Information Overload in Non-Visual Web Transaction: Context Analysis Spells Relief

Jalal Mahmud
Department of Computer Science
Stony Brook University
Stony Brook, NY 11794, USA
jmahmud@cs.sunysb.edu

ABSTRACT

Visually disabled individuals use screen readers to browse the Web. Sequential processing of screen readers makes Web browsing time-consuming and strenuous. The problem is further exacerbated when conducting Web transactions (e.g. buying books, paying utility bills, etc.), which typically involve a number of steps spanning several pages. Thus browsing becomes fatigue inducing and causes significant information overload. But usually one needs only small segments of Web pages for completing a Web transaction. Identifying and presenting such segments from Web pages can reduce information overload. An interesting idea is to use context surrounding a link to identify relevant information on the next Web page. I describe how context analysis coupled with Web content analysis can identify relevant content segments. Preliminary results based on my system incorporating this idea, show a lot of promise in combating the information overload problem encountered by visually disabled individuals when they do transactions over the Web.

Categories and Subject Descriptors

H.5.2 [**Information Interfaces and Presentation**]: User Interfaces—*natural language, Voice I/O*

General Terms

Algorithms, Design, Human Factors, Experimentation.

Keywords

Web Transaction, Context, Semantic Concept, Screen-Reader

1. INTRODUCTION

The Web has become an indispensable source of information. The primary mode of interaction with the Web is via graphical browsers, which are designed for visual interaction. As we browse the Web, we have to filter through a lot of irrelevant data. Sighted individuals can quickly segment any Web page and identify the information that is most relevant to them.

Individuals with visual disabilities, have to use screen readers [3, 1], which typically process Web pages sequentially. Although they provide short-cuts to skip segments of text, users still have to listen or skip through substantial page content before they get to the information. Thus Web browsing can cause significant information overload.

This problem is further exacerbated when Web browsing spans several pages as in online transactions, e.g. shopping, registrations, bill payments, etc. In particular, the loss of spatially organized content makes it difficult for users to comprehend the sequence of transactional steps. Thus, combating this information overload problem is critical especially when doing transactions using non-visual interaction modalities such as speech.

In this paper, I describe my ongoing research work on techniques to combat this information overload problem. Broadly, my approach partitions the Web page into segments, and identifies important segments containing semantic concepts (e.g. AddToCart, SearchForm) for conducting Web transaction. The key idea is to use the context surrounding a link (link context analysis is described in [4]) to identify the relevant segment on the next Web page. Such relevant segment contains semantic concept (e.g. Search Result, Item Details) for conducting Web transaction.

I have developed an algorithm to extract other important segments containing semantic concepts (e.g. AddToCart, SearchForm) using a small set of keywords (e.g. addtocart, search, shopping cart, checkout) as base knowledge. Then these relevant segments are presented to the user instead of the entire Web page. Such focused content delivery is the answer to the information overload problem.

2. ARCHITECTURE

Figure 1 shows the architecture of the system. The **Browser Object** module downloads the Web page every time the user requests a new page to be retrieved. **Geometric Analyzer** partitions the Web page into geometric segments containing semantically relevant items.

Context Analyzer is called twice for each Web page access. When the user follows a link, the module collects the context of the link. When a new page is retrieved, the module executes our algorithm in [4] to locate the content segment identified to be the most relevant with respect to the context of the followed link.

Concept Extractor identifies the semantic concept present in the most relevant segment. This module also identifies other concepts using a base knowledge containing keywords related to semantic concepts.

Users communicate with the system through the **Interface Manager**. It uses VoiceXML dialogs to interact with the users and presents the relevant segments in the Web page.

3. APPROACH

Here I briefly describe the main techniques used in the system.

Geometric Segmentation:

Observing that semantically related items exhibit spatial locality in Web pages [5] and share similar geometric alignment, I have developed a geometric Web page segmentation algorithm described in [4], that partitions the Web page into segments containing semantically related items. The algorithm exploits the layout and alignment of different Web page components to partition the Web page into geometric segments. Semantic concepts are contained within such geometric segments.

Context Analysis:

I have developed the context analyzer for CSurf [4], a context-directed non-visual Web browser. On following a link, context analyzer captures the context of the link, employing a simple topic-boundary detection technique [4]. Then, it uses the context and a Support Vector Machine (SVM), a statistical machine-learning model, to identify the most relevant segment on the next Web page.

