PRELIMINARY DEVELOPMENTS IN CUHK'S TRILINGUAL SPEECH INTERFACE FOR FINANCIAL INFORMATION INQUIRIES

Helen M. MENG[1], Tan LEE[2], Tien Ying FUNG[1], Wai Ching TSUI[1], Pui Yuk YAU[1]
Human-Computer Communications Laboratory
[1]*Department of Systems Engineering and Engineering Management*
[2]*Department of Electronic Engineering*
The Chinese University of Hong Kong, Shatin, N.T., Hong Kong, China
hmmeng@se.cuhk.edu.hk

Abstract This paper describes our preliminary effort in developing a trilingual speech interface for financial information inquiries. Our foreign exchange inquiry system consists of: (i) Monolingual and trilingual speech recognizers, which receives the user's spoken input, in the form of microphone speech. (ii) A real-time data capture component to retrieve financial data. (iii) A trilingual speech generation component, which generates English and Chinese text based on the raw financial data. The generated text is then transformed into spoken presentations. English text is processed by the FESTIVAL. Chinese text is sent to our syllable-based synthesizer, which employs a concatenative resequencing technique to produce spoken presentations in Putonghua or Cantonese.

1. Introduction

This paper describes our preliminary effort in developing a trilingual speech interface for financial information inquiries. Our speech interfaces supports the languages of Hong Kong – Cantonese, Putonghua and English. We have selected foreign exchange (FOREX) as our application domain. Our foreign exchange inquiry system consists of: (i) Monolingual and trilingual speech recognizers, which receives the user's spoken input, in the form of microphone speech. (ii) A real-time data capture component, which continuously updates financial data from the Reuters satellite feed, and retrieves the relevant data based on the user's request. (iii) A trilingual speech generation component, which generates English and Chinese text based on the raw financial data. The generated text is then transformed into spoken presentations. English text is processed by the FESTIVAL. Chinese text is sent to our syllable-based synthesizers. We have devised a technique which employs a concatenative resequencing to produce natural-sounding presentations in Putonghua or Cantonese. Figure 1 shows the overall architecture of our system.

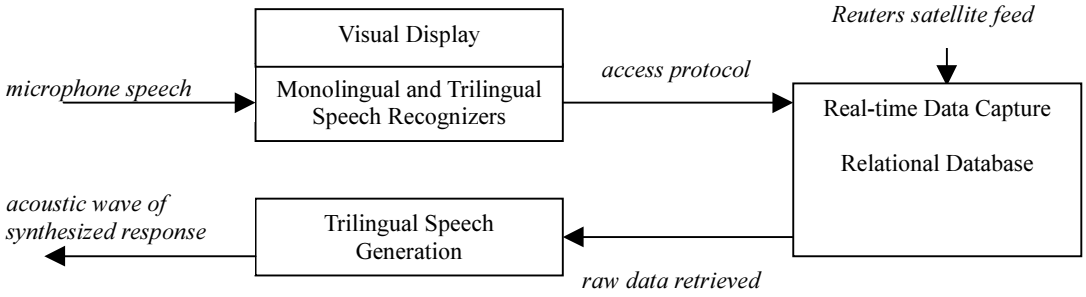

Figure 1. Overall system architecture of foreign exchange inquiry system

2. The Visual Interface

Our speech interface is enhanced with a Java applet visual display. Push-buttons enable users to select one of three monolingual recognizers (Cantonese, Putonghua or English), or the trilingual recognizer. The visual display is also furnished with push-buttons which provide audio help instructions in all the three languages. Another pair of push-buttons (labeled 'First Currency' and 'Second Currency') invokes the recording process for each of the two currencies of which the user wishes to find the exchange rate. These currency buttons, when pushed, plays an audio signal to instruct the user to speak after hearing a tone. Such instructions ease the process of end-point detection during recording. The layout and flow of the visual interface are depicted in the screen dumps in Figure 2. At each stage, we aim to provide sufficient feedback to the user with regards to the processing status of the system, as well as outcome of recognition.

3. Speech Recognition

As mentioned previously, we have developed four speech recognizers in total. The trilingual recognizer is formed by consolidating the vocabulary sets of the three monolingual recognizers. We have approximately 270 lexical entries in all, covering 36 international currencies. The lexical items include names of currencies, countries and their combinations. We also cover the common (colloquial) variations by which users may refer to the currencies. The recognizers are HMM-based [1], using word models with single Gaussian probability distributions. The HMMs are configured with 5 states per syllable for each Cantonese or Putonghua word model, and 8 states per syllable for each English word model. These state settings were determined by optimization based on training data. Our experimental corpus consists of gender-balanced, microphone

recordings collected from 60 speakers. In our experiments, we jack-knifed the data from 50 speakers for training and 10 speakers for testing. The overall performance accuracies were averaged among the 5 experimental runs.

