A speech interface for open-domain question-answering

Edward Schofield
ftw. Telecommunications Research Center
Vienna, Austria
Department of Computing
Imperial College London, U.K.
schofield@ftw.at

Zhiping Zheng
Dept. of Computational Linguistics
Saarland University
Saarbrücken, Germany
zheng@coli.uni-sb.de

Abstract

Speech interfaces to question-answering systems offer significant potential for finding information with phones and mobile networked devices. We describe a demonstration of spoken question answering using a commercial dictation engine whose language models we have customized to questions, a Web-based text-prediction interface allowing quick correction of errors, and an open-domain question-answering system, AnswerBus, which is freely available on the Web. We describe a small evaluation of the effect of recognition errors on the precision of the answers returned and make some concrete recommendations for modifying a question-answering system for improving robustness to spoken input.

1 Introduction

This paper demonstrates a multimodal interface for asking questions and retrieving a set of likely answers. Such an interface is particularly appropriate for mobile networked devices with screens that are too small to display general Web pages and documents. Palm and Pocket PC devices, whose screens commonly display 10–15 lines, are candidates. Schofield and Kubin (2002) argue that for such devices question-answering is more appropriate than traditional document retrieval. But until recently no method has existed for inputting questions in a reasonable amount of time. The study of Schofield (2003) concludes that questions tend to have a limited lexical structure that can be exploited for accurate speech recognition or text prediction. In this demonstration we test whether this result can endow a real spoken question answering system with acceptable precision.

2 Related research

Kupiec and others (1994) at Xerox labs built one of the earliest spoken information retrieval systems, with a speaker-dependent isolated-word speech recognizer and an electronic encyclopedia. One reason they reported for the success of their system was their use of simple language models to exploit the observation that pairs of words co-occurring in a document source are likely to be spoken together as keywords in a query. Later research at CMU built upon similar intuition by deriving the language-model of their Sphinx-II speech recognizer from the searched document source. Colineau and others (1999) developed a system as a part of the THISL project for retrieval from broadcast news to respond to news-related queries such as *What do you have on . . . ?* and *I am doing a report on . . . — can you help me?* The queries the authors addressed had a simple structure, and they successfully modelled them in two parts: a question-frame, for which they hand-wrote grammar rules; and a content-bearing string of keywords, for which they fitted standard lexical language-models from the news collection.

Extensive research (Garofolo et al., 2000; Allan, 2001) has concluded that spoken documents can be effectively indexed and searched with word-error rates as high as 30–40%. One might expect a much

higher sensitivity to recognition errors with a short query or natural-language question. Two studies (et al., 1997; Crestani, 2002) have measured the detrimental effect of speech recognition errors on the precision of document retrieval and found that this task can be somewhat robust to 25% word-error rates for queries of 2–8 words.

Two recent systems are worthy of special mention. First, Google Labs deployed a speaker-independent system in late 2001 as a demo of a telephone-interface to its popular search engine. (It is still live as of April 2003.) Second, Chang and others (2002a; 2002b) have implemented systems for the Pocket PC that interpret queries spoken in English or Chinese. This last group appears to be at the forefront of current research in spoken interfaces for document retrieval.

None of the above are question-answering systems; they boil utterances down to strings of keywords, discarding any other information, and return only lists of matching documents. To our knowledge automatic answering of spoken natural-language questions has not previously been attempted.

3 System overview

Our demonstration system has three components: a commercial speaker-dependent dictation system, a predictive interface for typing or correcting natural-language questions, and a Web-based open-domain question-answering engine. We describe these in turn.

3.1 Speech recognizer

The dictation system is Dragon NaturallySpeaking 6.1, whose language models we have customized to a large corpus of questions. We performed tests with a head-mounted microphone in a relatively quiet acoustic environment. (The Dragon Audio Setup Wizard identified the signal-to-noise ratio as 22 dBs.) We tested a male native speaker of English and a female non-native speaker, requesting each first to train the acoustic models with 5–10 minutes of software-prompted dictation.

We also trained the language models by presenting the Vocabulary Wizard the corpus of 280,000 questions described in (Schofield, 2003), of which Table 1 contains a random sample. The primary function of this training feature in NaturallySpeaking is to add new words to the lexicon; the nature of the other adaptations is not clearly documented. New 2-grams and 3-grams also appear to be identified, which one would expect to reduce the word-error rate by increasing the 'hit rate' over the 30–50% of 3-grams in a new text for which a language model typically has explicit frequency estimates.

