MUXING: A TELEPHONE-ACCESS MANDARIN CONVERSATIONAL SYSTEM[1]

C. Wang, S. Cyphers, X. Mou, J. Polifroni, S. Seneff, J. Yi, and V. Zue

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

ABSTRACT

MUXING is a telephone-based conversational system that allows users to access weather information in Mandarin Chinese over the telephone. Although MUXING utilizes the same architecture as well as most of the same human language technology components as its English predecessor, JUPITER, some modifications to the system were necessary to account for differences between English and Mandarin Chinese. In addition, the weather database needed to be modified to reflect regions of greater interest to potential Chinese users. This paper describes our system development effort, paying particular attention to Mandarin-specific changes to the original JUPITER system.

1. INTRODUCTION

For the past decade, our group has been conducting research leading to the development of conversational systems that enable users to access and manage information using spoken dialogue. In this context, multilinguality has always been an important research topic. Our approach to developing multilingual conversational systems is predicated on the assumption that it is possible to extract from the input a *common*, language-independent semantic representation, an *interlingua*. To promote portability, we have adopted the strategy of requiring that each component in the system be as language transparent as possible. Where language-dependent information is required, we have attempted to isolate it in the form of external models, tables, or rules. Thus far, we have applied this formalism successfully across several languages and domains [1, 2].

In 1997, we introduced the JUPITER weather information system in English [3]. As JUPITER gains maturity, it has become the platform for our multilingual spoken language research effort. This paper describes MUXING[2], a conversational system providing weather information in Mandarin Chinese. MUXING employs the same Galaxy Communicator architecture [4] as its English predecessor. It also utilizes most of the same human language technology components, although some modifications were necessary for differences between English and Mandarin Chinese. In addition, the weather database needed to be modified to reflect regions of greater interest to potential Chinese users. This paper describes our system development effort, focusing on Mandarin-specific issues in recognition, language modeling, translation of the weather reports, database, and synthesis. Due to space limitations, readers are referred to our other publications for a background description of JUPITER.

2. SPEECH RECOGNITION

Creating a lexicon for Mandarin is difficult, since the definition of a word is often arbitrary, leading to potential redundancy. For example, a city can be referred to by its name (e.g.,“beijing”), or by adding modifiers to the name, (e.g., “beijing city”, “beijing area”, or even “beijing city area”). If each of these is represented as a single word, the lexicon could become unwieldy. There is clearly a trade-off between defining larger lexical units by “underbarring” frequently appearing sequences of common words, at the expense of a substantial growth in the size of the lexicon.

Our solution to this problem is to use underbars sparingly, defining only the obvious words, such as “bei3_jing1(beijing)”, “shi4(city)”, and “di4_qu1(area).” To enhance the power of the language model, we have developed a mechanism to automatically create a statistical hierarchical language model, based on context free rewrite rules. We adopt pinyin as the symbol set, rather than characters. Thus, in the example, the concept of “beijing” city can be realized as “bei3_jing1” followed by optional modifiers such as “shi4,” as well as many other variants.

In the remainder of this section, we will first identify our baseline system and data. We will then describe the acoustic modeling aspects, followed by the language modeling approach. Finally, we summarize recognition results, for a number of different experimental conditions.

2.1. Baseline System

The recognizer uses the segment-based SUMMIT system [8]. The vocabulary has 750 words, covering about 400 place names and other common words used in weather queries. On average, each word contains 2.3 syllables. Chinese syllable initials and finals (i.e., onsets and rhymes) are used as acoustic model units. The baseline configuration uses segment models, a class bigram in the forward Viterbi search and a class trigram in the backward $A*$ search.

Acoustic models were trained using 1,250 within-domain *read* utterances, augmented with some 9,000 utterances from the YINHE domain [2]. Many of the YINHE utterances contain English words as well as Chinese words that are out of the vocabulary of the MUXING domain. In order to utilize as much data

[1] This work was supported by DARPA under contract N66001-99-1-8904 monitored through Naval Command, Control and Ocean Surveillance Center, by the National Science Foundation under Grant No. IRI-9618731, and by a contract from the Industrial Technology Research Institute.

