REAL-TIME TELEPHONE-BASED SPEECH RECOGNITION IN THE JUPITER DOMAIN

James R. Glass, Timothy J. Hazen, and I. Lee Hetherington

Spoken Language Systems Group
Laboratory for Computer Science
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139 USA

ABSTRACT

This paper describes our experiences with developing a real-time telephone-based speech recognizer as part of a conversational system in the weather information domain. This system has been used to collect spontaneous speech data which has proven to be extremely valuable for research in a number of different areas. After describing the corpus we have collected, we describe the development of the recognizer vocabulary, pronunciations, language and acoustic models for this system, the new weighted finite-state transducer–based lexical access component, and report on the current performance of the recognizer under several different conditions. We also analyze recognition latency to verify that the system performs in real time.

1. INTRODUCTION

Over the past year and a half, we have developed a telephone-based, weather information system called JUPITER [14], which is available via a toll-free number for users to query a relational database of current weather conditions using natural, conversational speech.[1] Using information obtained from several different internet sites, JUPITER can provide weather forecasts for approximately 500 cities around the world for three to five days in advance, and can answer questions about a wide range of weather properties such temperature, wind speed, humidity, precipitation, sunrise etc., as well as weather advisory information.

The JUPITER system makes use of our GALAXY conversational system architecture which incorporates speech recognition, language understanding, discourse and dialog modelling, and language generation [12]. JUPITER has been particularly useful for our research on displayless interaction, information on demand, and robust spontaneous speech recognition and understanding. Since we attempt to understand all queries (i.e., not spot words), and do not constrain the user at any point in the dialog, it is crucial to have a high accuracy speech recognizer that covers, as much as possible, the full range of user queries. This paper describes our work in developing a robust recognizer in this domain.

When the system was first deployed in late April 1997, the error rates of our recognizer initially more than tripled our laboratory baselines, due in part to the mismatch between the laboratory training and actual testing conditions. The real data had a much larger variation in environment and channel conditions (often with very poor signal conditions), as well as a much wider range of speakers (we had no children in our training data for example, and had mainly trained on native speakers without regional accents), speaking style (spontaneous speech vs. read speech), language (both for within-domain queries, and out-of-domain queries), and other artifacts such as non-speech sounds and clipped speech due to the user interface (we do not currently allow for barge-in).

As we have collected more data we have been able to better match the users' vocabulary, and build more robust acoustic and language models. The result is that we have steadily reduced word and sentence error rates, to the point of cutting the initial error rates by over two thirds. In this paper, we describe the methods we have used to develop this recognizer and report on the lessons we have learned in moving from a laboratory environment to dealing with real data collected from real users. Our experience has shown us clearly that while there is no data like more data, there is also no data like *real* data!

2. CORPUS

Several different methods have been employed to gather data for the JUPITER weather information system. Beginning in February and March 1997, we created an initial corpus of approximately 3,500 read utterances collected from a variety of local telephone handsets and recording environments, augmented with over 1,000 utterances collected in a wizard environment [14]. These data were used to create an initial version of a conversational system which users could call via a toll-free number and ask for weather information. The benefit of this setup is that it provides us with a continuous source of data from users interested in obtaining information. Currently, we average over 70 calls per day, and have recorded and orthographically transcribed over 60,000 utterances from over 11,000 callers, all without widely advertising the availability of the system. On average, each call contains 5.6 utterances, and each utterance has an average of 5.2 words. The data are continually orthographically transcribed (seeded with the system hypothesis), and marked for obvious non-speech sounds, spontaneous speech artifacts, and speaker type (male, female, child) [4].

3. VOCABULARY

The vocabulary used by the JUPITER system has evolved as periodic analyses are made of the growing corpus. The current vocabulary contains 1893 words, including 638 cities and 166 countries. Nearly half of the vocabulary contains geography-related words.

This research was supported by DARPA under contract N66001-96-C-8526, monitored through Naval Command, Control and Ocean Surveillance Center.

[1] In the United States and Canada please call 888 573-8255 or visit http://www.sls.lcs.mit.edu/jupiter.

can_you	when_is	never_mind
do_you	what_about	clear_up
excuse_me	what_are	heat_wave
give_me	what_will	pollen_count
going_to	how_about	warm_up
you_are	i_would	wind_chill

Table 1: Examples of multi-word units in the JUPITER domain.

