

WEB BROWSING FOR VISUALLY IMPAIRED

Gulden Uchyigit

University of Brighton, Brighton, U.K.

Keywords: Accessibility, web browsing, user interfaces.

Abstract: Availability of information is no longer an issue. The internet is a vast resource where details relating to any subject are available almost instantaneously; however availability does not equal accessibility. The internet is a highly visual medium and users with visual disabilities including blindness and print impairments have no choice but to use sub-standard screen readers or expensive Braille displays. Poorly designed web pages that do not conform to accessibility standards or have complex layouts are bewildering and users have difficulty in distinguishing between useful and irrelevant information. This paper presents a system which allows visually impaired users to interact with a computer in a more effective manner. The system employs speech synthesis as an inexpensive and portable method of output and uses text analysis methods to ease the process of navigation. The system's usability and learnability are demonstrated through end-user testing.

1 INTRODUCTION

The internet is a vast resource that has increased availability of information of information by stratospheric proportions; unfortunately information availability is not proportional to accessibility. The internet is overwhelmingly visual and the advancement of multimedia on the web is not improving the situation. Simple data retrieval operations that would be trivial to most users are comparatively complex for users who have a visual disability such as blindness or a print impairment.

Visually impaired users rely on tactile auditory output to rely on tactile auditory output to interact with computers, however web pages are not designed with these access methods in mind. Users often find themselves overwhelmed by complicated methods in mind. Users usually find themselves overwhelmed by complicated pages that are bloated with non cohesive information, lack structure and are inconsistent with other pages on the same site. Furthermore standard screen users, used to relay text on the screen to Braille reader or speech synthesizer, are confusing as a result of operating a highly visual environment such as the windows desktop. Screen readers do not have knowledge of web technologies and when used in conjunction with a web browser they simply read the pages' text and are unable to differentiate between different page elements or provide further information about them.

It is clear that a visually impaired user would be greatly assisted by a screen reader designed especially for web browsing providing functions that correspond to operations within a standard web browser. Moderate successes have been made in this area with proprietary speech browsers such as Brookes Talk which include large set of functions for browsing web pages and provides limited support for abstraction and page summarisation. However navigation between pages remains as a considerable difficulty.

Navigating through a series of pages without a visual interface requires considerable cognitive capacity. Users cannot quickly gain an overview of the page by skimming the text and they must remember where they have been (in case they need to backtrack), which page they are currently viewing and the position within the page and the information which they are trying to obtain which may be ambiguous or obscure.

This paper presents a tool that will provide blind and visually impaired users the necessary functions to browse the internet and provide additional information by employing text analysis techniques on relevant web pages.

2 BACKGROUND

Web Accessibility. Paciello (Paciello, 1997) refers to the movement of information from paper-based to online resources as "the publishing paradigm shift" and points out that although it has vastly increased the availability, data accessibility is still poor for people with disabilities including visual impairments. Most inaccessibility problems are caused by poor design decisions in the following areas: complex notation, image rendering, multimedia components and navigation. Complex notation, e.g. graphical representation of mathematical formulae or non standard characters, are difficult to represent in a form that can be interpreted by a blind user. Image rendering causes similar problems however the inclusion of alt and longdesc tags can be used to store descriptions of an image. The longdesc tag can be used to store a more comprehensive description of the image (than the alt tag) or link to a page containing the description. Use of the alt and longdesc parameters provided by html can increase the user's understanding of what the image represents. Multimedia components including video are difficult to annotate due the large amounts of visual data they contain, although solutions like descriptive video, which includes an optional narrative sound stream, have been proposed. Navigating between web pages is difficult for visually impaired users who must keep track of where they currently are, the options available to them, the links on the page and be able to distinguish page content from irrelevant data (e.g. advertising). A particular disadvantage is that they cannot quickly skim a document for relevance or jump to the part of the page they require immediately. Online organizations such as the Centre for Applied Special Technology (CAST) provide guidelines for building sites that are accessible and easy to navigate.

A considerable problem for web browsers designed for visually impaired users is poorly written html. Syntax errors such as overlapping or malformed tags can cause ambiguities in how the page should be rendered.

