Dialectal Chinese Speech Recognition: Final Report

†Richard Sproat, University of Illinois
(Thomas) Fang Zheng, Tsinghua University
Liang Gu, IBM
Jing Li, Tsinghua University
Yanli Zheng, University of Illinois
Yi Su, Johns Hopkins University
Haolang Zhou, Johns Hopkins University
Philip Bramsen, MIT
David Kirsch, Lehigh University
Izhak Shafran, Johns Hopkins University
Stavros Tsakalidis, Johns Hopkins University
Rebecca Starr, Stanford University
Dan Jurafsky, Stanford University

† (Corresponding author. Email: rws@uiuc.edu)

http://www.clsp.jhu.edu/ws2004/groups/ws04casr/

November 15, 2004

1 Introduction

As people become ever more mobile and national and global economies ever more integrated, an ever larger population finds themselves needing to communicate in a language that is not their own. For Automatic Speech Recognition (ASR) this vast number of speakers of multiple languages implies the need to deal with accented speech, and indeed adapting to foreign-accented speech is an important problem in current speech recognition research (Kumpf and King, 1996; Teixeira, Trancoso, and Serralheiro, 1996; Lincoln, Cox, and Ringland, 1998; Huang et al., 2000; Livescu and Glass, 2003; Chen et al., 2001; Teixeira et al., 2001; Schultz et al., 2002; Huang, Chen, and Chang, 2004; Wang, Schultz, and Waibel, 2003; Liu and Fung, 2003a; Liu and Fung, 2003b; Mayfield Tomokiyo and Waibel, 2001; He and Zhao, 2003; Mayfield Tomokiyo, 2000)

The extensive literature on foreign-accented speech has often focused on English spoken with a foreign accent. Recognition of accented English is a crucial problem since English is a world language, with a very large number of second-language speakers, whether as imigrants in English-speaking countries like the United States, Britian, or Australia, or as speakers who use English in the world market.

Our focus in this project is on a different problem: recognizing accented speech in Chinese. China is a huge country and speech recognition of Chinese is an socially, economically, and politically important goal. Furthermore, there is a single standard spoken language in China, called *Putonghua* ('common language') in Chinese. But Putonghua is spoken extremely differently in different parts of China. This is because Chinese is a language with many so-called dialects (方言 *fāngyán*), the major groups of which are Mandarin, Yue (including Cantonese), Min (including Fujianese, Taiwanese), Wu (including Shanghainese), Xiang (Sichuan) and Gan (Zhejiang) (Norman, 1988; Ramsey, 1989); see Figure 1. For political reasons these different forms of speech are called dialects, but from a linguistic point of view they are better thought of as separate languages, as different from each other as, say the Romance languages French, Spanish, Italian, Portuguese and Romanian. Within each 'dialect' region there are of course further regional variants, so that there are for example many dialects of Mandarin, of Yue, of Wu and so forth. The standard language, Putonghua, is a version of Mandarin, and indeed is often called *Mandarin* or *Standard Mandarin* in the speech recognition literature (for example in databases like *CallHome Mandarin*) but since the term Mandarin also denotes a 'dialect' region, we will exclusively use the term *Putonghua* to refer to this national standard language.

These days anyone from any of the 'dialect' regions who goes to school will learn Putonghua as part of their education. For speakers of Mandarin dialects this is similar to English speakers in various parts of the United States learning Standard American English. For speakers of other 'dialect' regions, such as the Wu region around Shanghai, the task is akin to a speaker of, say, Spanish going to school and learning, say, French. In essence, when a native speaker of Shanghainese learns Putonghua, they are learning a foreign language. But, again for political reasons, the situation is not viewed as a case of foreign language learning, but rather a case of people who natively speak a different 'dialect' learning the standard language.

Since China is so large, and the regional accents of Putonghua so varied and so numerous, the task of speech recognition of dialectally accented Putonghua is hugely important. Recognizers trained on speakers from Beijing, the capital, perform poorly when tested on speakers from other large cities like Guangzhou or Shanghai.

Our goal in this project is to study how to address this crucial problem of ASR on Putonghua spoken in dialect regions. We have chosen for the 6-week 2004 workshop to focus on one dialect region: Wu, which is the language that includes the city of Shanghai. We chose Wu because it has the largest number of speakers of any non-Mandarin Chinese language. One 1991 source estimates that there are 87 million native Wu speakers.

Figure 1: Map of Southern China with a boundary delimiting the Wu speaking region. From (Ramsey, 1989).

