. Symbol $w_{i,x}$ represents the weight of $x$ in the $C_{pi}$ . The semantic distance is labelled by dist(x,y).

As the semantic distance dist(x,y), we could use the number of nodes (synsets) in the tree structure that were crossed in shortest path between compared words (synsets). But this approach does not distinguish between the case, in which one synset is hypernym of the other one, and the case in which the synsets are siblings. In the first mentioned relationship (hypernyms, hyponyms) between synsets, the synsets are considered closer each other from my proposal's point of view because I need to find pages with similar meaning. So I prefer relationships in sense of hypernyms and hyponyms and I will use the semantic distance as defined in [7], where the author defines recursive semantic distance

**Alg. 1.** Algorithm for prefetching next users' request

```
Input: k-th order Markov model, user browsing history,
table T of links and their weighted sem. descriptions.

Output: the prediction of next page {r_i}

1: {nextPage_i} ← all possible pages found longest match
of sequence P_n in k-th order Markov
2: {prob_i} ← probabilities of all possible next pages
counted based on the longest match model
3: if |P_1|<|P_n| then
4:     {pageSeq_i} ← all sequences of pages from Markov
model ending in the p_n
5:     for all pageSeq_i in {pageSeq_i} do
6:         {C_pageSeqi)} ← acumulate concepts from all pages
in pageSeq_i
7:         if dist({P_n},{pageSeq_i}) > threshold_0 then
8:             {nextPage_i} ← nextPage from appropriate
Markov model
9:         end if
10:    end for
11: end if
12: if |{prob_i}| == 1 then
13:    {r_i} ← nextPage_i
14: end if
15: if |{prob_i}| == 0 then
16:    {nextPage_i} ← all links at current page
17:    for all nextPage_i in {nextPage_i} do
18:        prob_i ← 1/dist(currentPage, nextPage_i)
19:        if prob_i > threshold_1 then
20:            {r_i} ← nextPage_i
21:        end if
22:    end for
23: end if
24: if |{prob_i}|>1 and prob_i>threshold_2 for every i then
25:    for all page nextPage_i in {nextPage_i} do
26:        prob_i ← 1/dist(currentPage, nextPage_i)
27:        if prob_i>threshold_3 then
28:          {r_i} ← nextPage_i
29:        end if
30:    end for
31: end if
```

## 3 Future work

As this is mainly a proposition, the future work involves the implementation of this proposal and determination of constants designed in the proposal. Following the structure of the proposal the implementation will be executed in undermentioned steps.

*Semantic distance*
Experiments must be performed to determine constants for semantic distance. The aim of experiment is to determine which type of semantic measure describes the distance between set of concepts describing web page better for our purpose.

*Base for keywords selection*
The authors in [1] approve that the use of keywords from hyperlink anchor texts is sufficient for document description. Based on experiments with this module the algorithm may be enriched with other sources of keywords used for semantic description. Some pages are already annotated with semantic annotation and also the titles or headlines of Web pages can provide usable keywords. Currently we take into account just the hypernym/hyponym relationship. The experiments may show that more relationships may be used to get better accuracy. The prediction module is the main aim of the whole thesis. The basic proposal of algorithm [**Alg. 1**] will be refined to achieve the best possible performance. The experiments in this module concerns two fields: estimation of the order of Markov model and determination of thresholds used there.

*Estimation of the order of Markov model*
The main purpose of the whole proposal is to lower and prune basic Markov model to simplify its complexity. The lower the order of Markov model the worse accuracy. Using the semantic information I want to lower the order as much as possible. Experiments should establish the best proportion of order and efficacy using semantic description.

*Determination of thresholds*
In the algorithm [**Alg. 1**] the thresholds are mainly used to determine the boundary where it is profitable to pre-fetch suggested Web page. Again, the experiments should establish the best proportion.

## 4 Conclusion

To reach high accuracy for prefetching using Markov models we need to apply higher-order Markov models incorporating many links. The price is sophisticated computation. The suggested approach of use of keyword based history can reduce Markov models' orders as it can exploit the semantic information as well. Also the

problem of 'never visited pages' can be reduced as we can use the approach similar to the keyword-based semantic prefetching presented in [1].

The second application of this link predictor could be system aided web navigation. The link prediction could be used to build a navigation agent which suggests (to the user) which other sites/links would be of interest to the user based on the statistics of previous visits (either by this particular user or a collection of users).

**Acknowledgements.** This research has been partially supported by MSMT under research program No. 6840770014. This research has been partially supported by the grant of the Czech Grant Agency No. 201/06/0648. This research is supported by the internal grant of CTU No.CTU0909313.

## References

1. Xu, Cheng-Zhong and Ibrahim, Tamer I.: A Keyword-Based Semantic Prefetching Approach in Internet News Services, IEEE Trans. on Knowl. and Data Eng., vol.16, No. 5, (2004) 601–611.
2. B. D. Davison. Predicting web actions from html content. In Proceedings of the The Thirteenth ACM Conference on Hypertext and Hypermedia (HT'02, pages 159 -168, 2002.
3. D. Duchamp. Prefetching hyperlinks. In Proceedings of the 2nd USENIX Symposium on Internet Technologies and Systems (USITS'99), 1999.
4. M. Albanese, A. Picariello, C. Sansone, and L. Sansone. Web personalization based on static information and dynamic user behavior. In WIDM '04: Proceedings of the 6th annual ACM international workshop on Web information and data management, pages 80–87, New York, NY, USA, 2004. ACM.
5. F. Khalil. Combining web data mining techniques for web page access prediction. PhD thesis, University of Southern Queensland, AUSTRALIA, 2008.
6. H. Kurian. A markov model for web request prediction. Master's thesis, Kansas State University, Department of Computing and Information Sciences, Kansas, USA, 2008.
7. J. Radek. Automatic ontology linking. In In Innovations'07 Poster Session Proceedings [CD-ROM], pages 17–19. UAE University, 2007.