INTERNATIONAL JOURNAL OF SPEECH TECHNOLOGY 6, 301–309, 2003

Evolution of the Information-Retrieval System for Blind and Visually-Impaired People

SIMON DOBRIŠEK, JERNEJA GROS, BOŠTJAN VESNICER, NIKOLA PAVEŠIĆ AND FRANCE MIHELIČ
LUKS, Faculty of Electrical Engineering, University of Ljubljana, Slovenia
simond@fe.uni-lj.si

Abstract. Blind and visually-impaired people face many problems in interacting with information retrieval systems. State-of-the-art spoken language technology offers potential to overcome many of them. In the mid-nineties our research group decided to develop an information retrieval system suitable for Slovene-speaking blind and visually-impaired people. A voice-driven text-to-speech dialogue system was developed for reading Slovenian texts obtained from the Electronic Information System of the Association of Slovenian Blind and Visually Impaired Persons Societies. The evolution of the system is presented. The early version of the system was designed to deal explicitly with the Electronic Information System where the available text corpora are stored in a plain text file format without any, or with just some, basic non-standard tagging. Further improvements to the system became possible with the decision to transfer the available corpora to the new web portal, exclusively dedicated to blind and visually-impaired users. The text files were reformatted into common HTML/XML pages, which comply with the basic recommendations set by the Web Access Initiative. In the latest version of the system all the modules of the early version are being integrated into the user interface, which has some basic web-browsing functionalities and a text-to-speech screen-reader function controlled by the mouse as well.

Keywords: information retrieval system, spoken dialogue system, web browser for the blind

1. Introduction

Modern information technology facilities are often not suitable for blind and visually-impaired people. Such problems in communication are well known to many people with disabilities. If they are unable to use their hands, read or speak, they are forced to use technical aids to overcome their problems (Levitt, 1995). For blind or visually-impaired persons the Braille coding of texts is still a common aid in many countries all over the world. This type of coding requires special editions of written corpora or special additional hardware components when used with computers. The solution is relatively costly and requires special skills from the user.

Over the past ten years a considerable advance has been made in the development of automatic text-to-speech and speech-recognition systems. Such systems offer a more natural and user-friendly way of communication for a blind and visually-impaired persons; the communication goal can be achieved faster and they offer access to large text corpora via modern technical equipment (over computer networks, scanners, etc.) and have a relatively low price (Zajicek et al., 1999).

However, these new technologies are very language-dependent, and general solutions for all languages cannot be applied directly (Jelinek, 1998). If speech technologies are to be used with the Slovene language, the language-dependent parts of the systems must be developed for this purpose using knowledge of Slovene-language phonology, syntax and semantics.

Spoken-language technologies have been one of our main research activities for more than twenty years. Our prime interest is to develop a core technology for the Slovene spoken language that can be customised for different kinds of applications. We found the development of an information retrieval system for Slovene-speaking blind and visually impaired people important to our research for several technical and non-technical reasons, among them is the possibility to help people with disabilities.

In the article, a voice-driven text-to-speech dialogue system is presented. It was developed for reading Slovenian texts obtained from the Electronic Information System of the Association of Slovenian Blind and Visually-Impaired Persons Societies. The early version of the Homer system is a console application designed as a spoken dialogue interface to the Electronic Information System. Later, the Homer system was rebuilt as a specialized web browser that uses speech modules from the previous version.

First, the Electronic Information System is presented. This simple Internet database of text files represents a central information source for the Homer system. We also describe how blind people have accessed this database thus far just to provide an impression of what the Homer system tries to improve.

2. The Electronic Information System

The Electronic Information System (EIS, 2002) is an Internet database of text files available at the Internet site of the Association of Slovenian Blind and Visually-Impaired Persons Societies (ZDSSS). Even though the database is accessed via the HTTP Internet protocol it mainly contains plain text files and compressed archives of texts.

At the EIS members of the ZDSSS can find a collection of Slovenian periodicals that is updated daily, examples of classic Slovenian literature, catalogues of all the available texts at the site, a selection of technical manuals, some legal and legislative texts, and other information that is important to the ZDSSS members. The EIS database has access restricted to the ZDSSS members since many texts fall under copyright restrictions.

The majority of the text files is stored in a non-tagged plain format suitable for Braille embossers. Slovenian characters are coded using the Slovenian version of the ASCII coding or the Microsoft Windows code page 1250. The database also contains some texts that are stored in the Microsoft Word format and are intended to be read by visually-impaired people as Microsoft Word can display the texts in magnified fonts and high-contrast colours.

