Speech User Interfaces for Information Retrieval

Juan E. Gilbert
Computer Science & Software Engineering
107 Dunstan Hall
Auburn University, AL 36849
(334) 844-6316
gilbert@eng.auburn.edu

Yapin Zhong
Computer Science & Software Engineering
107 Dunstan Hall
Auburn University, AL 36849
(334) 844-3653
zhongya@eng.auburn.edu

ABSTRACT

The research proposed here concentrates on the problem of designing and developing a spoken query retrieval (SQR) system to access large document databases via voice. The main challenge is to identify and address issues related to designing an effective and efficient speech user interface (SUI), especially if the aim is to facilitate spoken queries of large document databases. Furthermore, the task of presenting large query result sets aurally should be performed such that the user's short term memory is not overloaded. In this paper, a framework allowing information retrieval to large document databases via voice is presented and findings from a research study using the framework will be discussed as well.

Categories and Subject Descriptors

H.5.2 [**Information Interfaces and Presentation**]: User Interfaces – *Voice I/O, Natural Language.*

General Terms: Human Factors, Experimentation

Keywords: Speech User Interfaces, Information Retrieval, VoiceXML, Spoken Query Retrieval

1. INTRODUCTION

The most common approach for interacting with the Web is using a computer connected to the Internet. This type of interaction means that the Web is almost entirely a visual medium. However, there are many cases when this approach is either impossible or inappropriate.

First, a very large part of the world population does not have access to either computers or the Internet. eTForecasts [5] reports there are only 666 million Internet users out of the over 5 billion population of the world. Meanwhile, the number of cellular phones sold world wide doubled from 200 million in 1999 to 400 million units in 2000 and is expected to reach 2 billion this year. In addition to these mobile users, the Washington-based telecom research firm TeleGeography reported that there were 969 million fixed telephone lines in 2000, although this number has remained relatively constant in recent years. These numbers reveal that there are far more cellular and telephone users than Internet computer users worldwide.

CIKM'03, November 3–8, 2003, New Orleans, Louisiana, USA.

Second, advances in technology are producing small, pervasive devices with tiny interfaces that allow people to interact with the Web (which come in the form of handheld computers, also known as personal digital assistants (PDA), cellular phones and, most recently, PDAs with cellular capabilities), these interfaces are difficult to use because of their limited viewing area, a severe handicap when it comes to the point of retrieving large amounts of information from the Web. Furthermore, interactions occur by using a keypad that has buttons smaller than M&Ms or by using a special stylus on a small touch screen, as seen for most PDAs. All of these features are inconvenient and tend to make users feel quite uncomfortable. These examples, coupled with the unprecedented explosion in the number of cellular phones, have created a need for an alternative approach to information retrieval. One emerging trend is information retrieval via voice.

With the recent progress in Automatic Speech Recognition (ASR), a number of speech-based methods have been explored in Information Retrieval (IR). According to Fujii and his colleagues [6], these methods can be classified into two fundamental categories: spoken document retrieval (SDR) and spoken query retrieval (SQR). In SDR, written queries are used to search speech archives for relevant speech information, while SQR uses spoken queries to retrieve relevant textual information. A large amount of the research in SDR has been promoted by the Spoken Document Retrieval track of TREC [7]. In contrast, very little work has been carried out in the area of SQR [1, 2]. For example, earlier research from Barnett et al. [1] showed that longer queries are more robust in terms of tolerating errors than shorter queries. More recently, Fujii and his colleagues [6] showed that using a language model generated from the target collection can significantly improve both the recognition and retrieval accuracy. However, these studies focused solely on investigating the effects of speech recognition accuracy on IR methods based on non-spontaneous and long queries and did not take into account the major properties of IR during the searching process, such as the effects of different query interfaces on the performance of IR systems. In fact, an SQR system is more complicated. It is integrates ASR and IR, but is not "simply connected by way of an input/output protocol" [6]. In this research, a three component architecture is proposed to investigate SQR systems.

2. SYSTEM ARCHITECTURE

The system consists of three components, as shown in Figure 1: Speech Interface, the Voice Portal, and the Backend Server. In the speech interface component, the user can initiate the query by speaking from a telephone, cellular phone, computer or some other handheld device. The telecommunications network may use Voice Over IP (VoIP), a Private Branch Exchange (PBX) switch or a

Public Switched Telephone Network (PSTN). In any case, the connection will be passed on to the voice portal.