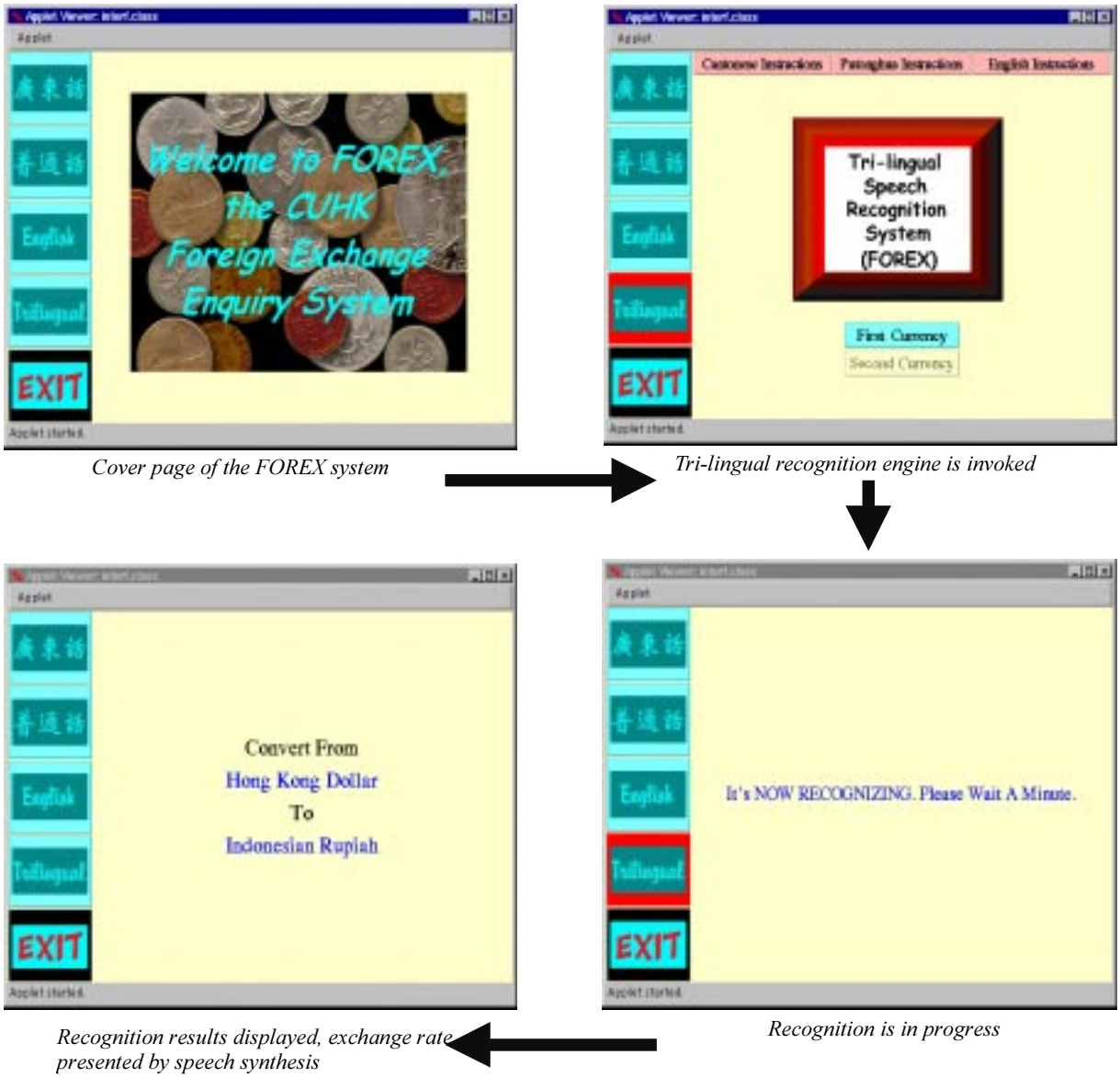

Figure 2. Interface layout and system flow of the FOREX inquiry system

3.1 Recognition Results

Figure 3 shows the performance of our speech recognizers. Results are based on the test sets for each monolingual recognizer. As we compare the performance between the monolingual and trilingual recognizers for each language, we observe a performance degradation due to a larger number of vocabulary items present in the trilingual recognizer. The degradation for Putonghua is particularly large, possibly because the monolingual recognizer only has a vocabulary size of 40, compared to the trilingual recognizer which has 270. Degradation in Cantonese and English are comparable.[1]