The EIS is old and limited. However the ZDSSS members are very used to it. Nevertheless, in line with the latest trends, this Internet database needs to be arranged differently. It would be much more convenient if the available texts were formatted as common HTML pages. However, there are many obstacles that prevent the technical staff of the EIS from achieving this. First of all, there are many different providers of the texts and many of them have established a routine procedure of acquiring the text files in a plain format. Some of them even use scanners together with optical character-recognition programs for this purpose. Added to this, a lot of ZDSSS members do not want to change the format, because they are very used to the existing one.

The main Slovenian press companies are obliged by law to provide promptly a special version of their periodical press to the blind and visually-impaired community. But again, they are not obliged to provide it in a specific format. A majority of them have a web-release version of their daily newspapers and magazines published on the Internet. However, these web releases are usually not accessible to blind people as they contain too many graphical elements.

Presently, all the daily newspapers and magazines available at the EIS are provided as compressed (zipped) archives of plain text files that are arranged according to newspaper pages or headings. The ZDSSS members access the EIS database using a conventional web browser on a PC that is equipped with expensive assistive technology, such as JAWS for Windows, that reads information from a computer display and sends it to a Braille embosser or speaks it to the user through a speech synthesiser.

First, they login to the EIS and then navigate through the EIS HTML pages until they find a link to a text file or a compressed archive of text files in which they are interested. Then they download and unpack the selected file to their PC and open it with a text editor. Finally, they browse through the text and read it using the text editor together with the above mentioned assistive technology. Until recently, they have used only Braille embossers as the primary output device, since there was no Slovenian speech synthesiser available to be used with this technology. A group of researchers at J. Stefan Institute has recently developed a Slovenian speech synthesis module (Šef et al., 2000) that can be used with JAWS for Windows.

Our main objective is to improve this expensive and rather complicated way of accessing the EIS texts.

3. The Homer System

The early version of the Homer system was designed exclusively as an interface to the EIS Internet database.

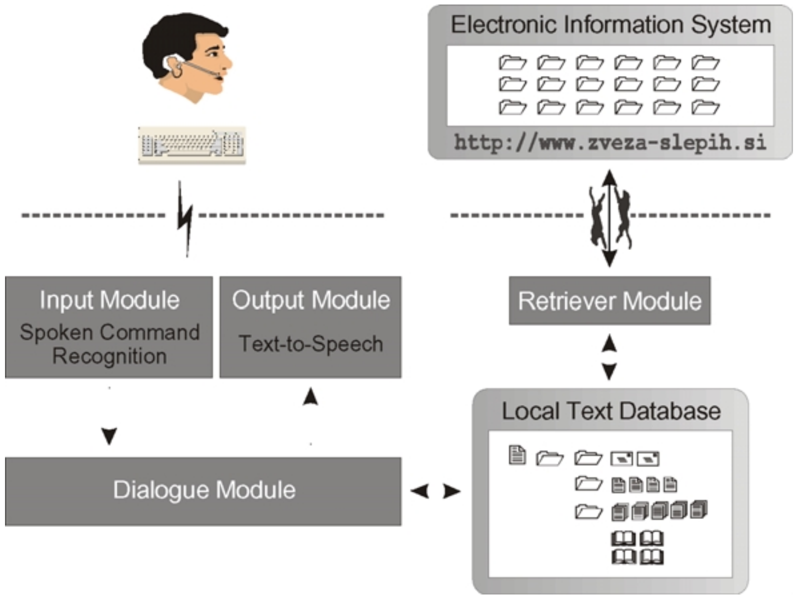

Figure 1. The structure of the early version of the Homer system.

The whole system consists of four main modules (Fig. 1). All the modules have evolved together with the whole system.

The first module enables Internet communication and retrieves text to a local disc. This so-called retriever module arranges the local text database using a content-type tagging of the plain text files retrieved from the EIS. The early version of this module still uses a non-standard tagging that is similar to XML.