[2] “Muxing” is the Chinese name for the planet Jupiter.

as possible, we configured a syllable recognizer specifically for deriving forced transcriptions for training. We also added a filler acoustic model to account for all the English words, so that we could skip the English words in an utterance and still use the rest of it for training. We were able to double the effective number of training tokens using this technique (from 236,760 to 440,947).

Once the system was available through telephone access, we collected an additional 1000 spontaneous utterances of users interacting with MUXING. We divided the MUXING data into two approximately equal sets, a 400 utterance development set used for tuning recognizer parameters, and a 450 utterance test set for final evaluation.

2.2. Boundary Model Training

In addition to segment models, SUMMIT also utilizes boundary models to provide contextual constraints. However, we have not been able to train boundary models for the Mandarin recognizer in the past, due to the difficulty in manually grouping cross-phone boundary classes based on phonological knowledge, especially when training data are very sparse. To address this problem, we have implemented a data-driven approach to derive boundary classes automatically, using a decision-tree based clustering technique.

Since it would be too computationally costly to derive all classes from a single pool of data, we designed a two step process to improve efficiency. We first define broad boundary classes, based on some limited phonological knowledge of Chinese. For example, syllable initials are grouped into stops, fricatives, etc., syllable finals on the left side are divided according to whether they have a nasal ending, and finals on the right side are divided according to their vowel nucleus, etc. Altogether, 41 broad boundary classes were obtained in this way.

Each of the broad boundary classes is then divided into more refined classes using decision tree clustering. Starting from a root node, each split seeks the leaf node and the question that maximizes the increase in log likelihood based on a single Gaussian density function. The process stops when the log likelihood increase falls below a threshold, or when a minimum count is reached at each leaf. This ensures adequate training data for the final boundary classes. The questions are based on phonological features, such as place of articulation, the voiced/unvoiced distinction, and retroflection. Individual phones are also considered, if data are sufficient. A total of 313 questions were asked about the left and right side of the boundary models. However, only a subset of them were effective for each broad class. We chose the stop criterion as a minimum of 50 data samples per leaf node, which reduced the boundary classes from the original 1,734 in the training data down to about 760.

2.3. Language Modeling

As mentioned previously, we have come up with a solution for language modeling which solves the tokenization problem while keeping the vocabulary down to a reasonable size. The language model, which is a stochastic context-free grammar, consists of two major components: (1) a class *n*gram, and (2) a set of context-free rules used to recursively expand each class into terminal words. The language model is trained by parsing a corpus using the TINA natural language formalism [7]. Each grammar rule is decomposed into a set of trigrams linking left siblings with possible right siblings, in the context of the parent. A set of distinguished categories are identified as classes for the top-layer class *n*gram. Once the grammar is trained, it is written out as a finite-state transducer, which can then be composed with the lexicon to define the search space for recognition.

An example sentence, decomposed into grammar rules, is shown in Figure 1. The highlighted categories are linked through the class *n*gram probabilities. Spacio-temporal trigrams are retained only for the portions of the parse tree below these categories.

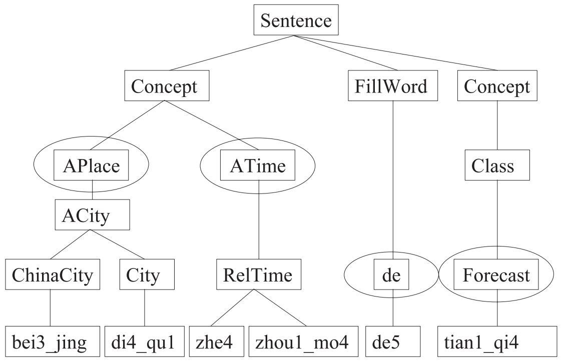

Figure 1: A simple example to illustrate the stochastic context free language model used in recognition. The highlighted categories are the ones involved in a class *n*gram.