[1] We have also performed another experiment investigating the effect of adding filler words to our system. A filler word is defined as sounds like "um", "ah", "la" etc., which are characteristic of spontaneous speech. Speakers commonly insert them at the beginning or end of their utterances. Injection of filler words (e.g. um, ah, ok) led to further performance degradations, ranging from 1 to 4% among the recognizers. Confusion between the filler words and the keyword is a main cause for such degradations.

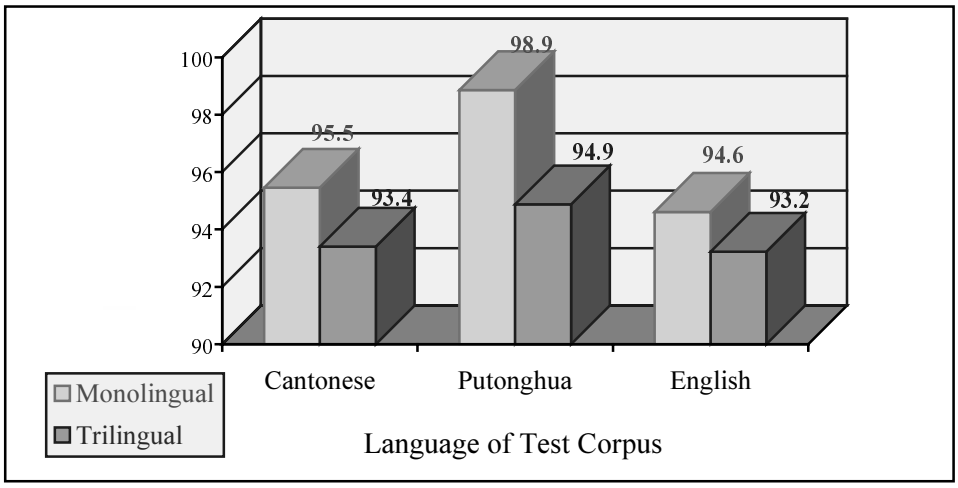

Figure 3: Performance comparison between the monolingual and trilingual recognizers for each language.

3.2 Error Analyses

Table 1 shows some examples of recognition errors, organized according to confusions *within* a language (intra-language confusions), as well as *across* languages (cross-language confusions). In the table, we have included the language labels E*=English, C*=Cantonese, and P*=Putonghua. For Cantonese, we have also included the phonetic transcription based on the LSHK standard [2], e.g. the first reference currency is the Cantonese version of the South Koren Won, pronounced as "waan4", where the number '4' indicates the fourth tone. Similarly, for Putonghua, we have included the phonetic transcription based on Hanyu Pinyin [3], e.g. the fourth reference currency is the Putonghua version of the US dollar, pronounced as two tonal syllables – "mei3_chao1".

Rows	Reference Currency	**Intra-language Confusion**		**Cross-language Confusion**	
		Recognizer's Hypothesis	Frequency of Confusion	Recognizer's Hypothesis	Frequency of Confusion
1	waan4, (won, C*)	toi4_waan1 (taiwan, C*)	3.33%	won (E*)	15%
2	sing1_gaa1_bo1 (singapore, C*)	san1_gaa1_bo1 (singapore, C*)	1.67%	singapore (E*)	15%
3	pat1 (baht, C*)	---	---	pound (E*)	3.33%
4	mei3_chao1 (US dollar, P*)	---	---	mei5_caau1 (US Dollar, C*)	36.67%
5	mei5_caau1 (US dollar, C*)	bei2_sok3 (peso, C*)	1.67%	mei3_chao1 (US dollar, P*)	25%
6	ren2_min2_bi4 (renminbi, P*)	---	---	renminbi (E*)	21.67%
7	ying1_bang4 (pound, P*)	---	---	jing1_bong6 (pound, P*)	8.33%
8	yuan (E*)	won (E*)	1.67%	yuan2 (yuan, P*)	8.33%
9	renminbi (E*)	germany (E*)	1.67%	ren2_min2_bi4 (renminbi, P*)	11.67%

Table 1. Intra-language and Cross-language confusions based on our recognition results. (E*=English, C*=Cantonese, P*=Putonghua)

The breakdown of these confusions in terms of the trilingual recognizer's errors are shown in Table 2.