In the voice portal, there are three sub components: the Voice Server, Language Model (LM), and Voice Navigator. The voice server contains telephony hardware that allows the server to answer the call. Also, the voice server has a VoiceXML interpreter and controller. It is responsible for automatic speech recognition (ASR), speech synthesis and handling Internet requests. The voice server is also called a voice browser. Since it will perform only lower language processing functions, here the term "voice server" is preferred over "voice browser". The language model (LM) provides a dynamic language vocabulary for ASR. The voice navigator is responsible for providing effective dialogue functions for document querying and browsing. The user's request is sent to the backend server via the voice server.

The Backend Server consists of an IR module, domain specific document database, and the language model agent (LMA). The IR module handles requests from the voice server. The LMA dynamically generates the language vocabulary for the LM according to the context of the user's request.

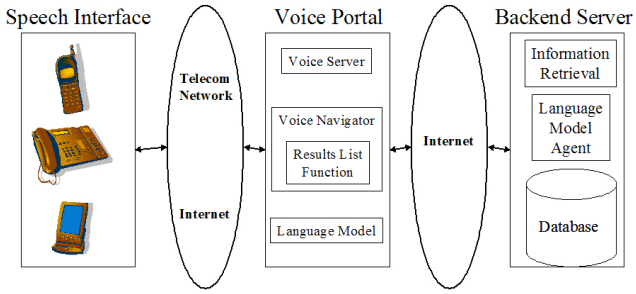

Figure 1: SQR Architecture

The system works in the following way. A user connects to the system using a telephone. After the user has successfully logged in by means of a user id, he or she can initiate a spoken query related to the domain of the document database. The voice navigator interacts with the user to identify the exact content of the spoken query. The query is then translated into text and sent to the IR module. The IR module matches the query terms against the indexes in the document database and produces a ranked list of documents. Documents in the ranked list are passed to the user via the voice navigator. Based on different result representation strategies, the user can listen to the title of each document in ranked sequence, ask to skip a document, save a document to the user's personal folder, or stop the process. Finally, the user can ask the system to deliver all documents of interest to his or her email box via the voice navigator or to a local library for printing. The following section will explain in detail how an effective speech user interface supports these functions.

3. SPEECH USER INTERFACE

When implementing a SQR system, there are three major problems that occur due to the unique nature of this environment. First, the speech user interface must provide the user with a usable query interface that doesn't overload the user's memory. Unlike visual interfaces, SUIs lack persistence. When the machine says something, it is immediately lost if the user doesn't hold that information in short term memory. Therefore, the design of the SUI must take into account the user's short term memory in accepting queries and processing query results. This problem is addressed using a number of SUI design principles.

Second, all SUIs have a language or grammar model that is used to recognize spoken words and phrases. These models often suffer from word recognition errors due to words that sound alike or words that are missing from the model. Furthermore, the language model for a SQR system can become very large, very quickly given all of the possible terms that are maintained in a document database. As the size of these models grows, they can exceed the memory capabilities of the voice server. In this research a bisecting k-means clustering [16] approach is proposed to address the language model issue by creating language model or grammar clusters. Grammar clusters consist of words that are found in similar documents, yet the words don't sound alike.

Third, even when a query is successfully processed, the results list of the query may be significantly large. For a SUI a large results list is typically greater than 10. Miller [11] noted that humans can handle 7 plus or minus 2 items in short term memory. Therefore, SQR systems have to overcome the problem of presenting results lists to the user that overload the user's short term memory. In order to address this problem, this research proposes a number of approaches that are referred to as information verbalization techniques. Information verbalization is the use of computer supported, auditory interactions to amplify understanding of abstract and/or large data. One technique that is being explored is the use of bisecting k-means to cluster like documents. Once the like documents have been clustered, the title of the top ranked document from each cluster is returned to the user. This reduces the size of the results list to a predefined number of clusters.

3.1 Query and Result Interface Strategies

Query and result interface designs for SQR are extensively affected by IR methods. Providing an effective approach for users to formulate the search terms is an important step. The four-phase framework for search in a graphical user interface (GUI) by Shneiderman et al. [13] suggests that the formulation is the most complex phase in that it involves decisions of several types. These decisions include the sources of the search, which fields of documents to search, what actual text to search for; and what variants of that text to accept. These problems exist in SQR systems as well with the added complexity of knowing ahead of time, what the user will say as a search phrase. Recall, that SQR has a language model that represents all of the spoken phrases that users will say. In a GUI, the user simply types a search phrase into a text field. In order to facilitate effective spoken query retrieval, a bisecting k-means clustering [16] approach was implemented.