2.4. Recognition Performance

Table 1 summarizes the recognition performance using syllable error rate on the development set and test set. After applying boundary models in addition to segment models, the syllable error rate was reduced to 15.1% from the baseline performance of 23.1% on the test set. When the class bigram and trigram were replaced with the statistical hierarchical language model obtained from TINA, the error rate was reduced to 19.3%. When both the boundary models and the TINA-based language model were utilized, the error rate was further reduced to 13.9%.

Configuration	Dev Set	Test Set
Baseline	20.1	23.1
+ Boundary	14.5	15.1
+ Tina LM	17.9	19.3
+ Both	13.7	13.9

Table 1: Summary of syllable error rate in percentage.

3. LANGUAGE UNDERSTANDING

The recognizer produces a set of N-best hypotheses, which are converted into a syllable graph and parsed by TINA, using a Viterbi search, to produce the final meaning representation, or *semantic frame*. The search algorithm utilizes linguistic scores that are trained using TINA's spacio-temporal trigram model. The underbars are discarded prior to parsing, such that the final organization of syllables into words is not necessarily the same as the organization produced by the recognizer. Word representations remain in pinyin format throughout, such that the character representations are never identified. The grammar for

Semantic Frame:

```
{c weather_event
  :pred
   {p becoming
      :topic
       {q weather_act
          :conditional "mostly"
          :name "sunny"
          :pred
           {p in_time
              :topic
               {q time_of_day
                  :modifier "late"
                  :name "afternoon" } } } } }
```

Pinyin Paraphrase:

bang4 wan3	zhuan3	zhu3 yao4 wei2	qing2 tian1
(next to evening	becoming	mostly	sunny)

Figure 2: Example semantic frame for the sentence, “becoming mostly sunny in late afternoon,” along with its pinyin paraphrase.

understanding is distinct from the grammar used to train the recognizer language model, containing a much more detailed syntactic and semantic description of the domain.

4. RESPONSE GENERATION

Response generation was probably the most challenging aspect of MUXING. This is because the weather reports are available mainly in English, making response generation essentially a translation task. Weather forecasts from the U.S. National Weather Service, a main source for U.S. cities, are manually prepared by weather forecasters, who are often quite expressive in their narrations. Since Mandarin word order is significantly different from English, it was interesting to determine whether our tools for generation were sufficiently sophisticated to deal with the word order constraints. We also encountered some interesting cases where translations required many-to-many mappings, and where word sense disambiguations were necessary, even though the domain of knowledge is restricted to weather.

TINA parses all incoming weather reports into semantic frames, illustrated in Figure 2. These are paraphrased into Mandarin by the newly developed GENESIS-II generation component [5]. The outputs can appear in three distinct textual forms (pinyin, and simplified and traditional Chinese characters), using a common grammar file and separate vocabulary files for each format.

GENESIS-II applies recursive rewrite grammar rules that work their way top down through the frame, beginning with the main clause. For the most part, grammar rules were straightforward, with a simple placement of constituents in the order in which they should appear in the surface form realization. However, several interesting challenges were identified, some of which could be solved by altering either (1) the parse tree itself, to reorganize the hierarchy in the frame, or (2) the mappings to a semantic frame, for example, by distinguishing explicitly between *in_time* and *in_loc*. In other cases, a major reorganization of the surface string could be effected by using a newly introduced “pull” mechanism, which allows a higher level constituent to pre-generate components from inside one or more of its descendents. This technique was necessary for generating the temporal modifier in the example, which, for Chinese, must appear *before* the verb “becoming” (“zhuan3”).