	Percentage of the Test Set (%)		
	Cantonese Test Set	Putonghua Test Set	English Test Set
Correct Recognition	93.4	94.9	93.2
Intra-language Confusion	4.4	1.1	3.8
Cross-language Confusion	2.2	4.0	3.0

Table 2: Distribution of correct and errorful test tokens of the trilingual recognizer.

From the two tables, we observe that the additional vocabulary items in the trilingual recognizer led to cross-lingual confusions, which contributed to the additional errors (ranging from 33.3% to 79%). Cross-lingual confusions may be classified into two categories:

(I) The currency was correct, but the language was wrong – most of the examples (except for Row 3) in Table 1 belong to this category. Such errors are common, possibly because the pronunciation in one language is often derived from the pronunciation in another language.
(II) Both the currency and language were wrong – an example is row 3 in Table 1, where the Thai Baht (in Cantonese) was mistaken for the British Pound in English. The two pronunciations are actually rather similar.

3.3 Remarks on Trilingual Recognition

We feel that the trilingual recogizer offers much flexibility to the user in terms of language selection, despite the degradation in performance due to the enlarged vocabulary size, when compared to the monolingual recognizers. In addition, cross-language confusion errors in which the identity of the currency remains the same may be harmless towards the task of retrieving exchange rates between two currencies. Our speech recognizers form the front end to receive the user's input, and the system responds by means of speech synthesis. In the following sections, we will describe our speech generation component.

4. Speech Generation

The outputs from speech recognition creates a simple semantic frame, which consists of a language identifier, and the recognition outputs for the pair of currencies. The semantic frame invokes the processes of real-time data retrieval (exchange rates), as well as speech generation. Our speech generation component performs the tasks of text generation (for both English and Chinese), as well as syllable-based concatenative synthesis for Putonghua and Cantonese. As for English, we have integrated with the FESTIVAL system [4], to which our generated text is sent directly.

Since we are performing synthesis for a restricted domain, our task complexity is lower when compared to synthesis for free-form running text. Hence we aim to produce synthesized speech with higher naturalness. The various processes in our speech generation component are described in the following subsections.

4.1 Text Generation

Our text generation procedure aims to present raw data in the form of a presentable sentence. The information provided include the date, time, currencies, bid/ask prices and some system messages. Numerals were handled with special care for each of the languages. For example, the text for the year '1999' was generated with English groupings as "nineteen ninety nine" and serially in Chinese as "一九九九". This is different from decimals such as '3.456', generated serially for both languages – "three point four five six" in English, and "三點四五六". This, in turn, is different from a price of '123', generated as "one hundred and twenty three" in English, and "一百二十三" in Chinese. In the last case, words need to be inserted to indicate the order of magnitude of the number.

4.2 Concatenative Synthesis

As mentioned previously, the generated English text was fed directly to FESTIVAL, and we did not develop concatenative synthesis for English. However, we did include SABLE [5] markings to provide better specification of the pronunciation of some vocabulary items in our domain.

Synthesis for Putonghua and Cantonese share the same Chinese text generation output.[2] Our approach for concatenative synthesis consist of the following steps:

(i) Each Chinese character in the text sentence is mapped to its appropriate *tonal syllable*, according to our Cantonese and Putonghua lexicons. Rules for *tone change* are then applied, e.g. if there is a series of syllables with the third tone in Putonghua, we change all the syllables to the second tone except for the last one (consider the case of '九九九').
(ii) For every tonal syllable in the sentence, we select the "appropriate" syllable wave file from a previously prepared wave bank. Details of the syllable selection process are provided in the next subsection. The wave files are then concatenated in order to form the synthesized sentence.

4.2.1 Coarticulation and the Use of Distinctive Features

In designing an algorithm for syllable-based concatenation, special attention should be paid to the effect of coarticulation in continuous speech. It is widely known that context dependence heavily affects the acoustic realization of a syllable. Consider the character 七 (i.e. the number '7'), which is pronounced as 'cat1' in Cantonese. In the context of "六七八" (i.e. the number sequence '678', pronounced as "luk6 cat1 baat3"), the syllable 'cat1' has a *left velar* context, and a *right labial* context. Due to coarticulation, speakers tend to assimilate the alveolar closure of the syllable 'cat1' with the right labial, resulting in the production of 'cab1' (e.g. 軋). Hence if we were to extract the syllable wave file for 七 from the spoken phrase "六七八", and use it to synthesize "八七六", the resulting waveform will sound like "八軋六", which sounds incorrect when compared to natural speech.

