SPOKEN LANGUAGE INPUT FOR A PATIENT NOTE SYSTEM

Sasha Caskey*, Kathleen McKeown*, Desmond Jordan† and Julia Hirschberg*
*Department of Computer Science, Columbia University, New York, NY 10027 U.S.A.

†Departments of Anesthesiology and Medical Informatics, Columbia University College of Physicians and Surgeons New York, NY 10032 U.S.A.

Keywords: Speech recognition, Mobile devices, Electronic medical records, Natural language processing.

Abstract: In developing a system to help CTICU physicians write patient notes, we hypothesized that a spoken language interface for entering observations after physical examination would be more convenient for a physician than a more traditional menu-based system. We developed a prototype spoken language interface, allowing input of one type of information, with which we could experiment with factors impacting use of speech. In this paper, we report on a sequence of experiments where we asked physicians to use different interfaces, testing how such a system could be used as part of their workflow as well as its accuracy in different locations, with different levels of domain information. Our study shows that we can significantly improve accuracy with integration of patient specific and high coverage domain grammars.

1 INTRODUCTION

Our long term goal is the development of a system to help physicians create progress notes for patients in the Cardio-Thoracic Intensive Care Unit (CTICU). In the CTICU, a physician writes one to two notes daily recording objective and subjective findings for each of the approximately 28 patients under their care. When tending to a patient, a physician reviews the notes written by other physicians and thus, note writing is an important form of communication between physicians caring for the same patient, increasing continuity of care. At the same time, note writing is a time consuming process and takes away from time spent on patient care.

We are developing I^3 (Intelligent, Interactive, Multimodal Information Ecosystem for Healthcare), an interactive system that will reduce the time it takes to generate progress notes. I^3 will draw relevant information from the voluminous patient record where appropriate, will generate inferences from raw patient data identifying clinical problems and highlighting acute medical care issues, and will generate a skeletal note. Some information from the note is not available in the online patient record, however, and can only be provided by the physician based on physical observations during rounds. We hypothesize that a mobile, spoken language interface would make it easy for a physician to enter information about a patient at the point at which they have available time, regardless of where they were. We note that physicians have ready access to cell phones, which could provide an easy-to-use method for calling in information.

There are many unresolved issues, about the feasibility of using a spoken language interface within the hospital setting. Is there a point in time at which physicians could naturally incorporate a phone call into their workflow? Does the noisy environment of the CTICU make it too difficult to obtain acceptable accuracy for spoken language input? While speech has been successfully used in medical domains before (see Wang et. al. 99 and Owens 05), the CTICU is a more difficult environment. Domain-specific, telephony-based, spoken language interfaces have proven useful when the range of spoken inputs in response to a prompt is sufficiently limited by the domain; spoken language interfaces have gained in commercial use in car direction systems, in airline reservation systems, and in directory and weather information systems (Zue et. al 00). Can we use the constraints of the medical domain to adequately restrict expectations for input, increasing accuracy? Finally, even if we can encode sufficient domain restrictions on expected responses to prompts, will physicians provide only the

Caskey S., McKeown K., Jordan D. and Hirschberg J. (2009).
SPOKEN LANGUAGE INPUT FOR A PATIENT NOTE SYSTEM.
In *Proceedings of the International Conference on Health Informatics*, pages 323-328
DOI: 10.5220/0001554603230328

information requested, or will their utterances be lengthier than required, straying off topic?

In order to investigate these issues, we developed a prototype telephony-based spoken language dialog system. To determine when, where, and how to elicit restricted responses, we deliberately focused our system to gather one type of information, the identification of a patient problem, and experimented with factors that would impact its use. In this paper, we describe preliminary studies, the system we developed, and our evaluation to assess the feasibility of using spoken language to gather input. Our studies show that we can significantly improve recognition accuracy using a dialog system that integrates patient-specific grammars with high domain coverage. Our user study shows physician satisfaction with a dialog strategy giving them control over how information is entered.

2 PRELIMINARY STUDIES

Before developing the spoken language system we describe here, we carried out several preliminary studies and user interviews to try to understand how the system would best fit within physician workflow and how physicians would interact with such a system.