The design of the geography vocabulary was based on the cities for which we were able to provide weather information, as well as commonly asked cities. Other words were incorporated based on frequency of usage and whether or not the word could be used in a query which the natural language component could understand. The 1893 words had an out-of-vocabulary (OOV) rate of 2.0% on a 2506 utterance test set.

Since the recognizer makes use of a bigram grammar in the forward Viterbi pass, several multi-word units were incorporated into the vocabulary to provide for greater long-distance constraint and, in some cases, to allow for specific pronunciation modelling. This would allow for explicit modelling of word sequences such as "going to" or "give me" to be pronounced as "gonna" or "gimme" respectively. Common contractions such as "what's" were represented as multi-word units (e.g., "what_is") to reduce language model complexity, and because these words were often a source of transcription error anyway. Additional multi-word candidates were identified using a mutual information criterion which looked for word sequences which were likely to occur together. Table 1 shows examples of multi-word units in the current vocabulary.

4. PHONOLOGICAL MODELING

In the current JUPITER recognizer, words are initially represented as sequences of phonetic units augmented with stress and syllabification information. The initial baseform pronunciations are drawn from the LDC PRONLEX dictionary. The baseforms are represented using 41 different phonetic units with three possible levels of stress for each vowel. The baseforms have also been automatically syllabified using a basic set of syllabification rules. After drawing the pronunciations for the JUPITER vocabulary from the PRONLEX dictionary, all baseform pronunciations were then verified by hand. Vocabulary words missing from the dictionary were hand coded. Alternate pronunciations are explicitly provided for some words. In addition to the standard pronunciations for single words provided by PRONLEX, the baseform file was also augmented with common multi-word sequences which are often reduced, such as "gonna", "wanna", etc.

A series of phonological rules were applied to the phonetic baseforms to expand each word into a graph of alternate pronunciations. These rules account for many different phonological phenomena such as place assimilation, gemination, epenthetic silence insertion, alveolar stop flapping, and schwa deletion. These phonological rules utilize stress, syllabification, and phonetic context information when proposing alternate pronunciations. We have made extensive modification to these rules, based on our examination of the JUPITER data.

The final pronunciation network does not represent the words using the original 41 phonetic units utilized in PRONLEX. Instead, a set of 105 different units were used which include sub-phonetic, supra-phonetic and non-phonetic units in addition to standard phonetic units. For example, the recognizer treats most within-syllable vowel-semivowel sequences and some semivowel-vowel sequences as single units in order to better model the highly correlated dynamic characteristics of these sequences. Thus, the phonetic sequence [ow] followed by [r] is represented as a single segmental unit [or]. The recognizer also incorporates various non-phonetic units to account for non-linguistic speech transitions and speech artifacts, silences, and non-speech noise. The 105 units also retain two levels of stress for each vowel unit. An example pronunciation graph for the word "reports" is shown in Figure 1.

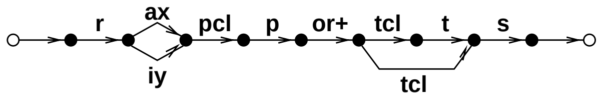

Figure 1: Pronunciation graph for the word "reports."

The arcs in the pronunciation graph can further be augmented with transition weights which give preference to more likely pronunciations and penalize less likely pronunciations. For JUPITER these weights were set using an error correcting algorithm on development data [13]. This algorithm adjusted the arc weights in an iterative fashion in order to reduce the error rate of the recognizer on development data.

5. LANGUAGE MODELLING

A class bigram language model was used in the forward Viterbi search, while a class trigram model was used in the backwards A^* search to produce the 10-best outputs for the natural language component. A set of nearly 200 classes were used to improve the robustness of the bigram. The majority of the classes involved grouping cities by state or country (foreign), in order to encourage agreement between city and state. In cases where a city occurred in multiple states or countries, separate entries were added to the lexicon (e.g., Springfield, Illinois vs. Springfield, Massachusetts). Artificial sentences were created in order to provide complete coverage of all of the cities in the vocabulary. Other classes were created semi-automatically using a relative entropy metric to find words which shared similar conditional probability profiles.