It is obvious that the contextual characteristics of a syllable are important for concatenation. However, if we were to consider both the left and the right syllable contexts simultaneously, we may need to store N^3 wave files for every syllable in our "wave bank", where N is the number of unique syllables in the language.[3] In order to minimize the size of our wave bank, we decided to consider *only* the place of articulation of the (optional) coda of the left syllable, and the (optional) onset of the right syllable. This facilitates more sharing and hence a smaller size for our wave bank. The left and right contexts are represented

[2] We are aware that wordings in Putonghua and Cantonese do differ. However, for the sake of simplicity, and within the constraints of our domain, we find that using the same Chinese text generation output suffices for both dialects of Chinese.
[3] N^3 is a rough upper bound. N is about 1800 for Cantonese, and about 1200 for Putonghua.

with a two-digit encoding which represents the left and right distinctive features [6]:

Right Context: labial, alveolar, velar, glide, lateral, palatal and neutral (for the aspirant and syllables without onsets)
Left Context: alveolar, velar, neutral(for syllables without codas)

We can see that the variations in the left context is much more constrained which is characteristic of Chinese syllables.

4.2.2 Wave Bank Preparation and Concatenative Resequencing

Our wave bank stores a large set of syllable wave files which are extracted from continuous speech. We have designed a series of recording prompts which fully covers the possible context variants of the syllables in our vocabularies (for Putonghua and Cantonese). We collected the recordings from two female speakers, one for each language. The recordings were subsequently segmented to yield syllable wave files. Segmentation was achieved by forced alignment, and postprocessed by hand, with reference to the spectrogram. We have a total of 2400 syllable segments in all.

For a given textual input (with transcribed syllables), our synthesis algorithm concatenates the syllable wave files sequentially from left to right. The *unit selection* process ensures that the syllable variant with matching left and right contexts is chosen. We also inserted short pauses in between phrases, and long pauses in between sentences. Both the unit selection process and the insertion of pauses were found to be important contributing factors towards naturalness in the synthesized outputs.

5. End-to-end Interaction

Our speech recognizers form the front end of our foreign exchange inquiry system, to receive the user's spoken input. The system subsequently retrieves the relevant data, and responds by speech generation. The system is fully integrated end-to-end. If both of the speaker's utterances (for the two currencies) are in the same language, the speech generation component will respond in the corresponding language. Alternatively, if the languages of the two utterances differ, the system simply selects one of the two languages to respond.

6. Summary and Future Work

This paper reports on our preliminary effort in developing a trilingual interface to support financial information inquiries. The three languages of interest are most commonly used in Hong Kong, namely, Cantonese, Putonghua and English. Based on the foreign exchange domain, we have presented preliminary speech recognition results of three monolingual speech recognizers, and the trilingual recognizer. Error analyses indicates that cross-language confusions generally occur for the same currency name, which is less detrimental from the perspective of task completion. We have also presented the design of our speech generation component. While English output is produced by the FESTIVAL system, synthesis of Cantonese and Putonghua are based on syllable concatenation. We have devised a procedure for concatenative resequencing, which captures the left and right articulatory contexts using distinctive features, to achieve enhanced naturalness in the synthesizer output. At present, we are continuing this work along the lines of telephone speech recognition, as well as natural language processing, to be incorporated into the system.

Acknowledgments

The first author wishes to thank her undergraduate project students (Group H and Group N, 1998-1999) for various implementational assistance. The authors also thank Prof. P. C. Ching and Mr. Wai-Kit Lo of the Department of Electronic Engineering for their support of this project.

References

[1] HTK Book – http://www.entropic.com/products/htk/HTKBook
[2] Linguistic Society of Hong Kong (香港語言學會), *Hong Kong Jyut Ping Character Table (粵語拼音字表)*, Linguistic Society of Hong Kong Press, 1997.
[3] Han yu pin yin, 中華新字典, 中華書局(香港)有限公司.
[4] P. Taylor, A. Black and R. Caley, "The architecture of the Festival speech synthesis system", in *Proceedings of the 3rd ESCA Workshop on Speech Synthesis*, pp.147-151.
[5] R. Sproat *et al*, "SABLE: a standard for TTS markup", in *Proceedings of 1998 International Conference on Spoken Language Processing*, Vol.5, pp.1719-1722, Sydney.
[6] K. Stevens, "The quantal nature of speech: Evidence from articulatory-acoustic data," *in Human Communication: A Unified View.* E. E. David and P. B. Denes eds. McGraw-Hill. New York. 1972.