3.2 Predictive typing interface

We have designed a predictive typing interface whose purpose is to save keystrokes and time in editing misrecognitions. Such an interface is particularly applicable in a mobile context, in which text entry is slow and circumstances may prohibit speech altogether.

We fitted a 3-gram language model to the same corpus as above using the CMU–Cambridge SLM Toolkit (Clarkson and Rosenfeld, 1997). The interface in our demo is a thin JavaScript client accessible from a Web browser that intercepts each keystroke and performs a CGI request for an updated list of predictions. The predictions themselves appear as hyperlinks that modify the question when clicked. Figure 1 shows a screen-shot.

3.3 Question-answering system

The AnswerBus system (Zheng, 2002) has been running on the Web since November 2001. It serves thousands of users every day. The original engine was not designed for a spoken interface, and we have recently made modifications in two respects. We describe these in turn. Later we propose other modifications that we believe would increase robustness to a speech interface.

Speed

The original engine took several seconds to answer each question, which may be too slow in a spoken interface or on a mobile device after factoring in the additional computational overhead of decoding the speech and the longer latency in mobile data networks. We have now implemented a multi-level caching system to increase speed.

Our cache system currently contains two levels. The first is a cache of recently asked questions. If a question has been asked within a certain period of time the system will fetch the answers directly

Table 1: A random sample of questions from the corpus.

How many people take ibuprofen
What are some work rules
Does GE sell auto insurance
The roxana video diaz
What is the shortest day of the year
Where Can I find Frog T-Shirts
Where can I find cheats for Soul Reaver for the PC
How can I plug my electric blanket in to my car cigarette lighter
How can I take home videos and put them on my computer
What are squamous epithelial cells

from the cache. The second level is a cache of semi-structured Web documents. If a Web document is in the cache and has not expired the system will use it instead of connecting to the remote site. By 'semi-structured' we mean that we cache semi-parsed sentences rather than the original HTML document. We will discuss some technical issues, like how and how often to update the cache and how to use hash tables for fast access, in another paper.

Output

The original engine provided a list of sentences as hyperlinks to the source documents. This is convenient for Web users but should be transformed for spoken output. It now offers plain text as an alternative to HTML for output. [1]

We have also made some cosmetic modifications for small-screen devices like shrinking the large logo.

4 Evaluation

We evaluated the accuracy of the system subject to spoken input using 200 test questions from the TREC 2002 QA track (Voorhees, 2002). AnswerBus returns snippets from Web pages containing possible answers; we compared these with the reference answers used in the TREC competition, overriding about 5 negative judgments when we felt the answers were satisfactory but absent from the TREC scorecard. For each of these 200 questions we passed two strings to the AnswerBus engine, one typed verbatim, the other transcribed from the speech of one of the people described above. The results are in Tables 2 and 3.

[1] See http://www.answerbus.com/voice/

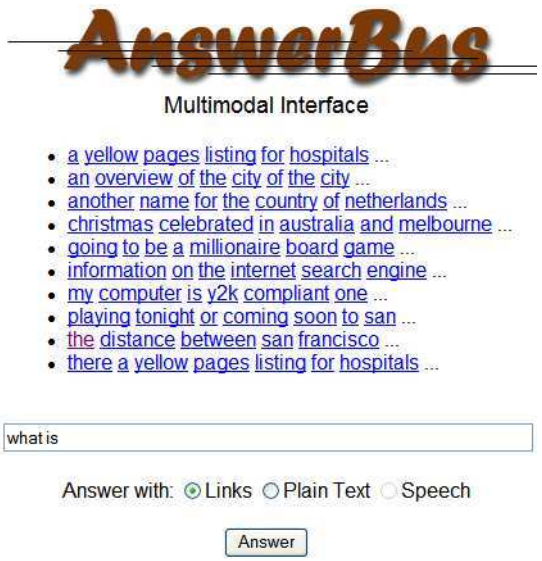

Figure 1: The interface for rapidly typing questions and correcting mistranscriptions from speech. Available at speech.ftw.at/~ejs/answerbus

Table 2: % of questions answered correctly from perfect text versus misrecognized speech.