The next module is the one that manages dialogues with users and performs access to the text database. The previously developed text-to-speech output module (Gros et al., 1997) enables automatic generation of Slovene speech from an arbitrary Slovenian text written in a non-tagged plain text format using one of the standard character codings, like the Slovenian version of the 7-bit ASCII coding or the WIN-1250 and the ISO-8859-2 codings. Input to the system is performed via a keyboard with some specially selected keys or by using the speaker-independent spoken command recognition module that runs in parallel with the other modules and was developed using the previously existing speaker-independent acoustical models (Dobrišek, 2001).

The Homer system was designed to be implemented on a standard PC with a minimum of 32 MB of RAM with a built-in standard 16-bit sound card and a standard headset with a close-talking microphone. The early version of the system was ported from Linux with a DOS emulator (Dobrišek et al., 1999) to the Microsoft Windows 9x/ME/NT/2000/XP operating systems. For optimal performance it uses multi-threading and other advantages of the 32-bit environment. It requires approximately ten MB of disk space for the program code and the text-to-speech and speech recognizer module inventory, as well as additional storage facilities for the locally cached text database.

3.1. *Internet Communication*

The strategy of the text-file acquisition task was switched from the specially designed modem-communication protocol in the very first version to the standard Internet communication protocol. The system was built for users who are connecting to the Internet from their home via a modem and their own computer using the Microsoft Windows operating system. Normally, the connection is established via dialogue boxes using a mouse and special keys and entering passwords and some other keywords. Since we could not expect this procedure to be performed by blind or visually-impaired people, a special software solution was designed to establish the connection and automatic text-file acquisition. The Microsoft product called RAS API (Remote Access Services Application Programming Interface) (Skonnard, 1999) was used to build the interface. The interface incorporates a feature that enables the establishment of a dial-up connection with the PPP (Point-to-Point Protocol) via the Internet provider and is a part of the Microsoft Windows 9x/ME/NT/XP/2000 operating systems.

In this version of the system the acquired text files are locally arranged in a database as a tree structure. The database is actually just a predefined directory structure of tagged text files. The first level represents the index of all the available texts' categories (e.g., instructions, newspapers, books, the ZDSSS documents) retrieved from the EIS. The second level consists of the table of contents (titles) for the text categories from the first level. The contents can be arranged in two different modes, using topic descriptions or page numbers. The third level includes the texts (articles), which are tagged in the form: title, subtitle, date, author, and body of the texts separated into paragraphs. Each of the database tree nodes represents a text category, a text itself, or a part of a text.

3.2. *Input Module*

The input interface manages the keyboard entry and/or spoken-command recognition. Each of the spoken control commands is associated with its accelerator key on a standard PC keyboard.

Speech recognition is based on well-known Hidden Markov Models (HMMs) (Huang et al., 1990). Models of phone-like units are used as the fundamental models in a silence/commands/silence HMM graph. The number of HMM states differs for each of the spoken commands.

In this version of the system discrete HMMs were used. The modelling was based on the prior vector quantisation of incoming acoustic feature vectors into 128 clusters using the k-means algorithm (Dobrišek et al., 1994). Spoken command recognition was speaker-dependent and yielded recognition accuracy of 95%, on average. The accuracy was estimated using a clean speech database of ten test speakers who uttered 100 training and 50 test commands each.

The main disadvantage of the early version was that it required a special training procedure from a new user. Additional programs for fast and user-friendly procedures of training sample collection and the whole training process were developed (Dobrišek et al., 1998).

Later, we introduced a speaker-independent spoken command recognition module based on tied-mixture continuous HMMs of fundamental phone transition-like units (Dobrišek, 2001). A number of improvements to the acoustic modelling were also introduced. Variable HMM structures were implemented, and a unique initialisation of the model parameters using a Slovene diphone speech database was used. The parameters of the fundamental models were estimated from the Gopolis spoken language database (Dobrišek et al., 1998), which contains several hours of speech from 50 speakers. Using this large database the fundamental HMMs were made speaker-independent.

The speech-recognition module is designed in an open manner, enabling fast adaptation to different applications with isolated spoken command recognition input, and also for larger vocabularies of up to several hundred words. The recognition procedure also offers the rejection category classification, which activates a request for repetition of the command. The latest version of the speech-recognition module allows use of a spoken command grammar which is translated into an HMM graph.

A preliminary off-line evaluation of the spoken command module recognition accuracy, using a clean speech database of ten test speakers that uttered 200 commands each, yielded an average recognition error rate of 2%. However, the actual recognition rate is strongly dependent on the spoken command grammar and the user's behavior while interacting with the system. In practice, it is estimated that the online average recognition error rate increases, but remains below 5%.