We hypothesized that the best time to gather information based on patient observations would be during formal medical rounds. At this time most of the content for note creation is readily available. We used an off-the-shelf commercial recognizer from ScanSoft, allowing the physicians to say whatever they wanted during rounds. While this should have worked well to obtain note content, we found that having to record their observations while teaching residents was too much of a cognitive load, rounds took twice as long as usual, and accuracy was very poor, as input was given during conversation with interruptions. We also experimented with an approach where physicians provided a two minute briefing about the patient following rounds, again using the ScanSoft system, but accuracy was much too low to be usable. These approaches both indicated the need for dialog with restricted input outside of rounds.

3 OVERVIEW OF THE SPOKEN LANGUAGE SYSTEM

We designed a spoken language system to collect the current problems of a patient in the CTICU. These problems are objectively defined ICD-9 codes (International Classification of Diseases and Related Health Problems)(ICD9), published by the World Health Organization, which provides codes to classify diseases, signs, symptoms, abnormal findings, complaints, social circumstances and external causes of injury/disease. Thus, we could experiment with different dialog strategies for collecting this well-defined, objective type of information and try different methods to increase system accuracy.

The system was deployed on the commercial grade Genesys VoiceBrowser Platform, using IBM‘s Websphere Voice Server (WVS) 5.1.3 for Automatic Speech Recognition (ASR) and Text-to-Speech (TTS)(VoiceServer). The application was developed in VoiceXML and grammars were written in SRGS format. Figure 1 shows how the different components in the I^3 application interact. I^3 was built as a web application and hosted from a Tomcat web server. It was responsible for generating the dynamic VoiceXML content, creating patient-specific grammars on the fly. Patient-specific information was supplied by an inference engine applied to the CTICU Electronic Medical Record (EMR - Eclipsys). This engine was designed to generate real-time ICD-9 problem codes for the patients being evaluated by the system.

Users access the system by calling from any telephone including landlines and cell phones. Once connected, the system asks the physician for the patient’s reference number (RefN). The reference number was used to associate the information collected by telephone with a unique set of patient specific clinical problems inferred over the EMR(PIS). For development we used an anonymized dataset. If the system cannot match the spoken RefN against the PIS, it proceeds to collect patient problems without access to information from PIS. The system explains to the user that s/he must speak one problem at a time and listens for input.

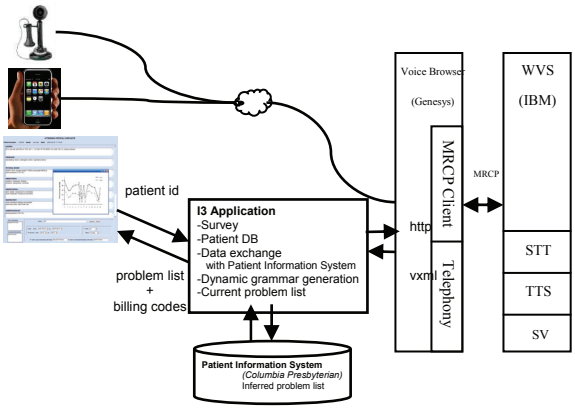

Figure 1: I^3 Architecture Description.

The user can stop the system at any time by saying "I'm done." The system supports collecting information for multiple patients in one call and also allows users to interrupt the system when they become familiar with it and need not listen to prompts. Once the session ends, the system records each of the current problems collected and their equivalent ICD-9 codes in the patient database.

Below is a sample interaction of a user with I^3.

System *I'm here to collect information about your patients. What is the patient's 7 digit Ref N?*
User *7432516*
System *Ok, now let's collect the current problems for this patient. Please speak one problem at a time. When you are done, just say "done." What's the next problem?*
User *acute respiratory failure*
System *Ok got it! What's the next problem?*
User *hyperglycemia*
User *done*
System *Given this patient's profile I would like to suggest 3 other possibilities, would you be willing to confirm them for me? Please say yes or no.*
User *yes*
System *Does the patient have pulmonary edema?*

The main challenges in the system from a speech recognition perspective are: number recognition, for the RefN; Yes/No recognition for confirmations and patient problem recognition, which is unique to this task. Number recognition is a well studied problem in the field of speech recognition and commercial recognizers claim to do quite well (recognition accuracy in the 90th percentile) under normal operating conditions.