The theme of this project is thus *Wu Dialectal Chinese Speech Recognition* or in other words ASR aimed at recognizing speech in the Putonghua language when spoken by people who are native speakers of the Wu languages. We will focus mainly on native speakers of the particular Wu language known as Shanghainese, and spoken in Shanghai.

Methods for dealing with accented speech vary from simply collecting data in that accent and training a recognizer, to various ways of adapting recognizers trained on unaccented speech.

Our focus in this work is on adaptation. We collected a database of Shanghainese accented Putonghua, and then investigated a number of different methods for improving recognition on this task by adapting recognizers that had been trained on standard Putonghua.

One class of methods we investigated is acoustic adaptation. We studied both MLLR and MAP

adaptation, in which relatively small amounts of Shanghainese-accented Putonghua data are used to adapt acoustic models that were trained on standard Putonghua.

We also performed a number of experiments on lexicon adaptation, in which the standard Putonghua pronunciation dictionary was adapted in various ways to Shanghai-accented Putonghua.

Finally, we explored the idea that speakers have varying degrees of accent in their Putonghua. Previous work on accented Chinese speech recognition — e.g. (Huang et al., 2000; Chen et al., 2001; Huang, Chen, and Chang, 2004; Liu and Fung, 2003b) — has tended to treat speakers from a given 'dialect' region as a single class. However speakers clearly have differing degrees of accent, so a central goal of this project was to investigate the utility of detecting and utilizing degree of accent in ASR.

In summary, the goals of this project were as follows:

- To provide useful baselines and upper bounds, and to investigate a variety of techniques for modeling accented speech in ASR.
- To investigate both acoustic and lexical techniques for adapting standard Putonghua recognizers to accented Putonghua.
- To develop methods for detecting the degree of accentuation, and test whether modeling degree of accentuation is useful for improving ASR on accented speech.

2 Background

2.1 Accent Adaptation: Background and Previous Research

Many researchers have studied the problem of adaptation to an accented speaker. Most methods have focused on one of two areas: acoustic adaptation or lexicon adaptation.

2.1.1 Acoustic Adaptation to Accent

A wide variety of acoustic adaptation methods have been investigated. The simplest method of course, is simply to train on accented data. Even a small amount of accented training data seems to be sufficient. Wang, Schultz, and Waibel (2003) investigated German-accented English speakers in the VERBMOBIL (conversational meeting planning) task. They showed that training on 52 minutes of non-native data (German-accented English) was much better (WER=43.5%) than training on 34 hours of native English data from the exact same domain (WER=49.3%). The next natural approach is to pool accented acoustic training data with un-accented data and train acoustic models on the combination. Wang, Schultz, and Waibel (2003) showed that simply pooling 34 hours of native data with 52 minutes of non-native data dropped the error rate slightly to 42.3%. Combining the two with an interpolation weight in order to give more weight to the accented data can work even better; Wang, Schultz, and Waibel (2003) found that when an optimal (oracle) interpolation weight was chosen, the error rate dropped to 36.0%.

A similar method is to train models on native speech, and then run a few additional forward-backward iterations with non-native speech. Mayfield Tomokiyo and Waibel (2001) examined the

task of recognizing Japanese-accented English in the VERBMOBIL domain. She had native English speaker data in the VERBMOBIL domain. She then collected 3 hours of wideband English speech from native speakers of Japanese who had had 6-8 years of English study, had lived in English-speaking country for 6-12 months, and reported difficulty in making themselves understood. She showed that simply pooling well-trained native English VERBMOBIL models with 3 hours of Japanese-accented English dropped WER from approximately 63% to 53%. When instead of retraining, she tried training on native English and then and then doing 2 additional forward-backward iterations with the 3 hours of accented data, WER dropped to 48%.

The effect of training directly on accented speech is of course even more profound if more data is available, or if the native speech was not in the same domain as the accented speech. (Ikeno et al., 2003) found that WER on a Spanish-accented conversational English test set dropped almost in half, from 68.5% (when trained on out of domain (WSJ) text from native English speakers) to 39.2% (when trained on 20 hours of in-domain Spanish accented English).

In conclusion, training on non-native data, especially when mixed with in-domain native data, provides the most obvious gains in performance on accented data.

More sophisticated acoustic methods for accent involve applying standard speaker adaptation techniques like MLLR and MAP adaptation to accented speakers. Both methods have been applied by many researchers to accented speech with extremely good results.