First, a domain specific document collection is selected, i.e. TREC-9 filtering collection [12], Figure 2. Next, the bisecting k-means clustering algorithm is applied to the collection with a goal of creating five document clusters, Figure 3. The number five was selected because of the 7 plus or minus 2 short term memory limitation described by Miller [11]. In Figure 3, there are five document clusters. Each cluster has a centroid document represented by a red circle in the middle of the cluster. The terms, minus the stop words, from each centroid document are used to create the language model for the SQR interface. From experimentation, with the TREC-9 collection and creating five document clusters, it has been found that the coverage of all terms, minus the stop words, for each

cluster by using the terms from the centroid document are 93%, 96%, 98%, 94% and 93%. In other words, the terms from the centroid documents cover at least 93% of all the terms within their respective clusters. Once the language model was defined using the terms from the centroids, the SQR interface design was complete. The next challenge is the representation of large results lists.

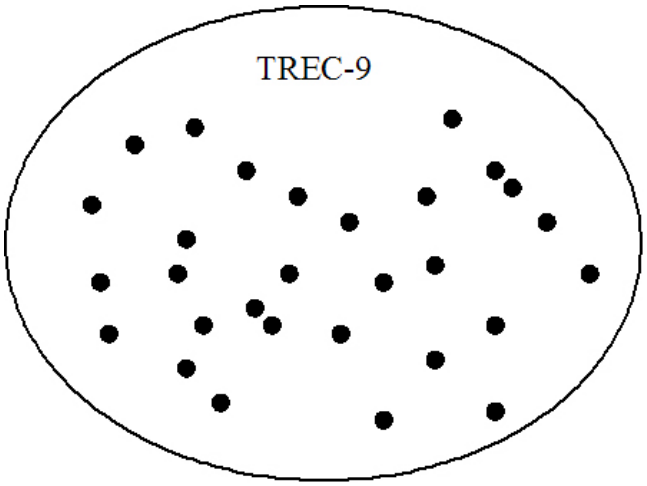

Figure 2: TREC-9 Documents

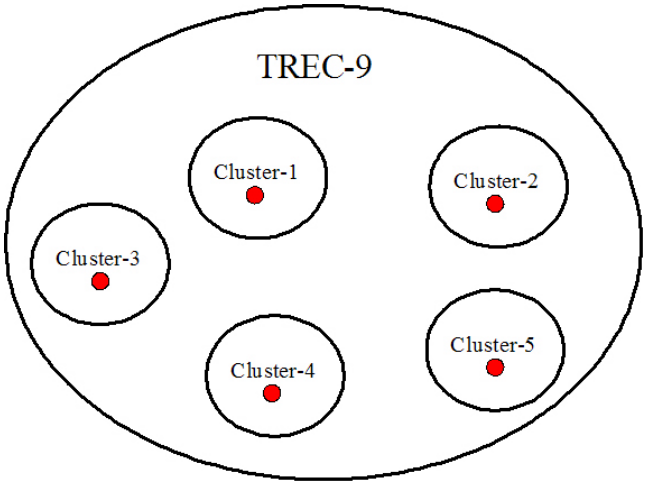

Figure 3: TREC-9 Document Clusters

When a spoken query is submitted to the backend server, a results list is generated. In a GUI environment, the results are presented to the user in 10 or 25 document titles at a time with an option to view the next 10 or 25. As described earlier, this model does not work in a speech only user interface. Therefore, an adjustment had to be made to restrict the number of documents that appear in the results list. Bisecting k-means clustering [16] was used to accommodate this requirement. The backend server uses a standard vector space IR model [15]. The results are ranked [4] within each cluster. In the experiment, the top 5 ranked document titles where presented to the user and all the other documents were ignored. However, current research is being done on presenting the titles from the top ranked documents from each cluster to the user. This approach will reduce the results list significantly and retrieves the most relevant documents from each cluster.

4. EXPERIMENTS

After completing the system implementation based on the proposed approaches, a formal experiment was conducted for the SQR system.