3.3. *Dialogue Module*

This module manages dialogues with users, accesses the local text database and performs the system-control function. Its design is based on our experiences with the dialogue module in another speech-recognition system (Ipšić et al., 1995).

As the locally arranged text database is in the form of a tree, the dialogue module enables transitions among all the tree nodes at any stage of the processing and navigation through a list of sub-nodes at each of the tree nodes as well. There are only three main actions that the dialogue module takes or offers to user. These actions are: opening of a database tree node, navigating through a list of sub-nodes, and closing a database node. Each of the database tree nodes represents a text category, a text itself, or a part of a text.

All the actions either can be offered to the user or are taken immediately after the dialogue module successfully interprets a recognized spoken command or a keyboard entry, even though the dialogue is still in the process of describing the current position in the database tree or is in the process of reading a part of the text. As a result, two different dialogue strategies were implemented. When an action is explicitly offered then a *yes/no* answer is expected from the user. This dialogue strategy is called *passive-user behavior*. On the other hand, an *active-user behavior* strategy means that the dialogue module expects the user to interrupt the dialogue process with spoken commands.

The first strategy is more suitable for beginners; the second is more suitable for expert users since it enables faster navigation. Both strategies must be combined when the dialogue is in the process of reading a selected text. In this case the user is allowed to interrupt the reading process occasionally with commands.

As an example, a typical *passive-user* dialogue with the system would be something like the following (user commands are written in bold text):

"Welcome to the Homer system!"
"Shall I open the ZDSSS messages ..."
"***Skip!***"
"Shall I open the daily newspaper section?"
"***Yes!***"
"Shall I open The Independent?"
"***Yes!***"
"Shall I open the News ..."
"***Next!***"
"Shall I open the Sports page?"
"***Yes!***"
"Shall I read the article: Mika won again?"
"***Yes!***"
"The selected article has 456 words. The subtitle is ..."
"***Skip!***"
"Mika had a brilliant race and succeeded to burst ..."
"***Close!***"
"Shall I read the article: Keep the gold coming?"
"..."

The dominantly *passive* dialogue strategy requires only *yes/no* answers and five additional navigation commands. Their English translations are: "*Open!*", "*Close!*", "*Skip!*", "*Previous!*" and "*Repeat!*". We found that this dialogue strategy provides a very comfortable interaction with the system, even though it has proved to be rather time-consuming and even annoying to expert users.

The *active* dialogue strategy requires some additional navigation commands. In the best case we could use a dynamic spoken command grammar instead of using just a static list of commands. The early version of the speech recognition input module did not allow us to use the dynamic spoken command grammar. Thus we carefully selected a small number of navigation commands. We found that the navigation speed is very comfortable when using only eight basic commands. Their English translations are: "*Open!*", "*Close!*", "*Skip!*", "*Previous!*", "*Repeat!*", "*Restart!*", "*Pause!*", and "*Resume!*". With these commands a user can open and close database tree nodes and navigate through a list of sub-nodes at any position in the tree.

A typical *active-user* dialogue now would be something like:

"Welcome to the Homer system!"
"I have the ZDSSS messages – daily newspapers–..."
"***Open!***"
"I have The Daily Star – The Independent – ..."
"***Open!***"
"I have the News – the Sports – ..."
"***Open!***"
"I have the articles: Mika won again – ..."
"***Open!***"
"The selected article has 456 word. The subtitle is ..."
"***Skip!***"
"Mika had a brilliant race and succeeded to burst ..."
"***Repeat!***"
"Mika had a brilliant race and succeeded to burst ..."
"***Close!***"
"I have the articles: Keep the gold coming – "
"– Hughes Takes Gold – ..."
"***Restart!***"
"I have the articles: Mika won again – "
"– Keep the gold coming – ..."
"***Open!***"
"..."

Please note that the newspaper titles in the above examples do not really exist in the EIS database. The Slovenian newspapers are Delo, Dnevnik, Večer, etc.

The latest version of the dialogue module supports only the two described dialogue strategies. The presented list of basic navigation commands was extended with the names of the most frequently accessed sections at the EIS. Thus, the navigation commands: "*Open daily newspapers!*", "*Open The Daily Star!*", etc., are now supported. By using these commands the speed of navigation is increased even more.

