Voice Information Retrieval for Course Resources

Yapin Zhong
Auburn University
Computer Science & Software Engineering
107 Dunstan Hall
Auburn, AL 36849 USA
zhongya@eng.auburn.edu

Juan E. Gilbert
Auburn University
Computer Science & Software Engineering
107 Dunstan Hall
Auburn, AL 36849 USA
gilbert@eng.auburn.edu

Weihong Hu
Auburn University
Computer Science & Software Engineering
107 Dunstan Hall
Auburn, AL 36849 USA
huweiho@eng.auburn.edu

Abstract: Advances in speech recognition technology provide students an alternate way to search for course resources via mobile devices such as cell phone instead of a computer. However, how to design voice information retrieval systems that access large course resources, still remains challenging (Coden & Brown 2002). This paper introduces a prototype of a voice information retrieval system for college course resources by using Voice Extensible Markup Language (VoiceXML).

Introduction

The traditional way for students to search for course resources is using a computer. The advances in speech recognition technology provide students an alternate way to search for course resources via mobile devices such as telephones instead of a computer. Designing voice information retrieval systems has its own difficulties, such as the design of the voice query interface and how to present a large result set to users aurally. As a result, the research topic, *Information Verbalization*, has emerged to allow voice access to large databases through traditional telephones, cellular phones, and other portable devices. The main difference in information verbalization and traditional voice information retrieval environments is that the previous allows the user to interact with a large database and a large set of retrieval results via the user's phone. Thus, when a user issues a query that returns a large set of results, the system would perform an intelligent summarization of those results and present the information to the user verbally.

VoiceXML (Voice Extensible Markup Language 2002) was used to implement our system. It is designed for creating audio dialogs that feature synthesized speech, digitized audio, recognition of spoken words/phrases and DTMF key input, recording of spoken input, telephony, and mixed-initiative conversations. One of its main benefits is to bring the advantages of Web-based development and content delivery to interactive voice applications.

Although there are several well-known projects involving voice information systems, they do not apply the voice interface to handling large information result spaces. For example, JUPITER (Zue et al. 2000) only provides a conversational interface to access and receive on-line limited weather forecast information. ELVIS (Walker et al. 1998) is an experimental spoken dialogue system that reads emails. FIMU (Cenek 2001) proposes a dialogue interface for library systems to allow users to access the library

catalogues. None of these systems allow the user to interact with large data results, such as college course resources which contain large documents. The following section will introduce a prototype of a voice information retrieval system for database course resources at Auburn University. The prototype has the benefit of giving students access to their course resources with a telephone anywhere and anytime. Also, the system provides an opportunity for students with visual and/or physical disabilities to access the resources easily.

System Design

The system consists of three main parts shown in Figure 1: User Interface, Voice Server, and Information Retrieval (IR) System. In the User Interface section, the user can start querying by speaking or using DTMF key input over the phone. After the system processes the query, the retrieval results are spoken to the user. The voice server is responsible for interpreting and controlling VoiceXML, such as speech recognition, speech synthesis and handling Internet requests. The IR system deals with the request from the voice server and sends responses back after interacting with the resource database. With the system architecture in Figure 1, the following section will describe how the system interacts with the user via voice.