.I 1

.U

87049087

.S

Am J Emerg Med 8703; 4(6):491-5

.M

Allied Health Personnel/; Electric Countershock/*; Emergencies; Emergency Medical Technicians/*; Human; Prognosis; Recurrence; Support, U.S. Gov't, P.H.S.; Time Factors; Transportation of Patients; Ventricular Fibrillation/*TH.*

.T

Refibrillation managed by EMT-Ds: incidence and outcome without paramedic back-up.

.P

JOURNAL ARTICLE.

.W

Some patients converted from ventricular fibrillation to organized rhythms by defibrillation-trained ambulance technicians (EMT-Ds) will refibrillate before hospital arrival. The authors analyzed 271 cases of ventricular fibrillation managed by EMT-Ds working without paramedic back-up. Of 111 patients initially converted to organized rhythms, 19 (17%) refibrillated, 11 (58%) of whom were reconverted to perfusing rhythms, including nine of 11 (82%) who had spontaneous pulses prior to refibrillation. Among patients initially converted to organized rhythms, hospital admission rates were lower for patients who refibrillated than for patients who did not (53% versus 76%, P = NS), although discharge rates were virtually identical (37% and 35%, respectively). Scene-to-hospital transport times were not predictively associated with either the frequency of refibrillation or patient outcome. Defibrillation-trained EMTs can effectively manage refibrillation with additional shocks and are not at a significant disadvantage when paramedic back-up is not available.

.A

Stults KR; Brown DD.

Figure 4: TREC-9 Sample Document

The objectives of this evaluation are an extension of a previous study [18] and are focused mainly on measuring user satisfaction and investigating how the spoken query user behaviors are different from GUIs. The PARADISE framework [10] was used to evaluate user satisfaction.

4.1 Document Databases

The document collection for this experiment was based on the TREC-9 filtering collection [12] which is a slightly modified version of the OHSUMED test collection available from Hersh et al. [8].

The TREC-9 filtering collection consists of Medline documents from the years 1987-1991 and a set of topics and relevance judgments. The entire collection contains 348,566 documents. The available fields are title, abstract, MeSH, author, source, and publication type. Figure 4 shows a sample of OHSUMED. The field MeSH contains human assigned index terms. These are assumed to arrive in identifier order, at a rate of approximately 6000 documents per month. The 1987 data was originally extracted from the dataset to provide training material; the test set is therefore the 1988-1991 data.

4.2 IR Engines

For this experiment, a statistical information retrieval engine was built based on the standard vector space IR model [15]. In future research the extended boolean model will be explored [14]. In the vector space IR model, the documents and queries are represented as vectors where each component in the vector is an indexing term. Each term has an associated weight based on the term's occurrence statistics both within and across documents; the weight reflects the relative discrimination capability of that term. The weight of the term within a document is called the term frequency (*tf*). The weight of the term cross documents is called the inverse document frequency (*idf*). Although *idf* is a more accurate quantification of the term frequency, there must always be two steps used to index the documents when the document database is updated. The first step is to index the documents and to compute the collection statistics, and the second step is to adjust the document term weights to include the IDF factor. In order to be computationally efficient, the IDF factor is included in the query terms but not the document terms.

A similarity measure between document and query vectors is computed and is used to score and rank the documents in order to perform retrievals. Here, Cosine functions were used to measure the similarity between the document and query vectors.

4.3 Participants and Procedures

Approximately 26 college level students were recruited as subjects. Since the system is designed to search for information in large document databases, all potential subjects should have some basic experience in searching for information a GUI. All of the subjects were randomly chosen from Auburn University students. To ensure all potential subjects have a similar level of knowledge of using the Web to search for documents, all subjects were enrolled in at least one of three different computer science courses at Auburn University.

The usability evaluation was a controlled experiment. To reduce the differences caused by other factors, the following controls were applied:

- All participants used the same telephone sitting in the same chair in the same room with the researcher.
- The tasks that were completed by the participants were the same. Furthermore, all participants were asked to do the same task in the same order.
- The delay time for each participant to start the interview and survey was the same. They all started their interview and survey right away after they completed the search task.
- All participants were told not to talk to their classmates about the experiment to make sure that all participants have an equal knowledge of the experiment.

Each subject was given the same instructions for the first two minutes of the experiment. After instructions were given, the participants read a sample document from the collection, Figure 5, and were asked to do research on the topic covered by the sample document using the SQR system. The participants called the SQR system using the BeVocal Cafe [2] and performed their searches. At the end of the call, each user filled out a survey giving a subjective evaluation of the system's performance.