Since filled pauses (e.g., uh, um) occurred both frequently and predictably (e.g., start of sentence), they were incorporated explicitly into the vocabulary, and modelled by the bigram and trigram. Original orthographies were modified for training and testing purposes by removing non-speech and clipped word markers. When trained on a 26,000 utterance set, and tested on a 2506 utterance set the word-class bigram and trigram had perplexities of 18.4 and 17.1, respectively. These are slightly lower than the respective *word* bigram and trigram perplexities of 19.5 and 18.8. Note that the class bigram also improved the speed of the recognizer as it had 22% fewer connections to consider during the search.

6. ACOUSTIC MODELLING

The JUPITER system makes use of the segment-based SUMMIT recognizer which can utilize acoustic models based on segments or landmarks [3]. The nature of the acoustic models has varied over the course of system development, depending in large part on the amount of available training data. The current JUPITER configuration makes use of context-dependent landmark-based diphone models which require the training of both *transition* and *internal* diphone models. Internal diphones model the characteristics of landmarks occurring within the boundaries of a hypothesized phonetic segment, while transition diphones model the characteristics of landmarks occurring at the boundary of two hypothesized phonetic segments.

Given the 105 phonetic units used in the JUPITER system, and the constraints of the full pronunciation graph, there were 4,822 possible diphone transition models and 105 internal models needed. We have explored two different methods of modelling transitions. The first method trained models for frequently occurring transitions, and used one "catch-all" model for remaining transitions. This method worked well, and was simple to train. We currently use a reduced set of 782 equivalence classes which were determined manually to insure that an adequate amount of training existed for each class and that the elements of each class exhibited contextual similarity. This method performs slightly better than the "catch-all" method.

For each landmark, 14 MFCC averages were computed for 8 different regions surrounding the landmark, creating 112 different features. This initial feature set was then reduced from 112 features to 50 features using principal component analysis. The acoustic models for each class modeled the 50 dimensional feature vectors using diagonal Gaussian mixture models. Each mixture model consisted of a variable number of mixture components, dependent on the number of available training vectors for that class, with a maximum of 50 mixture components.

The diphone models were trained on a subset of data which excludes utterances with out-of-vocabulary words, clipped speech, cross-talk, and various types of noise. The training data also excludes all speech from speakers deemed to have a strong foreign accent. The full set of within-domain training utterances used for acoustic modelling consisted of 20,064 utterances, which was 76% of the available data at the time.

7. LEXICAL ACCESS

We have recently re-implemented the lexical access search components of SUMMIT to use weighted finite-state transducers with the goals of increasing recognition speed while allowing more flexibility in the types of constraints. We view recognition as finding the best path(s) through the composition $A \circ U$, where A represents the scored (on demand) acoustic segment graph and U the complete model of an utterance from acoustic model labels through the language model. We compute $U = C \circ P \circ L \circ G$, where C maps context-independent labels on its right to context-dependent (diphone in the case of JUPITER) labels on its left, P applies phonological rules, L is the lexicon mapping pronunciations to words, and G is the language model. Any of these transductions can be weighted. A big advantage of this formulation is that the search components operate on a *single* transducer U; the details of its composition are not a concern to the search. As such, U can be precomputed and optimized in various ways or it can be computed on demand as needed. This use of a cascade of weighted finite-state transducers is heavily inspired by work at AT&T [8, 10].

We have achieved our best recognition speed by precomputing $U = C \circ \text{minimize}(\text{determinize}((P \circ L) \circ G))$ for G a word-class bigram. This yields a deterministic (modulo homophones), minimal transducer that incorporates all contextual, phonological, lexical, and language model constraints [8]. For JUPITER, U has 89,452 states and 699,172 arcs. We apply a word-class trigram and compute N-best in a second pass utilizing an A^* search.

Test Set	# Utts	WER	SER
Standard	2506	24.4	43.0
In domain	1806	13.1	28.6
Male (In domain)	1290	9.8	24.1
Female (In domain)	274	13.6	31.8
Child (In domain)	242	26.3	48.8
Out of domain	700	61.8	80.1
Non-native (In domain)	3225	29.9	60.1
Expert (In domain)	354	2.3	7.1

Table 2: Current JUPITER performance on various test sets.