3.4. Output Module

For the automatic conversion of the output text into its spoken form the first Slovenian text-to-speech module called S5 (Gros et al., 1995) based on diphone concatenation, was applied. The non-tagged plain text is transformed into its spoken equivalent by several submodules. A grapheme-to-allophone sub-module produces strings of phonetic symbols based on information in the written text. A prosodic generator assigns pitch and duration values to individual phones. The final speech synthesis is based on diphone concatenation using TD-PSOLA (Moulines and Charpentier, 1990).

The task of building a text-to-speech synthesis system for the Slovene language involved some specific challenges. Slovene speech prosody is unique and differs greatly from the prosody of other spoken Slavic languages. Prior to our work, systematic prosody measurements of Slovene spoken language were almost non-existent so we had to start from scratch. An additional obstacle proved to be the lack of an overall pronunciation dictionary for Slovene words with indicated stress positions. Namely, the position of a stressed syllable in a Slovene word hardly obeys any rules.

The quality of the synthetic speech generated by the output module was evaluated in terms of naturalness and intelligibility. The experiment was performed according to the ITU-T Recommendation P.85, which defines a testing method for evaluating the subjective quality of synthetic speech in real-application voice servers available to Public Switched Telephone Network subscribers. The method takes into account both the performance and the attitudes of the users. The attitudes are assessed through the use of multiple scales.

The subjects that were involved in this experiment were not blind, since this evaluation was performed only for the output module alone. They were asked to fill in different templates in their response sheets related to the chosen application domain based on the information they heard. The application domain chosen was airline-timetable information retrieval. Over 90% of the templates were filled in correctly. The incorrectly understood items mainly were the names of foreign airports, quite unknown to the audience and difficult to spell. About two-thirds of the test subjects considered that the TTS system was appropriate for use in an information-retrieval system. The remaining third often commented that although the synthetic speech quality was good enough they strongly opposed the process of machines taking over human work.

The second part of the test served to compare several features describing the synthetic voice quality to those describing the quality of natural speech distorted with different levels of Gaussian noise. The experiment was carried out according to ITU-T Recommendation P.81, which describes a method of comparing synthetic speech to natural speech that is distorted by a modulated noise reference unit. The synthetic speech received a mean opinion score that was between the distorted natural speech with a SNR ratio of 5 dB and 10 dB.

4. The Latest Version of the System

The EIS system is old and has limited functionality. Further improvements to the system became possible with our decision to transfer the EIS text corpora to the new web portal called Kalliope, and to rearrange the text database. In the latest version of the system the text files are not retrieved directly from the EIS anymore but from the web portal Kalliope, where the text database is arranged as a structure of common HTML/XML pages. The relationships among the EIS, Kalliope and the latest version of the Homer system are shown in Fig. 2.

Consequently, the development strategy of the Homer system was also changed. We soon found that an HTML/XML parser is the most critical part of the new system. Even though all the web pages at Kalliope were planned to be simple and to use only a few basic HTML tags with a few additional XML tags, we decided not to develop our own HTML/XML parser but to use one we found in an existing publicly available web browser. This led us to the idea that the Homer system could be rebuilt as a specialized and simplified web browser.

There are many research and development efforts under way to develop so called voice browsers for the web. These browsers will allow any telephone to be used to access appropriately designed web-based services, and will be suitable for people with visual impairments or those needing web access while keeping their hands and eyes free for other occupations. The World Wide Web Consortium (W3C) has been a leader in these activities (Voice Browser Activity, 2002).

We decided not to follow the W3C specifications and guidelines for developing voice browsers, as our prime interest is to develop a small self-voicing web browser designed for blind users for accessing common web pages. Such a browser would never be used over a telephone line and, in addition, it needs to have a mouse-driven screen reader.