You are taking an introductory level health course. You have been given the assignment of writing a term paper. In order to do your term paper, you were given the abstract below. Please read the abstract and use the SQR system to obtain more documents related to this abstract.

More than 50 percent of the chronically mentally ill receive their medical, psychiatric, and social support services from primary care physicians in the general health sector. Despite this high level of involvement with these patients, the majority of family physicians consider their training in the management of patients with mental disorders to be inadequate. This paper describes six categories of critical competencies that should be included in the mental health curricula of family physician training programs: therapeutic attitudes and skills, diagnosis and differential diagnosis, functional assessment, psychopharmacology, management of emergencies, and psychosocial treatments. It outlines the manner in which specific competencies could be incorporated in medical school, in family practice residency training, and in postgraduate continuing medical education as well as the specific elements included in each. The discussion is based on the assumption that more effective participation by family physicians in the treatment of chronic psychiatric illness requires active attention throughout the continuum of medical education.

Figure 5: An example of a task scenario

4.4 Data Collection

Once the experiment was completed, the values for a range of evaluation measures were extracted from the resulting data, including system logs, dialogue recordings, and surveys. The following is subset of the data collected.

- User satisfaction via survey.
- Task Success via Kappa statistic [10].
- Interface Quality, based on ASR Rejections, Timeouts, Help Requests, Mean Recognition, and Barge Ins.
- Interface Efficiency, based on System turns, User turns, and Elapsed Time.
- Speech recognition accuracy via recognition errors.
- User spoken query behaviors via spoken query terms

Each question from the survey was designed to measure a particular factor, such as the quality of the text-to-speech engine, the performance of the ASR engine, task ease, system response,

interaction pace, and so on. User responses were measured on a nine point likert scale.

To measure task success, the task scenario key and scenario execution attribute value matrices were compared by using the Kappa coefficient as described in the PARADISE framework [10].

To measure interface quality, the number of prompts per test for Help Requests, ASR Rejections, and Timeouts were extracted from the system logs. Also, a manual analysis was conducted to determine how many times the user interrupted the system (Barge-in).

To measure dialogue efficiency, the number of System Turns and User Turns were extracted from the system log, and the total Elapsed Time was determined from the recording.

After the results are obtained from the task success metrics, interface efficiency metrics, and interface qualitative metrics, the user satisfaction and other system performance measurements were calculated.

4.5 Results and Discussion

The average user satisfaction rating was 80% based on 31 satisfactory factors from the survey. One of the most significant findings with respect to user satisfaction was the user's perception of their task completion. This survey question resulted in an 83% satisfaction rating. In other words, the users felt that they completed the task with an average success of 83%. The average user satisfaction rating (80%) and the average self satisfaction rating (83%) indicate that the SQR system was relatively easy to use.

A language model was created using the terms from the centroid documents as explained in section 3.1. The language model consisted of 389 terms; again this represents at least a 93% coverage of all the terms found in all the documents. This experiment found that there were a mean of 2.43 terms per spoken query with a range of 1 to 8. From Jansen and his colleague's study [9], the mean of Web search terms is 2.21. There appears to be no significant difference between the lengths of search terms for spoken queries and typed queries. Table 1 contains additional statistics related to the spoken queries and search terms.

Table 1: Numbers of users, queries, and terms

Total Number of Participants	26
Total Number of Spoken Queries	135
Average Number of Spoken Queries per Users	5.19
Number of Unique Queries	95
Total Number of Spoken Terms	328
Total Number of Uniquely Spoken Terms	70
Mean Number of Terms	2.43

There were 135 spoken queries submitted by 26 participants with an average of 5.19 queries per participant. One participant submitted 13 queries, which was the maximum for any single participant, and others submitted 1 query, which was a minimum. There were 95 unique spoken queries covering 328 terms, where the language model contained 389 terms. There were 6 unique terms that were spoken by participants that did not appear in the language model. Those 6 terms were "of", "in", "med", "for", "discover", and "resolve". Three of those terms are stop words, "of", "in" and "for", which were removed from the language model. Therefore, the experimental coverage of the language model was 96%, excluding the stop words. Additionally, of the 135 spoken queries, 13 were unrecognized by the ASR engine. In other words, the participants spoke words that were in the language model, but the ASR engine misunderstood the participants' utterances. This yields a recognition rate of 90.3%. In most of the 13 unrecognized utterances, the participant's speech was too low. Therefore, the participant had to speak up and the ASR engine recognized the participant's query. In all the cases where a misinterpretation occurred, the participant was able to recover within 3 attempts.