In MLLR adaptation (Leggetter and WoodlandP, 1995), counts from an adaptation dataset are used to train a transformation which is applied to the mean vectors of the gaussian PDFs. The transformation matrices are trained via EM to maximize the likelihood of the adaptation data. There can be a single matrix, or multiple transforms can be built, perhaps one for each context-independent phone.

The simplest use of adaptation was merely the direct use of MLLR to adapt individually to each test speaker. In an extremely similar task to ours, (recognizing Shanghainese-accented Putonghua) Huang et al. (2000) applied MLLR adaptation to a Microsoft Whisper system that had been trained on 100,000 sentences from 500 speakers from the Beijing area. Their test set was 10 male speakers from the Shanghai area. Their 187 phones were classified into 65 regression classes, and the MLLR transformation ioncluded both diagonal matrix and bias offsets. They applied MLLR adaptation indidivually to each speaker, using from 10 to 180 sentences from each speaker. On average, use of MLLR dropped the WER from a baseline of 23.18% to 21.48% after seeing 10 sentences (a decrease of 1.7% absolute) and to 15.50% after seeing 180 sentences (a decrease of 7.68% absolute).

A more complex use of MLLR was to adapt not just to the single accented test speaker, but to a larger number of accented speakers. Mayfield Tomokiyo and Waibel (2001) studied Japanese-accented English in the VERBMOBIL domain as discussed above. They used 50 sentences from test-speakers to do MLLR adaptation. Their baseline was to train on native English speakers, but use MLLR on individual test speakers for a WER of 63%. Using MLLR to adapt to 150 sentences from 3 speakers decreased WER to 58%, Using MLLR to adapt to 750 sentences (50 each from 15 speakers) decrease WER to 53% (unfortunately all WERs are approximate since they were taken from charts).

This use of MLLR on pooled groups of speakers was taken up for a German-accented English task by Wang, Schultz, and Waibel (2003). As discussed above, they studied German-accented English speakers in the VERBMOBIL (conversational meeting planning) task. They pooled 64 speakers, and applied MLLR on various amounts of this data. They showed that us-

ing MLLR on 7 minutes of adaptation data from 64 speakers, brings a drop in WER of approximately 1.5% absolute (from approximately 49.5% to approximately 46.8%). Unlike the Japanese-accented baseline discussed above, their baseline system does not seem to have used MLLR, so some of this 1.5% might be equally achieved by MLLR on the test speaker. Using very large amounts of adaptation data (50 minutes) for MLLR, the WER decreased approximately 4.5% to approximately 44.0%. Both the Mayfield Tomokiyo and Waibel (2001) Japanese-accented English and the Wang, Schultz, and Waibel (2003) German-accented English study only used a single MLLR transformation matrix, while Huang et al. (2000) used 65 separate transforms. This may explain the relatively small improvement Mayfield Tomokiyo and Waibel (2001) and Wang, Schultz, and Waibel (2003) achieved, even when using much more adaptation data, compared to Huang et al. (2000). Similarly, Goronzy, Sahakyan, and Wokure (2001) used only one MLLR regression class for German and Italian speakers of English and found that while performance improved on 3 out of 12 speakers in the test set, it declined for 9 out of 12 speakers! This suggests that training multiple transforms are quite important when doing adaptation to accented speech. But the comparision is not clear, because Mayfield Tomokiyo and Waibel (2001) used speaker-dependent MLLR as their baseline when evaluating MLLR adpatation to a class of accented speakers. Huang et al. (2000) only report the standard use of MLLR on individual test speakers. In essence, then the Huang et al. (2000) baseline was set artificially low, for some reason, turning off the MLLR that their standard system otherwise used.

Another widely used acoustic adaptation method that has been applied to accented speech is MAP adaptation. In MAP estimation, the estimate of the gaussian means for each model is formed by a weighted average of the training means and the adaptation data means. If there is insufficient adaptation data for a model, the training mean alone is used. On the German-accented English task described above, Wang, Schultz, and Waibel (2003) separately tested both MLLR and MAP adaptation. With large amounts of adapatation date (more than 20 minutes) they found MAP better at decreasing WER than MLLR. For example, using all 52 minutes of accented data from 64 speakers, MAP estimation reduced error rates from approximately 49.5% to approximately 38%, while MLLR only reduced WER to 44%. They did not attempt to combine MAP and MLLR.