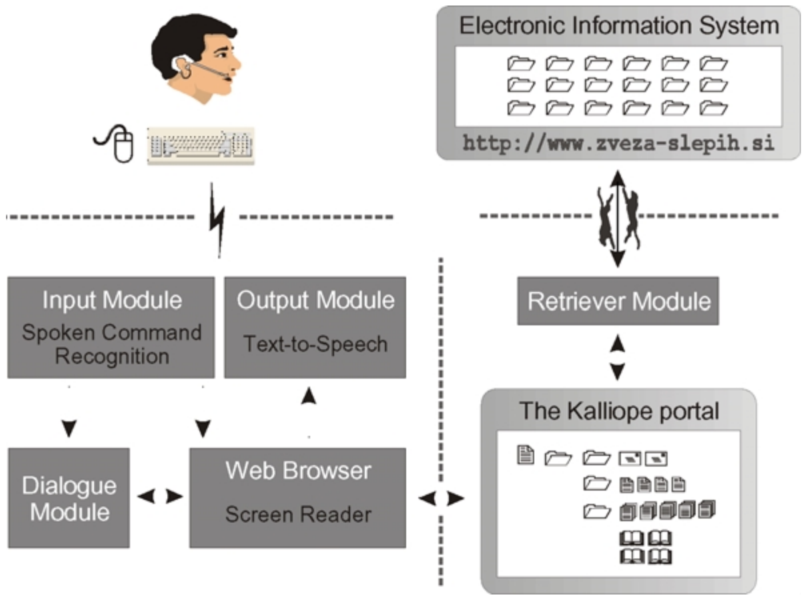

Figure 2. The structure of the latest version of the Homer system.

4.1. *The Kalliope Web Portal*

The web server called Kalliope has a long-term ambition to become a specialized web portal for blind and visually-impaired people in Slovenia. The Kalliope web portal is planned to retrieve the contents of the EIS. All the web pages at Kalliope will comply with the basic recommendations set by the Web Access Initiative (2002) and will be tagged with a few additional XML tags, which will enable user-friendly navigation using the presented dialogue module. The portal will also serve as a site that links to other web sites in Slovenia that are important to the blind and visually-impaired community and are accessible via the Homer system. The portal will have its access restricted to ZDSSS members since many texts from the EIS database fall under copyright restrictions.

Our first task was to reformat the EIS text corpora and to transfer them to the new portal. The majority of the text files at the EIS database are stored in a plain, non-tagged text format, so a special HTML/XML tagger is needed to convert these texts into a structure of common HTML/XML pages. Virtually all of the available text files at the EIS require a unique tagger function for this conversion because the texts are provided from different sources. Presently, scripting programs for such conversions are being developed.

Initially, we concentrated on Slovenian daily newspapers, which probably are the most interesting and the most frequently accessed texts at the EIS. The scripting programs automatically retrieve the original compressed newspaper text archives from the EIS. A tagger function, specially designed for each of the newspapers, then forms the structure of the HTML/XML pages. The HTML/XML structure is formed and refreshed at the Kalliope server every few hours. The first page contains links to issues for all weekdays and a link to the most recent issue. The sub-pages contain links to the newspaper heading pages with links to the individual article pages. All the pages contain hidden XML tags, which are required for the dialogue module to make a distinction between different parts of text. Examples of such pages for the *Delo* daily newspaper are shown in Fig. 3.

4.2. *The Homer Web Browser*

As the Kalliope text database is in the form of common HTML/XML pages the latest version of the Homer System was rebuilt as a specialized and simplified web browser. The Homer web browser was not built from scratch. The existing modules simply were introduced into the source code of one of the publicly available web browsers. When seeking the appropriate web browser we considered the following criteria:

- The source code has to be written entirely in C.
- It has to be a multiplatform browser.