The World Wide Web Consortium (W3C) have introduced standards that aim to unify the way web pages are written and interpreted and provide validation services where designers can ensure that their pages meet these standards. W3C's cascading style sheet (CSS) standard has also made information extraction easier by separating design and content. When using CSS style information is included in a separate file and not defined by tags

within the html. The use of CSS simplifies the html documents and reduces the scope for error when extracting data.

Designing Software for Disabled Users. Stephanidis (Stephanidis, 1997) explains how the existing tools for disabled users are "adaptive solutions" whose primary function is to extend an existing piece of software to incorporate some new functionality that improves disabled user interaction. Such adaptations can be categorized in three different levels:

Alternative access adaptations extend an existing solution to provide support for specialist input/output equipment such as Braille displays or screen magnifiers.

Information content adaptations involve modifying the representation of data to make it more accessible. Such an adaptation may extend html to make it more accessible to text-to-speech engines by adding further descriptive tags to the visual elements of a page.

User Interface adaptations involve changing the way users interact with an application in order to cater for the needs of a specific user group. Such an adaptation may alter a web browser to render web pages without images and with large text so that they may be easily read by those who are visually impaired.

These adaptive approaches allow developers to take advantage of the pre-existing functions present in an application, which they would not have the time or resources to implement themselves. However it can be technically difficult to adapt a piece of software to perform a different task to which it was originally intended, especially if the code is highly complex or proprietary.

| CONSONANTS (PULMONIC) | | | | | | | | | | | | | | | | | | | |
|-----------------------|----------|---|-------------|---|--------|----------|---|--------------|-----------|---------|-------|--------|------------|---------|---|---|---|---|---|
| | Bilabial | | Labiodental | | Dental | Alveolar | | Postalveolar | Retroflex | Palatal | Velar | Uvular | Pharyngeal | Glottal | | | | | |
| Plosive | p | b | | | | t | d | | ʈ | ɖ | c | ɟ | k | g | q | ɢ | ʔ | ʕ | |
| Nasal | | m | ɱ | | | | n | | ɳ | ɲ | | ŋ | ɴ | | | | | | |
| Trill | | | | | | | r | | | | | | ʀ | | | | | | |
| Tap or Flap | | | | | | | ɾ | | ɽ | | | | | | | | | | |
| Fricative | ɸ | β | f | v | θ | ð | s | z | ʃ | ʒ | ʂ | ʐ | x | χ | ʁ | ħ | ʕ | h | ɦ |
| Lateral fricative | | | | | | | ɬ | ɮ | | | | | | | | | | | |
| Approximant | | | | ʋ | | | ɹ | | ɻ | j | | ɰ | | | | | | | |
| Lateral approximant | | | | | | | l | | ɭ | ʎ | | ʟ | | | | | | | |

Where symbols appear in pairs, the one to the right represents a voiced consonant. Shaded areas denote articulations judged impossible.

Figure 1: Symbols used in International Phonetic Alphabet to represent consonants.

3 SYSTEM OVERVIEW

The implemented system is a set of tools which are able to read the contents of the web pages to the user as the user is browsing the web. The system is also able to read out a summary of pages pointed to by url of the page which is being browsed. The user is able to “jump” from one url to another url when examining the contents of a web page and request to have a short summary to be read out of the pages which the url is pointing to. This makes the browsing of the page quicker for the user.

The system is composed of three runnable applications: a command line interface, a stand alone graphical interface and a web interface which allows the system to run along side a visual rendition of the current site (see Figure 2).



Figure 2: Command Line interface running along side a web browser.

A graphical interface can also be initiated by using the command StartGraphicalInterface. This will then present the user with a GUI with buttons relating to the functions as before.

The web interface interacts with voiceServer using TCP/IP sockets. The tool bar appears at the top of the web browser, the same function keys can be used for the user to control the interface.



Figure 3: Web Interface.

Command-line browser

The command line browser can be started by navigating to the directory of the installation and

typing startCommandLine. The user is then presented with the title of the page. The system will automatically begin reading the system. Several function keys can be used to can be used to control navigation around the current page.



Figure 4: Command Line browser.