Because we had domain knowledge but not much spoken data, we chose to model the problems using a grammar. We use domain knowledge to generate a Context Free Grammar (CFG) that represents the possible inputs for this specific domain. The grammar was constructed by encoding each of the 109 ICD-9 problems which are chargeable in the CTICU as a rule in the grammar. Note that an individual problem could consist of a single word (e.g., "hypertension") or a multi-word phrase (e.g., "systolic heart failure"). We then modified the grammar to account for variations in how these terms were spoken, allowing for optional words in the descriptions. For example: the rule *[acute] systolic heart failure*, where brackets denote an optional parameter, would cover both *systolic heart failure* and *acute systolic heart failure*. Paraphrases (e.g., abbreviations vs. full phrases) were added as additional rules. The resulting grammar contained 122 rules.

Our spoken language application was connected to a database containing patient information (the PIS in Figure 1) supplied by the inference engine, developed by physicians at Columbia Presbyterian Medical Center, which infers the current status of a patient by correlating perturbations in the patient parameters (e.g. lab results, vital statistics, and so on) with possible diagnoses; it produces a list of possible problems the patient could have. This list of ICD9 problems was designed to be a superset of all possible problems and tended to range from 10 to 30 entries. We used the problem list produced by the inference system to generate on-the-fly a patient-specific grammar which we hypothesized could increase accuracy.

4 METHODS

We structured our experiments in such way as to gain a better understanding into the following questions

- Is speech recognition accurate enough for the hospital environment?
- Can domain knowledge improve recognition accuracy?
- Will physicians provide expected answers from a short list versus unconstrained spoken input?
- How can we balance user and system initiative for efficient interaction?

4.1 Study Design

For the experiment, the I^3 application was configured so that it had access to information on the current patients within the CTICU. After IRB approval, only physicians who had patients in the CTICU used the system during the experiments. This allowed us to collect information about patients during the normal course of the physician's daily

routine. The experiments were conducted over multiple days, usually right after morning rounds.

Physicians called from the operating room, after the patient had been anesthetized, and from various locations in the ICU, including hallways, meeting rooms, patient rooms and nurses' stations; calls usually shortly after physicians examined their patients.

We compared two different modes of interaction with the physician users. The first was *system-driven*; the system provided the list of patient problems using inference engine output and the physician simply confirmed problems that s/he believed were associated with a patient. The second one was *user-driven* and allowed physicians to identify the problems they thought were relevant to the patient. Our system-driven approach requires the user to pay close attention to what the system proposes. In our user-driven system the user is free to enter information as s/he pleases. (Ackermann and Libosse) observed that systems which require more usage of human memory were more prone to errors and took longer to complete.

In the system-driven mode, once the physician had entered the patient's RefN, s/he was presented with a list of problems associated with that patient and asked to say *yes* to any that should be included in the note. At the end of this process, the physician was asked if s/he would like to augment the list with additional problems which may not have been deduced by the inference engine.

The user-driven mode allowed the caller to first speak all the problems for a given patient. The system would then compare the collected problems with those produced by the inference engine (usually a larger set) and would ask the caller if she would like the system to present the remaining inferred problems for inclusion in the patient's profile. If the caller responded with *yes*, the system would then list the remaining problems one at a time; the caller would say *yes* to include a problem in the profile. It was possible to skip this process by saying *done* at any point during the listing.

Since it relies on yes-no recognition, the system-driven mode should have the advantage of high accuracy in recognizing problems and should expose users to problems they might not have thought of. The user-driven mode should give users more control over the direction of the dialog, allowing them to enter the problems they felt were more important first, and hopefully reducing the cognitive overhead.

After collecting the speech, we also experimented with using different combinations of grammars and language models to recognize the input. We experimented with the ICD9 grammar and the ICD9 plus patient-specific grammar. When used alone the patient specific grammar yielded poor results. This is because physicians would often express problems using a more varied vocabulary and order than encoded in the grammar. UMLS has been shown to provide useful strings for natural language processing when properly selected (McCray, et al.) We experimented with a larger grammar constructed from the UMLS (16391 Entries) comparing recognition accuracy with the UMLS grammar alone, the UMLS plus the ICD9 grammar and the UMLS plus both the ICD9 and the patient specific grammar. We also experimented with various combinations of language models. Our language models were trained on data from various sources including: the UMLS database of disease descriptions; anonymized discharge notes and transcriptions of medical interviews. The model trained on all the data sets combined gave the best performance in terms of WER. See Table 1 LM+ICD9+Patient Specific for more details. There were 389k sentences and 49k words used to train our tri-gram language model.