For greater system flexibility, we can compute $U = (C \circ \text{minimize}(\text{determinize}(P \circ L))) \circ G$, performing the composition with G on the fly during the search. For example, the use of a dynamic language model that changes during a dialogue would require this approach. However, with on-the-fly composition we have found that the system runs about 40% slower than for the fully composed and optimized U.

8. EXPERIMENTS

Over the course of the past year the JUPITER recognizer has had a steady improvement in its performance; this has been a result of both an increase in training data and improvements to the system's modeling techniques. The test data consists of sets of calls randomly selected over our data collection period. The current test set consists of 2506 utterances, of which 1806 were considered to be "in domain" as they were covered by the vocabulary, were free of partial words, crosstalk, etc. Of these sentences, 1290 were from male speakers, 274 from females, and 242 from children. Table 2 shows the performance of the JUPITER recognizer on this test set using word error rate (WER) and sentence error rate (SER) as the evaluation metrics.[2] As can be seen in the table, the system tends to perform reasonably when it encounters queries spoken by adults without a strong accent, that are covered by the domain, and that do not contain spontaneous, or non-speech artifacts. Females had 50% more word errors than males, while children had 300% more word errors than males. This is probably a reflection of the lack of training material for females and children. The system has considerable trouble (64.5% WER) with "out of domain" utterances containing out-of-vocabulary words, partial words, crosstalk, or other disrupting effects. This rate is artificially high, however, due to the nature of the alignment procedure with reference orthographies (e.g., partial words *always* cause an error for example, due to the nature of our mark-up scheme).

Table 2 also shows the performance on speakers judged to have strong foreign accents, who were not included in the standard test set. These data consisted of 3,225 in-domain utterances, and had an error rate more than double the baseline in-domain error rate. Finally, we also evaluated the recognizer on "expert" users (i.e., mainly staff in our group) who have considerable experience using the JUPITER system, but were not used for training. The system had extremely small error rates for these users. This behavior

[2] These error rates are slightly different from those reported in [4]. The reason is that we have increased pruning to achieve real-time performance.

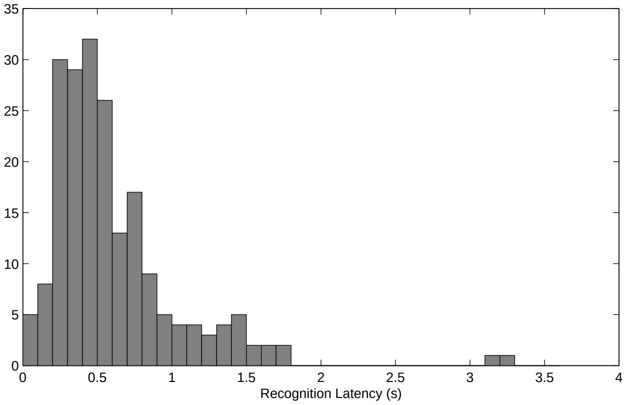

Figure 2: Histogram of JUPITER recognition latency.

is typical of users who become familiar with the system capabilities (a case of users adapting to the computer!).

Since JUPITER is a conversational system, rapid system response is critical. We consider a recognizer to run in real time if its latency (time after utterance is complete) is independent of utterance duration. An initial analysis of latency showed that while the latency was generally less than 1s, the worst cases took substantially longer. Not surprisingly, most of the worst cases were due to out-of-domain utterances containing out-of-vocabulary words. To combat the worst-case latency, we have added count-based beam pruning to limit the number of active nodes kept at any given point in time. Previously, we limited the beam solely with a score-based pruning threshold. With aggressive count-based pruning on a 300MHz Pentium II, we find a correlation coefficient between latency and utterance length of only -0.08, meaning that they are independent and we are achieving real-time performance. Figure 2 shows a histogram of the latency: 85% of the time the latency is less than 1s, and 99% of the time it is less than 2s.

9. DISCUSSION & FUTURE WORK

The speech recognizer described in this paper is only one component of the full JUPITER conversational system [11, 14]. The current interface between the recognizer and our language understanding component is via an N-best interface. Although we have reported only first-choice error rates in this paper, the understanding error rates are typically better, since many word confusions do not impact understanding.