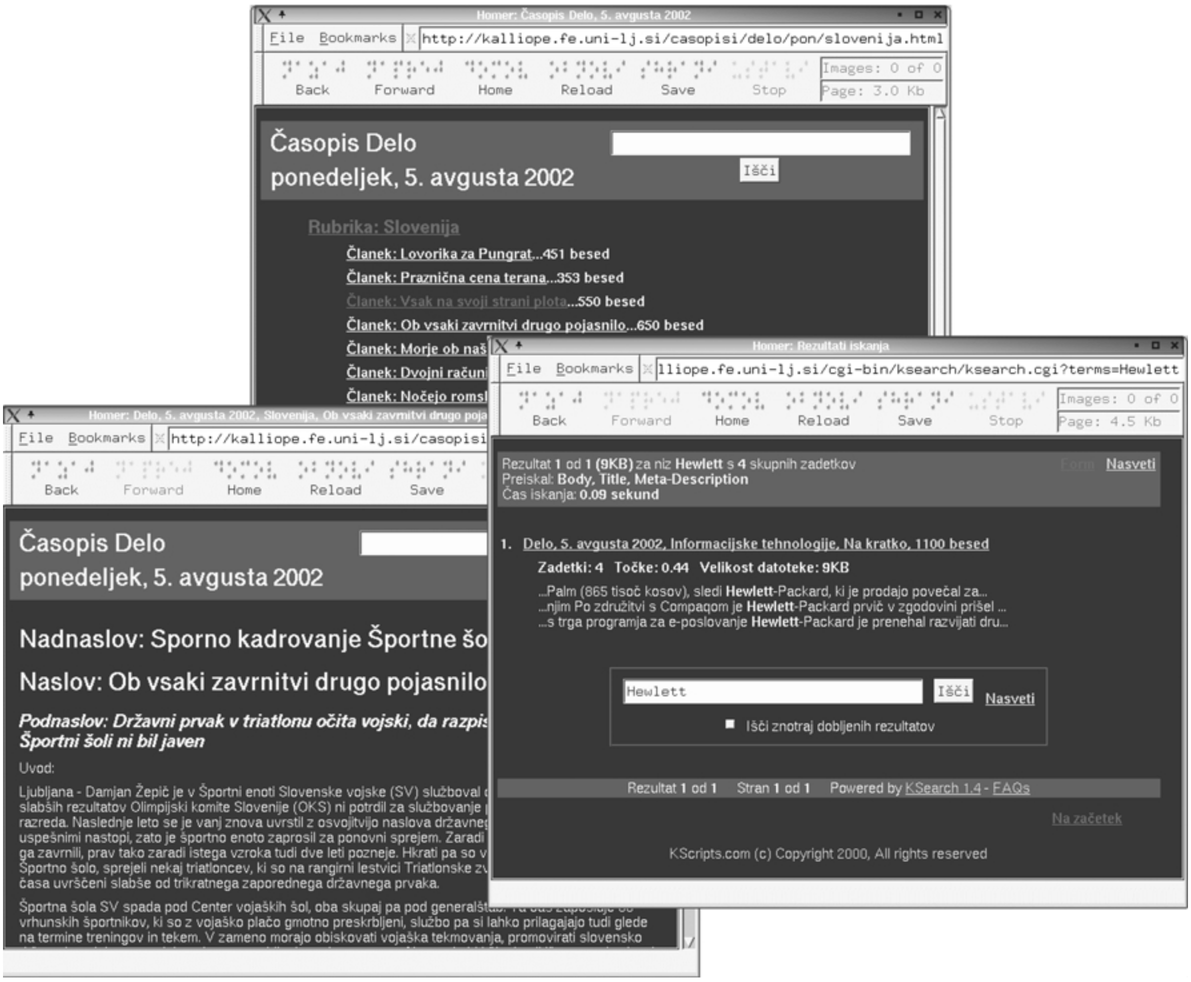

Figure 3. Examples of Slovenian daily newspaper web pages on Kalliope as “seen” with the Homer web browser. The upper page contains a list of articles for the heading *Slovenija* and the left bottom page the selected article. The right page shows the result of a search query.

- It has to be small, stable, developer-friendly, usable, fast, and extensible.
- It has to be a free software project in the terms of the GNU general public license.

We found that the GTK web browser Dillo (2002) was perfect for our needs.

Our first step was to add a screen reader function to the existing Dillo source code. The built-in screen reader is now triggered by pointing the mouse and uses the text-to-speech module for its output. When a user stays for a moment at a certain position on the web page the text beneath the pointer is sent to the output text-to-speech module. The output module works in a separated thread with a time-out function that prevents the user from overfilling the synthesis buffer with a fast pointer motion when browsing through the web page.

An important feature of the screen reader is that it generates special distinctive non-speech sounds which inform users about changes to the current positions of the mouse pointer. Such sounds are generated when users leave or enter a particular area of the displayed web page.

The screen reader function supports not only text parts of common web pages but some basic graphic objects as well, such as non-animated images, lines, bullets, buttons and input text fields. When a user stays for a moment with a mouse pointing at such a graphic object, the system sends a short description of the object to the text-to-speech module. An example of such a description would be: "*Button labelled 'send', sized* 60×20 *pixels*!".

The screen reader works in several different modes. It can read individual words, sentences, lines and paragraphs of the displayed web page. It can read page headings and the whole page as well. The reading mode can be changed by using the function keys on a standard PC keyboard.

Even though a screen reader that is controlled by a mouse is the most important improvement of the new system, the voice-driven dialogue function from the previous versions of the Homer system will remain. A structure of common web pages can always be presented as a tree structure, and the developed dialogue module together with the input module can be used for navigation. At the moment, the existing dialogue module is being introduced into the Homer web browser.