After using one of the versions of the system, the users were asked to complete a survey about their experience with the system. They were asked to answer four questions, using a scale from one to five, where one generally meant a negative response and five a very positive one. They were allowed to speak or type their answers using the telephone's touchpad. The questions were:

Q1 *Would you find this system helpful for collecting patient information?*
Q2 *Does the system ask questions efficiently?*
Q3 *Was the system knowledgeable about your patient?*
Q4 *Would you want to use this system to retrieve information about your patients?*

5 RESULTS

During our experiment we received 44 calls from both physicians and students. The students were given a script with made-up patient information. The physicians called in from the CTICU and were asked to enter information about their current patients. The average number of turns per call was 18, where each turn is an interaction between the system and the user. The average call duration was 3 minutes and 42 seconds, and the longest call lasted almost 24 minutes.

The overall Word Error Rate (WER), which is a ratio of errors (measured by substitutions, insertions and deletions of words compared to a reference transcription) over the total number of words, was 23.39%. (Bangalore and Johnston) reported similar rates for a multimodal conversational system with 36% WER, when trained on out-of domain data. When they trained on in domain data they achieved WER of 25% in offline testing. For most spoken dialog applications Semantic Accuracy (SA) is more relevant then WER, though there is usually a correlation between the two. The SA for the overall application was 79.31%. To understand the difference between the two, consider a recognition output of *yes that's it* where the user actually said *uh yes.* This would increase the WER by three but since both have the same semantic tag (e.g. yes) the semantic result would be correct. In our grammars for patient problems we used the ICD-9 codes as the semantic annotation.

Table 1: WER/SA report broken down by grammar.

Categories	WER	SA	SentErr
RefN	7.71	74.19	-
Yes/No	20.78	94.91	-
Done	0	100	-
ICD9	62.42	50.64	50.64
ICD9 + patient specific	59.70	53.20	48.72
LM	56.67	55.76*	53.85
LM+ICD9+patient specific	**43.94**	**61.53***	**39.10**
Overall (base)	32.55	76.93	27.80
Overall (LM)	**23.39**	**79.31***	**24.56**

We calculated semantic accuracy for the grammars by comparing the semantic results to the semantic transcriptions. For the language model, we computed semantic accuracy using a unigram classifier.

We break these numbers down by answer category for WER and SA in *Table 1*. The most interesting category, is the one where patient problems are collected, handled by our ICD9 and patient-specific grammars. While we see WER rates of over 40%, SA is over the 60th percentile at 61.53%. To understand whether the patient-specific grammars are helping us, we ran the same data through a recognizer configured only with the ICD9 grammar (ICD9 only in *Table 1*.) We achieved a WER of 62.42% and a SA of 50.64% for the current problems section, which show that combining the patient-specific information improves the grammar by 5%. The best results were achieved by running both grammars and language model in parallel which improved recognition accuracy by 15% for SA and 26% for WER in the current problems section. If we look at our overall numbers (*Overall (LM)* in *Table 1*) we see our changes improved the system in all three categories (WER, SA, SentErr) when compared to the base system (e.g. *Overall (base)* in *Table 1*).

5.1 Survey Analysis

Each user was randomly routed to one of the two systems (user- or system-driven.) Once the user was finished with a patient s/he would be asked to answer four questions over the phone. We collected responses from eight physicians. Results are shown in Figure 2.

Figure 2 shows that users preferred the system that allowed them to say the problems first (e.g. *user-driven,*) as it scored higher in all questions.

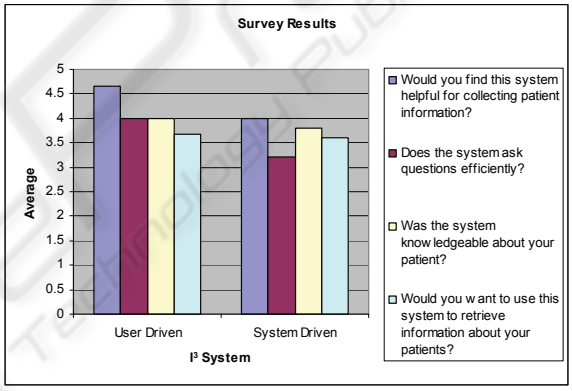

Figure 2: Survey results of questions for each system.