Another interesting aspect of the example is the technique for combining the two words “late” and “afternoon” to generate the Chinese translation “bang4 wan3.” The problem is that it is not possible to choose a single meaning for “late” that is correct for “morning,” “afternoon,” and “evening,” and likewise, it is not possible to choose a single word for “afternoon” that is correct for “early,” “late,” or no qualifier. Figure 3 shows the vocabulary entries that achieve this multi-word translation, where the modifier, “late” sets the selector *$:late*, which then controls the lookup from the vocabulary entry for “afternoon.”

late	$:as_noun “wan3 xie1 shi2 hou4” ; $:late
afternoon	“xia4 wu3” $:late “bang4 wan3”
evening	“wan3 jian1” $:early “bang4 wan3” $:late “ye4 jian1”

Figure 3: Vocabulary entries to illustrate mechanisms to creatively combine two words into a single mapping.

The weather domain contains a rich set of descriptive modifiers, whose temporal order varies significantly from language to language. An example is “brief light early morning rain.” We assigned these modifiers to a small number of logical groups such as “:temp_qualifier,” (temporal) and “:loc_qualifier” (location). However, some words were inappropriately assigned according to their expected positional constraints in Chinese. GENESIS-II provides a mechanism to allow a vocabulary item to generate a *[:key value]* pair that is inserted directly into the semantic frame. This mechanism leads to essentially post-hoc editing of the frame to reassign inappropriately labelled keys, and, consequently, to reposition their string expansions in the surface form generation. In the example in Figure 4, the word “brief,” originally assigned as a “:qualifier,” generates to a null string, but, as a side effect, gets retagged as a “:time_qualifier,” with the appropriate Chinese translation as its value. It subsequently obeys ordering constraints of all other temporal qualifiers, rather than those of qualifiers, as desired.

The second example in Figure 4 concerns word sense disambiguation. The word “light” translates differently for “light rain,” (*xiao3*) and “light wind” (*wei1*). The grammar rule for “wind” sets up a $:WIND *selector*, which then controls the selection of the appropriate word sense.

(1)	brief	“” :time_qualifier “zai4 duan3 shi2 jian1 nei4”
(2)	light	“xiao3” $:wind “wei1”

Figure 4: Lexical entries to illustrate (1) mechanisms to reassign the key for a vocabulary entry, and (2) mechanisms for word sense disambiguation. See text for details.

5. WEATHER DATABASE

In addition to issues concerning recognition and understanding of Mandarin Chinese, we felt that MUXING should be able to talk about a larger set of Chinese cities than JUPITER. We have added a total of 96 more cities specifically for MUXING, 78 in mainland China and 18 in Taiwan. This has required adding a new source of weather data from the Web to find forecasts for these cities.

With so many more cities, we could no longer enumerate the entire set in response to queries such as "What cities do you know in China?" The JUPITER domain makes use of a hierarchy of cities, states, and regions in the U.S., and an equivalent hierarchy of cities, provinces, and regions was created for mainland China and Taiwan for MUXING. Queries that result in too many cities to speak are automatically processed through this hierarchy (in both English and Chinese), resulting in responses aimed at focusing the user on a particular geographic region (e.g., "I know of the the following regions in China, ...").

Both MUXING and JUPITER use the same dialogue manager and database server, meaning that all forecasts are available and able to be understood and paraphrased in either language. However, we believed that most U.S.-based, English-speaking users of JUPITER would be concerned with a larger set of American cities while Chinese-speaking users of MUXING would know and care about a larger set of Chinese cities. Since information about the specific domain language is encoded in the frame that is sent to the dialogue manager, we are able to restrict queries to JUPITER to a small set of well-known, large Chinese cities, while MUXING is correspondingly more restricted in the U.S. cities it knows. Once we have obtained a full set of responses to a query such as "What cities do you know in Shandong province" or "Where is it raining in New England" we use GENESIS-II's paraphrasing mechanism as a way of filtering out cities that we have decided to treat as unknown to either JUPITER or MUXING.

6. SYNTHESIS

Currently, MUXING utilizes a Mandarin text-to-speech system provided to us by the Industrial Technology Research Institute. We have recently assembled a preliminary Mandarin Chinese speech synthesis system using an updated version of our corpus-based waveform concatenation synthesizer, ENVOICE [6].