The following function keys can be used to control navigation around the current page.

| Function Key | Action |
|--------------|---|
| > | <i>Skip forward</i> – Move onto the next page. |
| < | <i>Skip Back</i> – Move back before the text currently being read |
| r | <i>Restart</i> - Start reading the document |
| j | <i>Jump</i> – Start reading form the next marker on the page |
| q | <i>Quit</i> – Exit the program |
| p | <i>Pause</i> – Pause/resume the voice synthesizer |
| g | <i>GoTo</i> – Load an new user defined web site |
| l | <i>Enter Link Navigation Mode</i> |

4 IMPLEMENTATION DETAILS

All of the system components were implemented using Java 1.4.2 Standard Edition .

Speech Synthesizer

FreeTTS (FreeTTS) was the text-to-speech engine used to generate the synthetic voice used by the system. Although there were several Text-to-Speech engines available FreeTTS was the only Open Source Java implementation. FreeTTS was accessed through the Java Speech API{cite{JSAPI}} (JSAPI), a standard developed at Sun for invoking commands on Java text-to-speech implementations. FreeTTS could be potentially be replaced by another JSAPI compatible implementation without modification of the code.

5 EVALUATION

Usability

In an attempt to test the usability of the system a set of five visually impaired volunteers were asked to use the system. Test-subjects were initially asked to use the integrated browser tool; this tool operates within a browser and is presented above a full rendering of the web site, it also provides the user with a series of buttons clearly showing which functions are available to them. This enables the test subjects to rapidly become familiar with the functionality and the limitations of the system (usually less than two minutes). This period of learning would take longer for visually impaired users, who would have to read a Barillle manual to get familiar with the system. The test subjects were asked to browse the BBC low-graphics news website for five minutes to get used to using the system.

The test subjects were asked to stop using the graphical version of the system and switch to the interface that uses a series of function keys to accept user input and does not provide any form of output apart from synthesized speech; they were then asked to return to the BBC news web site and continue browsing. The subjects were then given the task of retrieving the screening schedule for the Kensington Odeon. This is the typical task that would not pose a problem to any visually abled user with a small amount of internet experience, however the task involves navigating through a series of pages each containing images, links, large amount of text and complex formatting data. The inclusion of a specific goal also introduced the quantitative results relating to whether the goal was achieved and the time taken to achieve it.

The times taken to complete the tests are shown in the Table 1. All the test subjects reached the destination page however the times ranged considerably.

Table 1: Time taken to navigate through Odeon web site.

| Test Subject | Experience Level | Goal Achieved | Time (minutes) |
|--------------|------------------|---------------|----------------|
| 1 | moderate | yes | 3.75 |
| 2 | high | yes | 2.50 |
| 3 | very high | yes | 4.13 |
| 4 | very high | yes | 5.22 |
| 5 | high | yes | 6.48 |

In discussion with the test subjects feedback concerning graphical interface was overwhelmingly positive and all were in agreement that if used in conjunction with the browser's built in features, such as text size and font/colour modification, it

would be useful for partially sighted users. Some users were initially confused about the functions associated with particular buttons, especially "jump" and suggested that more descriptive labels be used.

Most users felt that the page navigation options were sufficient and felt that skipping between tags were more appropriate than skipping words or sentences at a time and that if these features were incorporated it would be harder to learn all of the functions. Users were particularly impressed by "jump" capability which allowed the user to skip over large sequences of links and navigational options irrelevant to the page's subjects.

REFERENCES

- Paciello, M. "People with Disabilities Can't Access the Web" World Wide Web Journal Volume 2 , Issue 1 1997
<http://www.mardiros.net/accessible-web-design.html>
- Stephanidis, C. et al. "Supporting Interface Adaptation: the AVANTI Web-Browser" 3 rd ERCIM Workshop on User Interfaces for All, Alsace, France, November.", 1997
<http://freetts.sourceforge.net/>
- Zajicek, M, Venetsanopoulos, I. and Morrissey, W. "Web access for visually impaired people using active accessibility" International Ergonomics Association 2000/HFES 2000, San Diego
- Montaner, M. et al. "A Taxonomy of Recommender Agents on the Internet"
- P. Roth , L. S. Petrucci and A. Assimacopoulos and Thierry Pun "Audio-Haptic Internet Browser and Associated Tools for Blind and Visually Impaired Computer Users" Workshop on Friendly exchanges through the Net March 22-24 2000.