	Speaker 1	Speaker 2
Misrecognized speech	39%	26%
Verbatim typing	58%	60%

5 Discussion

We currently perform no automatic checking or correction of spelling and no morphological stemming

Table 3: # of answers degraded or improved by the dodgy input.

	Speaker 1	Speaker 2
Degraded	12	34
Improved	5	0

of words in the questions. Table 3 indicates that these features would improve robustness to errors in speech recognition. We now make some specific points regarding homographs, which are typically troublesome for speech recognizers. QA systems could relatively easily compensate for confusion in two common classes of homograph:

- plural nouns ending *–s* versus possessive nouns ending *–'s* or *–s'*. Our system answered Q39 *Where is Devil's tower?*, but not the transcribed question *Where is Devils tower?*
- written numbers versus numerals. Our system could not answer *What is slang for a 5 dollar bill?* although it could answer Q92 *What is slang for a five dollar bill?*.

More extensive 'query expansion' using synonyms or other orthographic forms would be trickier to implement but could also improve recall. For example, Q245 *What city in Australia has rain forests?* it answered correctly, but the transcription *What city in Australia has rainforests* (without a space), got no answers. Another example: Q35 *Who won the Nobel Peace Prize in 1992?* got no answers, whereas *Who was the winner . . . ?* would have found the right answer.

6 Conclusion

This paper has described a multimodal interface to a question-answering system designed for rapid input of questions and correction of speech recognition errors. The interface for this demo is Web-based, but should scale to mobile devices. We described a small evaluation of the system's accuracy given raw (uncorrected) transcribed questions from two speakers, which indicates that speech can be used for automatic question-answering, but that an interface for correcting misrecognitions is probably necessary for acceptable accuracy.

In the future we will continue tightening the integration of the components of the system and port the interface to phones and Palm or Pocket PC devices.

Acknowledgements

The authors would like to thank Stefan Rüger for his suggestions and moral support. Ed Schofield's research is supported by a Marie Curie Fellowship of the European Commission.

References

J. Allan. 2001. Perspectives on information retrieval and speech. *Lect. Notes in Comp. Sci.*, 2273:1.

E. Chang, Helen Meng, Yuk-chi Li, and Tien-ying Fung. 2002a. Efficient web search on mobile devices with multi-modal input and intelligent text summarization. In *The 11th Int. WWW Conference*, May.

E. Chang, F. Seide, H.M. Meng, Z. Chen, S. Yu, and Y.C. Li. 2002b. A system for spoken query information retrieval on mobile devices. *IEEE Trans. Speech and Audio Processing*, 10(8):531–541, nov.

P. R. Clarkson and R. Rosenfeld. 1997. Statistical language modeling using the CMU–Cambridge toolkit. In *Proc. ESCA Eurospeech 1997*.

N. Colineau and A. Halber. 1999. A hybrid approach to spoken query processing in document retrieval system. In *Proc. ESCA Workshop on Accessing Information In Spoken Audio*, pages 31–36.

F. Crestani. 2002. Spoken query processing for interactive information retrieval. *Data & Knowledge Engineering*, 41(1):105–124, apr.

J. Barnett et al. 1997. Experiments in spoken queries for document retrieval. In *Proc. Eurospeech '97*, pages 1323–1326, Rhodes, Greece.

J. S. Garofolo, C. G. P. Auzanne, and E. M. Voorhees. 2000. The TREC spoken document retrieval track: A success story. In *Proc. Content-Based Multimedia Information Access Conf.*, apr.

J. Kupiec, D. Kimber, and V. Balasubramanian. 1994. Speech-based retrieval using semantic co-occurrence filtering. In *Proc. ARPA Human Lang. Tech. Workshop*, Plainsboro, NJ, mar.

E. Schofield and G. Kubin. 2002. On interfaces for mobile information retrieval. In *Proc. 4th Int. Symp. Human Computer Interaction with Mobile Devices*, pages 383–387, sep.

E. Schofield. 2003. Language models for questions. In *Proc. EACL Workshop on Language Modeling for Text Entry Methods*, apr.

E.M. Voorhees. 2002. Overview of the trec 2002 question answering track. In *The 11th Text Retrieval Conf. (TREC 2002)*. NIST Special Publication: SP 500-251.

Z. Zheng. 2002. AnswerBus question answering system. In *Human Lang. Tech. Conf.*, San Diego, CA., mar.