5. CONCLUSION

The aim of this research study was to investigate SQR systems using VoiceXML as the voice interface and bisecting k-means to create the language model. This research study investigated user satisfaction and language model coverage for a SQR system. The findings from this experiment proved to be very promising with an overall user satisfaction rating of 80% and a language model that covered at least 93% of the terms found in the document database. Furthermore, the SQR system experienced 96% coverage through experimentation, excluding the 3 stop words that participants used. Also, the SQR system performed at a successful recognition rate of 90.3%. In the cases where there was an ASR error, all of the errors were corrected within 3 attempts. These findings are very promising for SQR system research. This study is an initial study leading to additional research on SQR systems. Future research will include larger domain specific document repositories and digital libraries that yield larger language models. The domain specific requirement is suggested in order to optimize the ASR language model performance by reducing the language model size. Furthermore, domain specific repositories will yield more similar questions/queries, which will enhance the ASR language model performance as well. More extensive research using more participants from various backgrounds is required as well. In future research, other clustering algorithms will be investigated for use during the results list presentation.

6. REFERENCES

[1] Barnett, J., Anderson, S., Broglio, J., Singh, M., Hudson, R., and Kuo, S. Experiments in spoken queries for document retrieval. In Proceedings of Eurospeech97, pages 1323-1326, 1997.

[2] BeVocal Cafe. Available at http://cafe.bevocal.com, 2003.

[3] Crestani, F. Spoken Query Processing for Interactive Information Retrieval. Data and Knowledge Engineering, 41(1):105-124, 2002.

[4] Drori, O., Improving Display of Search Results in Information Retrieval Systems - User's Study. Technical Report of the Leibnitz Center for Research in Computer Science, No. 200034, 2000.

[5] eTForecasts.com. [Online] Available: http://www.etforecasts.com/products/ES_pcww.htm#list. 2002.

[6] Fujii, A., Itou, K., and Ishikawa, T. Speech-Driven Text Retrieval: Using Target IR Collections for Statistical Language Model Adaptation in Speech Recognition, Proc. of the

SIGIR'01 Workshop on Information Retrieval Techniques for Speech Applications, Springer LNCS 2273, 2002.

[7] Garofolo, J., Auzanne, C., and Voorhees, E. The TREC spoken document retrieval track: A success story. In Proceedings of TREC-8 (1999), NIST special publication, 2000.

[8] Hersh, W., Buckley, C., Leone, T., and Hickam, D. OHSUMED: An Interactive Retrieval Evaluation and new large test collection for research. In Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pages 192-201, 1994.

[9] Jansen, B. J., Spink, A., Bateman, J., & Saracevic, T. Real life information retrieval: a study of user queries on the web. SIGIR Forum, Vol. 32. No. 1, pp. 5-17, 1998.

[10] Kamm, C.A. & Walker, M.A. Design and evaluation of spoken dialog systems. In Proceedings of the ASRU Workshop, 1997.

[11] Miller, G.A. The magical number seven, plus or minus two: Some limits on our capacity for processing information. Psychological Science, 63, pp. 81-97, 1956.

[12] Robertson, S. & Hull D. The TREC-9 Filtering Track Final Report. [online] Available: http://trec.nist.gov/pubs/trec9/papers/filtering_new.pdf, 2000.

[13] Shneiderman, B. The Future of the Web: Visual, Social, Universal. [Online] Available: http://www.cs.umd.edu/hcil/pubs/presentations/FutureWeb/. 2000.

[14] Salton, G., Fox, E.A. and Wu, H. Extended Boolean Information Retrieval. Communications of the ACM, 26(11):1022-1036, 1983.

[15] Salton, G. & McGill, M. Introduction to Modern Information Retrieval. McGraw-Hill Book Co., New York, 1983.

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

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

[18] Zhong, Y., Gilbert, J., & Hu, W. Speech User Interface for Document Retrieval. In *Proceedings of the 41st Annual ACM Southeast Conference*, Savannah, Georgia, March 7-8, 2003.