There remain a considerable number of ongoing areas of research we are presently pursuing, which should help improve performance. Recent developments in probabilistic segmentation [7], near-miss modelling [1], heterogeneous classifiers [5], and tighter integration of linguistic knowledge [2], have shown improvements in our JUPITER baseline, although they have not yet been propagated to the data collection system.

The system to date has used a pooled speaker model for all acoustic modelling. It should be possible to achieve gains through speaker normalization, short-term speaker adaptation, and better adaptation to the channel conditions of individual phone calls. Adaptation may also be useful to help improve performance on non-native speakers. Since a phone call could have multiple speakers, we are exploring within-utterance consistency techniques that have given us gains elsewhere [6].

The data collection efforts have produced a gold-mine of spontaneous speech effects which are often a source of both recognition and understanding errors. For example, partial words typically cause problems for the speech recognizer. Another source of recognition errors is out-of-vocabulary words, which are often cities not covered in the vocabulary. These issues have caused us to begin work in confidence scoring, which was an area we had not previously addressed [9]. Finally, we plan to explore the use of dynamic vocabulary and language models, which may help to alleviate some of the unknown city problems.

10. ACKNOWLEDGMENTS

The JUPITER data collection effort was coordinated by Joe Polifroni, and heroically transcribed by Sally Lee. Jon Yi syllabified the LDC PRONLEX dictionary.

11. REFERENCES

[1] J. Chang, *Near-Miss Modeling: A Segment-Based Approach to Speech Recognition*. Ph.D. thesis, MIT, May 1998.

[2] G. Chung and S. Seneff, “Improvements in speech understanding accuracy through the integration of hierarchical linguistic, prosodic, and phonological constraints in the JUPITER domain,” in *Proc. ICSLP*, Sydney, 1998.

[3] J. Glass, J. Chang, and M. McCandless, “A probabilistic framework for feature-based speech recognition,” in *Proc. ICSLP*, Philadelphia, PA, pp. 2277–2280, 1996.

[4] J. Glass and T. Hazen, “Telephone-based conversational speech recognition in the Jupiter domain,” in *Proc. ICSLP*, Sydney, 1998.

[5] A. Halberstadt and J. Glass, “Heterogeneous measurements and multiple classifiers for speech recognition,” in *Proc. ICSLP*, Sydney, 1998.

[6] T. J. Hazen, *The Use of Speaker Correlation Information for Automatic Speech Recognition*. Ph.D. thesis, MIT, January 1998.

[7] S. Lee and J. Glass, “Real-time probabilistic segmentation for segment-based speech recognition,” in *Proc. ICSLP*, Sydney, Australia, 1998.

[8] M. Mohri, M. Riley, D. Hindle, A. Ljolje, and F. Pereira, “Full expansion of context-dependent networks in large vocabulary speech recognition,” in *Proc. ICASSP*, Seattle, WA, vol. 2, pp. 665–668, May 1998.

[9] C. Pao, P. Schmid, and J. Glass, “Confidence scoring for speech understanding,” in *Proc. ICSLP*, Sydney, Australia, 1998.

[10] F. Pereira and M. Riley, “Speech recognition by composition of weighted finite automata,” in *Finite-State Language Processing* (E. Roche and Y. Schabes, eds.), pp. 431–453, Cambridge, MA: MIT Press, 1997.

[11] J. Polifroni, S. Seneff, J. Glass, and C. Pao, “Evaluation methodology for a telephone-based conversational system,” in *Proc. First Int’l. Conference on Language Resources and Evaluation*, Granada, Spain, pp. 43–49, 1998.

[12] S. Seneff, E. Hurley, R. Lau, C. Pao, P. Schmid, and V. Zue, “Galaxy-II: A reference architecture for conversational system development,” in *Proc. ICSLP*, Sydney, Australia, 1998.

[13] V. Zue, J. Glass, D. Goodine, M. Phillips, and S. Seneff, “The SUMMIT speech recognition system: Phonological modelling and lexical access,” in *Proc. ICASSP*, Albuquerque, NM, pp. 49–52, 1990.

[14] V. Zue, S. Seneff, J. Glass, L. Hetherington, E. Hurley, H. Meng, C. Pao, J. Polifroni, R. Schloming, and P. Schmid, “From interface to content: Translingual access and delivery of on-line information,” in *Proc. Eurospeech*, Rhodes, Greece, pp. 2047–2050, 1997.