5. Conclusions and Future Work

The development of the Homer system and the Kalliope portal is still in progress. We expect the system to evolve towards a specialized web browser with a mouse-driven text-to-speech screen reader and a voice-driven dialogue manager that handles all the web pages arranged at the Kalliope portal or at sites that are linked from this portal.

Improvements in the sense of more accurate and robust speech recognition and a user-friendly system to control high-quality speech synthesis are planned for the future. Work on speech recognition that incorporates a larger dynamic spoken command grammar is already under way. To improve the synthetic speech some additional measurements and research in the field of micro and macro prosody modelling of Slovene speech should be done as well as recordings of new diphone databases with additional speakers. The whole system also needs to be evaluated in the near future.

Acknowledgment

This work was partly supported by grant no. 3411-00-22 2109 from the Slovenian Ministry of Education, Science and Sport, the Association of Slovenian Blind and Visually-Impaired Persons Societies and the HP Voice Web Initiative philanthropic project http://webcenter.hp.com/grants/.

References

Dobrišek, S. (2001). Analysis and recognition of phones in speech signal. Ph.D. Thesis (in Slovene), University of Ljubljana, Slovenia.

Dobrišek, S., Gros, J., Mihelič, F., and Pavešić, N. (1998). Recording and labelling of the GOPOLIS Slovenian speech database. *Proceedings of the First International Conference on Language Resources and Evaluation*. Granada, Spain, vol. 2, pp. 1089–1096.

Dobrišek, S., Gros, J., Mihelič, F., and Pavešić, N. (1999). HOMER—a voice-driven text-to-speech system for the blind. *ISIE'99 Proceedings. Bled*, Slovenia, University of Maribor, pp. 12–16.

Dobrišek, S., Mihelič, F., and Pavešić, N. (1994). Merging of time delayed feature vectors into extended vector in order to improve phoneme recognition. *Proceedings of the 4th COST 229 Workshop on Adaptive Methods and Emergent Techniques for Signal Processing and Communications*. Ljubljana, Slovenia, University of Ljubljana, pp. 145–150.

Gros, J., Pavešić, N., and Mihelič, F. (1997). Text-to-speech synthesis: A complete system for the Slovenian language. *Computing and Information Technology*, CIT-5, *1*:11–19.

Huang, X.D., Ariki, Y., and Jack, M.A. (1990). *Hidden Markov Models for Speech Recognition*. Edinburg Information Technology Series. Redwood Press Limited, London.

Ipšić, I., Mihelič, F., Dobrišek, S., Gros, J., and Pavešić, N. (1995). Overview of the spoken queries in European languages project: The Slovenian spoken dialog system. *Proceedings of the Scientific Conference on Artificial Intelligence in Industry*. High Tatras, Slovakia, pp. 431–438.

Jelinek, F. (1998). *Statistical Methods for Speech Recognition*. MIT Press. Cambridge, Massachusetts.

Levitt, H. (1995). Processing of speech signals for physical and sensory disabilities. *Speech Communication*, *9*:453–467.

Moulines, E. and Charpentier, F. (1990). Pitch-synchronous waveform processing techniques for text-to-speech synthesis using diphones. *Proceedings of the National Academy of Sciences of the USA*, pp. 9999–10006.

Skonnard, A. (1999). *Using the Windows Internet API with RAS, ISAPI, ASP, and COM*. Addison-Wesley, New York.

Šef, T., Dobnikar, A., and Gams, M. (2000). An agent speech module. *International Journal of Applied Mathematics*, *3*:267–280.

Zajicek, M., Powell, C., and Reeves, C. (1999). Ergonomic factors for a speaking computer interface. In M.A. Hanson, E.J. Lovesey, and S.A. Robertson (Eds.), Contemporary Ergonomics-Proceedings of the 50th Ergonomics Society Conference, Leicester University. Taylor and Francis, London, pp. 484–488.

Web References

Dillo—The Web Browser. (2002). http://dillo.sourceforge.net/

EIS—the Electronic Information System. (2002). http://www.zveza.slepih.si/zdsss/eis/

W3C—Web Access Initiative. (2002). http://www.w3.org/TR/WAI-WEBCONTENT/

W3C—Voice Browser Activity. (2002). http://www.w3.org/Voice/