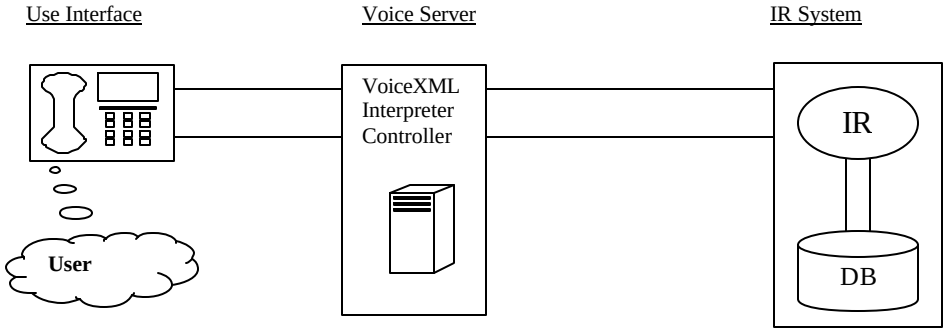

Figure 1: The system architecture

Voice User Interface

The voice user interface consists of four dialogues: Main Menu, Query, Result, and Save. Figure 2 is a voice dialogue diagram for the voice information retrieval system. After a registered user successfully logs into the system, the user reaches the Main Menu dialogue. The Main Menu dialogue allows the user to choose one of four search functions, “search for keyword”, “search for title”, “search for author”, or “search for year”. The “search for keyword” option can be used to retrieve all the documents that contain a specific word. If the user knows any word in the title of the document, the method “search for title” can be used to find all the possible titles of the documents that contain that word. The user can select “search for author” when she wants to retrieve all the documents written by a particular author. The final method “search for year” is used when the user wants to retrieve all the documents published in a particular year. The Query Dialogue allows the user to say the words that will be used during the search. The voice server then sends the query information to the IR system for processing, where the results are collected. The Result dialogue presents the list of retrieval documents to the user. The total number of documents found is spoken to the user followed by the spoken title of each document. The user can say “next”, “previous”, “stop” or “repeat” to navigate the result title list. The user can also say “detail” for any document to hear the document’s abstract. During reading the abstract, the system can respond to the user’s command. For example, when the user speaks “next”, the system goes to the next title, or “previous” for the previous title, or “stop” to exit the results list. If the user finds an interesting document which she wants to save for later use, she can simply speak “save” to go to the Saving dialogue. The Saving dialogue allows the user to request a copy of the article. The article can be sent via email or delivery to a local library. Before saving a document, the user must navigate the result list, which could be large. If a large set of the retrieved documents is returned, it is impossible for the user to listen to all the document titles one by one. For example, there maybe 1,000 documents returned, how does the user quickly identify a specific document? A document clustering approach attempts to solve this problem.

The system automatically compares the similarity of the documents and applies the Bisecting K-Means (Steinbach et al. 2000) to cluster the large result set to five distinct clusters. Documents within a cluster are similar with respect to topic. The user will be presented with a document from each cluster. The user can listen to more documents similar to the presented document by verbally selecting the presented document. Currently, this feature has not been integrated into the system completely. An example of interacting with the system is described as following.

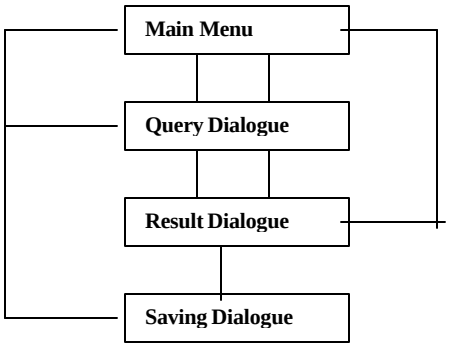

Figure 2: The voice user interface of the system

System : Welcome to Database I course reading references search main menu. You can say main menu or exit at any time. Please say one of the following : search for title, search for author, search for year, search for keyword…

User : search for keyword

System : Please say a keyword in the article…

User : xml

System : You are searching for 'xml'. You have found five results. Here are the result title list. You can say PREVIOUS, NEXT, DETAIL, STOP, REPEAT, SAVE or TRY AGAIN through the list. (time break)

Document one, Relevance is 87 percent. Structural Joins: A Primitive for Efficient XML Query Pattern Matching. (time break)

Document two, Relevance is 85 percent. TAX: A Tree Algebra for XML…

User : previous

System : Structural Joins: A Primitive for Efficient XML Query Pattern Matching…

User : detail

System : (audio clip) XML queries typically specify patterns of selection predicates on multiple elements that have some specified tree structured relationships…

User : save

System : Say email to save the article using your email address
Say library to save the article using your library.

User : email

System : At the tone, please say your email address.
When you're done, press pound.

User : sampletest@eng.auburn.edu (press pound)

System : Here's your email address : sampletest@eng.auburn.edu. If you are happy with this email address, say yes. If you would like to re-record your email address say no.

User : yes

System : saving your email address. What would you like to do next? Main menu, try again or exit?

User : exit

System : You are disconnecting database I course reading reference. Bye bye! Have a nice day!

Conclusion

This paper introduces a prototype of a voice information retrieval system for college course resources. Furthermore, it emphasizes how the system interacts with the user through a telephone. The system gives students an alternative approach to access Web course resources anywhere and anytime. Also, the system provides an opportunity for people with disabilities to access Web resources.

The important features and usability of the proposed system will be clearly demonstrated during the presentation. The presentation will include a demonstration of how the system retrieves the course resources with users using cellular phones. The architecture and implementation will be discussed during the presentation as well.

References

Cenek, P. Dialogue interface for library System. (2001), [On line]. http://www.fi.muni.cz/informatics/reports/files/2001/FIMU-RS-2001-04.pdf.

Coden, A. & Brown, E. W. (2002), Information Retrieval Techniques for Speech Applications. SIGIR FORUM, Volume 36 Number 1, Sprint 2002.

Steinbach, M., Karypis, G., and Kumar V. (2000), A Comparison of Document Clustering Techniques. In KDD Workshop on Text Mining.

Voice Extensible Markup Language (VoiceXML) Version 2.0. (2002), [On line]. http://www.w3.org/TR/voicexml20/.

Walker M.A., Fromer, J.C., & Narayanan, S. (1998), Learning optimal dialogue strategies: A case study of a spoken dialogue agent for email. In *Proceedings of ACL/COLING 98*.

Zue, V. et al. (2000), JUPITER: A telephone-based conversational interface for weather information, IEEE Trans. on Speech and Audio Processing, vol. 8, no. 1.