6 DISCUSSION

From the survey responses, it was clear that physicians found this system helpful for collecting patient information (avg. 4.5). While they found the user-driven method more efficient than the system-driven approach, they mentioned that they were at times frustrated by the speed and performance of the system, indicating that there is still room for improvement. One of the most populated areas, the nurse's station, provided the worst acoustic environment for speech recognition. We found that other areas such as patient rooms, O.R. and hallways had less effect on performance.

By analyzing the anonymized transcriptions we noted that physicians provided expected answers in over 60% of the inputs for patient problems, especially those problems which were common in

the CTICU. However, in other cases they tended to combine problems or modify their descriptions of them slightly. We plan to develop models that would allow some provision of multiple problems in one input.

Our recognition results were clearly affected by the adverse environment, though we plan to investigate methods to overcome these barriers, including building recognizers robust to machine noises (beeps.) and further smart dialog strategies.

6.1 Related Work

Other researchers have investigated the use of speech recognition for translation of physician diagnostic questions to the patient language using a linguistically based constrained domain grammar and achieve accuracy results of 69% (Bouillan et al). Strzalkowski et al experiment with the use of post-processing correction on a speaker-dependent system for recognition of dictated radiology reports. They first observe that recognition accuracy is far below the advertised 5% and averages 14.3%. Their linguistically based correction model reduced the rate to 11.3%. Thus, even in a quiet environment, with trained, speaker dependent models and close-talking microphones, error rate is just 10% above what we achieved. Bangalore & Johnston experiment with mixing grammar and rule based models in their MATCH application. MATCH is a multimodal application that enables mobile users to access subway and restaurant information for New York City. In their best results they report 25% for WER and 59.5% for SentErr. Our numbers for the overall application are in line with those and in fact show a slight improvement.

7 CONCLUSIONS

Our research shows that by using patient specific inferences, we can increase the semantic accuracy of clinical problem recognition by 5% and by augmenting our patient specific grammars with a large coverage language model, we can further increase semantic accuracy by 15% and WER by 26%. Thus, an approach which is tightly integrated with underlying patient-specific systems shows promise for providing a usable spoken language interface. Our survey shows that physicians find this to be a helpful method for providing patient information. Analysis of input shows that physicians provide short responses that directly answer the given questions.

REFERENCES

Wang SS, Starren JB. A Java speech implementation of the mini-mental status exam. Proceedings of the AMIA Annual Fall Symposium; Hanley&Belfus, Philadelphia, 1999: 435-439.

Owens, S.. New Operations in Speech. Speechteg Magazine, August 2005

Zue, V., S. Seneff, J. Glass, J. Polifroni, C. Pao, T. Hazen and L. Hetherington, Jupiter: A Telephone-based Conversational Interface for Weather Information, IEEE Trans. on Speech and Audio Processing, 8(1), 2000.

International Classification of Diseases, Ninth Revision, Clinical Modification, NCHS, 2007.

National Library of Medicine. Documentation, UMLS Knowledge Sources. 2007AC Edition, May 2007.

McCray AT, Bodenreider O, Malley J and Browne AC. Evaluating UMLS Strings for Natural Language Processing. Proc AMIA Symp 2001.

Ackermann, Chantal and Libossek, Marion (2006): System-versus user-initiative dialog strategy for driver information systems, In INTERSPEECH-2006, paper 1172-Mon2FoP.3.

Bouillon, P. and Halimi, S. and Rayner, M. and Hockey, B. A. Adapting a Medical speech to speech translation system (MedSLT) to Arabic. Proceedings of the 2007 Workshop on Computational Approaches to Semitic Languages: Common Issues and Resources

Strzalkowski, T. and Brandow, R. A Natural Language Correction Model for Continuous Speech Recognition. 5th Workshop on Very Large Corpora, EMNLP 1997

Bangalore, S. and Johnston, M. Balancing data-driven and rule-based approaches in the context of a Multimodal Conversational System. HLT-NAACL 2004: Main Proceedings

IBM Websphere Voice Server http://www.ibm.com/software/pervasive/voice_server/