Concept Extraction:

The semantic concepts (e.g. Search Result, Item Detail) present in the relevant segment are identified by context analysis. To identify other concepts (e.g. AddToCart), a small set of keywords is used as base knowledge. For example, the keywords "addtocart", "shopping cart" are used to identify "AddToCart" concept. The geometric segment that contains such keywords (relevant to a semantic concept) is identified as the segment containing the semantic concept.

4. EVALUATION

I have evaluated the main components of the system. The geometric segmentation and context analysis algorithm have been extensively evaluated. The evaluation result of these algorithms exhibit reasonably high precision and recall (see [4] for evaluation result of context analysis). A preliminary evaluation of the concept extractor for 12 Websites has been done. Extensive evaluation of the concept extractor is work in progress.

I plan to conduct user studies of the integrated system with blind as well as blind and deaf students at Helen Keller Services for the Blind and the Helen Keller National Center in New York.

5. RELATED WORK

This work has broad connections with research in non-visual Web access, Web page segmentation, etc.

Blind users access the Web using screen-readers, such as JAWS [3] and IBM's Home Page Reader [1]. BrookesTalk [7] facilitates non-visual Web access by providing a summary of the Web page, by using NLP-based text abstracting and summarization techniques.

Substantial research has been done on segmenting Web documents [6, 2]. These techniques are either domain specific [6] or depend on fixed sets of manually specified rules [2]. In contrast, my geometric segmentation is scalable over domains and does not depend on manually specified rules.

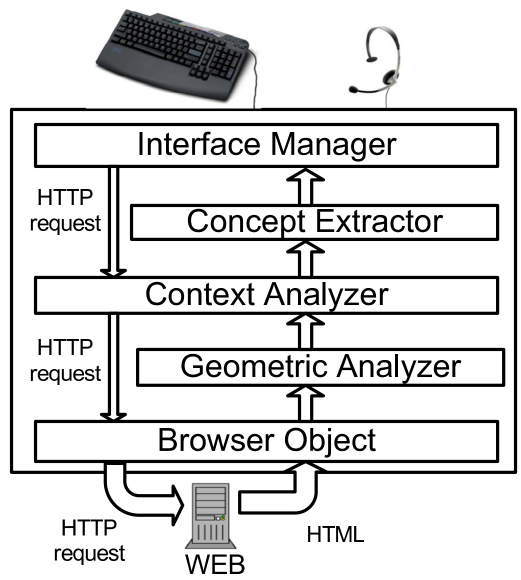

Figure 1: Architecture of the System

6. CONCLUSION

The use of complete Web pages for doing online transactions using non-visual interaction modalities can cause severe information overload. In this paper, I described a novel technique centered on contextual information surrounding a hyper-link coupled with very limited and shallow domain knowledge to combat this information overload problem. Preliminary experimental evidence of my system shows significant promise.

7. REFERENCES

[1] C. Asakawa and T. Itoh. User interface of a home page reader. In *ACM Intl. Conf. on Assistive Technologies (ASSETS)*, 1998.

[2] D. W. Embley, Y. Jiang, and Y.-K. Ng. Record-boundary discovery in web documents. In *ACM Conf. on Management of Data (SIGMOD)*, 1999.

[3] `http://www.freedomscientific.com/`.

[4] J. Mahmud, Y. Borodin, and I. V. Ramakrishnan. Csurf: A context-directed non-visual web-browser. In *WWW '07: Proceedings of the 16th international conference on World Wide Web*, 2007.

[5] S. Mukherjee, G. Yang, and I. Ramakrishnan. Automatic annotation of content-rich html documents: Structural and semantic analysis. In *Intl. Semantic Web Conf. (ISWC)*, 2003.

[6] H. Takagi, C. Asakawa, K. Fukuda, and J. Maeda. Site-wide annotation: Reconstructing existing pages to be accessible. In *ACM Intl. Conf. on Assistive Technologies (ASSETS)*, 2002.

[7] M. Zajicek, C. Powell, and C. Reeves. Web search and orientation with brookestalk. In *Proceedings of Tech. and Persons with Disabilities Conf.*, 1999.