The ENVOICE synthesizer performs unit selection using a phonological criterion, whereby concatenation and substitution costs are determined based on local phonetic context. Equivalence classes are used to group phones into sets which exhibit similar concatenation behavior, and can thus share the same concatenation or substitution costs. For our preliminary work with Mandarin Chinese, equivalence classes were essentially the same as for English (e.g., manner and place of articulation), except that the phoneme inventory was based on Mandarin syllable initials and finals, as it was for the recognizer. Other minor modifications included a new contextual class for retroflexed consonants used in calculating substitution costs.

Lexical modeling uses a two-stage process, implemented with finite-state transducers, to expand the tokenized Chinese characters generated by GENESIS-II into syllable initials and finals. First, tokenized characters (encoded in UNICODE [3]) are mapped into individual pinyin symbols. Then, these pinyin symbols are expanded into initials and finals, the fundamental synthesis units.

Our efforts in developing this preliminary system are continuing. Some long-term research issues in Chinese synthesis include treating lexical tones along with their associated tone sandhi phenomena and increasing coverage of vocabulary and sentence structures. Improving vocabulary coverage may require study of the transliteration process for foreign names. There are often patterns in how phonetic and graphemic sequences from foreign languages are translated to Chinese characters. For example, "ton" in "Boston", "Houston", and "Washington" all translate to the same Chinese character, pronounced "dun4." This shared unit could then be reused. These and other issues will form the basis for further synthesis research.

7. SUMMARY

This paper describes our effort in developing MUXING, a telephone-based conversational system that allows users to access weather information in Mandarin Chinese over the telephone. In particular, we focused our discussion on Mandarin-specific changes to our original JUPITER system.

MUXING is a system under active development. We have thus far used it to collect some 1200 sentences from 235 Chinese speakers in the Greater Boston area, and have used these data for system evaluation and refinement. In the coming months, we expect to continue this data collection process, perhaps in collaboration with organizations in a Chinese speaking region, so that it may achieve performance similar to JUPITER.

8. REFERENCES

1. J. Glass, G. Flammia, D. Goodine, M. Phillips, J. Polifroni, S. Sakai, S. Seneff, and V. Zue, "Multilingual Spoken Language Understanding in the MIT VOYAGER System," *Speech Communication*, vol. 17, no. 1-2, pp. 1-18, 1995.
2. C. Wang, J. Glass, H. Meng, J. Polifroni, S. Seneff, and V. Zue, "YINHE: A Mandarin Chinese Version of the GALAXY System," *Eurospeech'97*, pp. 351-354, Rhodes, Greece, 1997.
3. V. Zue, S. Seneff. J. R. Glass, J. Polifroni, C. Pao, T. J. Hazen, and I. L. Hetherington, "JUPITER: A Telephone-based Conversational Interface for Weather Information," *IEEE Trans. on Speech and Audio Processing*, vol. 8, no. 1, pp. 85-96, 2000.
4. S. Seneff, E. Hurley, R. Lau, C. Pao, P. Schmid, and V. Zue, "Galaxy-II: A Reference Architecture for Conversational System Development," *ICSLP'98*, pp. 931-934, Sydney, Australia, 1998.
5. L. Baptist and S. Seneff, "GENESIS-II: A Versatile System for Language Generation in Conversational System Applications," *These Proceedings*, Beijing, China, 2000.
6. J. Yi, J. R. Glass, and I. L. Hetherington, "A Flexible, Scalable Finite-State Transducer Architecture for Corpus-Based Concatenative Speech Synthesis," *These Proceedings*, Beijing, China, 2000.
7. S. Seneff, "TINA: A Natural Language System for Spoken Language Applications," *Computational Linguistics*, Vol. 18, No. 1, pp. 61-86, 1992.
8. J. R. Glass, J. Chang, and M. McCandless *et al.*, "A Probabilistic Framework for Feature-based Speech Recognition," *ICSLP'96*, pp. 2277-2280, Philadelphia, PA, USA, 1996.

[3] We have been using UNICODE, an international encoding understood by most Web browsers, for multi-lingual development. See http://www.unicode.org for more details.