Finally a number of researchers have attempted to apply other acoustic adaptation methods to accented speech. A number of researchers, for example, explore the use of L1, the native language of these accented speakers, as a source of acoustic training data (Liu and Fung, 2000; Mayfield Tomokiyo and Waibel, 2001; Wang, Schultz, and Waibel, 2003). (In our task, for example, L1 would be Shanghainese). Other researchers have explored other models, such as using the Polyphone Decision Tree Specialization (PDTS) method of modifying the decision tree used to cluster context-dependent phone models (Wang, Schultz, and Waibel, 2003).

In summary, previous research suggests that MLLR can be used on groups of speakers to help adapt acoustic models to foreign accent. Previous applications of MLLR in this multi-speaker adaptation environment, however, have been limited to a single global transform. Previous research has shown, not surprisingly, that MAP performs better than MLLR with enough training data. But previous research has not shown whether combining MAP and MLLR could be useful for adaptation specifically to accented data. This state of the art suggests a few goals for the acoustic adaptation portion of our work. First, we need to explore the usefulness of more complex uses of MLLR, involving more specific transforms rather than a single global transform. Second, we need to explore combinations of MLLR and MAP as applied to the accent problem.

2.1.2 Lexicon Adaptation to Accent

In addition to acoustic adapatation, many researchers have studied the role of lexicon adapatation to deal with accented speakers. By lexicon adaptation, we mean modification of the pronunciation dictionary to have accent-specific pronunciations. For speech recognition in general, lexicon adaptation has shown much smaller gains than acoustic adaptation. Some researchers, however, have argued that accented speech is a particularly appropriate task for pronunciation modeling to help (Goronzy, Sahakyan, and Wokure, 2001).

In general, research on lexicon adaptation has focused on augmenting or modifying the base lexicon to have accent-appropriate pronunciations. In what might be called the 'standard approach' to pronunciation modeling, pronunciation variants are created to add to the dictionary. These pronunciations may be written by hand, extracted by hand from phonetically transcribed corpora, extracting automatically from corpora for example by running a phone recognizer, or generated by rule from previous pronunciations. If a training corpus is available, the resulting expanded lexicon is generally then force-aligned with the training set to learn pronunciation probabilities. Finally, the pronunciations are pruned in some way to result in a relatively small number of pronunciations per word (usually on the order of 1.4).

Probably the most successful class of methods for building an augmented accented lexicon has been to use accented training data to learn a set of rules, mapping, or phone confusions which are then used to augment the dictionary.

(Humphries, Woodland, and Pearce, 1997) was one of the earliest applications of lexicon adaptation to accent modeling, specifically adapting a standard (London and South East England) lexicon to improve recognition of Lancashire and Yorkshire accented English. Their goal was to build phone-confusion decision trees, which would expand the standard lexicon with accented pronunciation variants. They first generated a phonetic transcription of the accented data, by taking the canonical lexicon, and building an augmented lexicon in which each vowel could to be replaced by any other vowel. (On the assumption that most accent pronunciation variation takes place only in the vowel, and the simplifying assumption that only phone substitutions, and not insertions and deletions, would occur. They then force-aligned the training set using this expanded lexicon, producing a phonetic transcription. This transcription of the accented data was then aligned with a canonical dictionary transcription. The resulting phone alignment was then used to train a (pruned) decision tree to map unaccented phones into accented phones. This decision tree was then applied to the canonical dictionary to produce additional accented pronunciations for each word. The resulting dictionary, with the optimal decision tree pruning threshold, reduced WER from 17.2% to 13.9% on the accented test data.

(Livescu and Glass, 2003) explored the use of lexicon augmentation for the JUPITER system. In their system, there were many different accented speakers, but there were not enough of each type to train accent-specific lexicons. They therefore built a single 'non-native-speaker' lexicon. Like (Humphries, Woodland, and Pearce, 1997), their idea was to induce native-to-accented phone mappings from a training set, and use these mappings to expand the native lexicon with alternative accented pronunciations. Their method was first to run a phone recognizer on the mixed accented training data, constrained by a phone bigram language model. They then aligned this phone transcription with the canonical (native) pronunciations from a native dictionary, and used this aligned corpus to learn a set of phone confusions. They then pruned this confusion matrix, and used the

pruned phone confusions to expand each pronunciation in the canonical dictionary. They then took this augmented canonical dictionary, force-aligned it to the training set, took the single best best pronunciation path for each training sentence, and build a smaller, more constrainted phone confusion matrix. This phone confusion matrix achieved The baseline system had achieved WER of 20.9% on non-native data and WER of 10.5% on native data; using the phone confusion matrix reduced the WER on non-native data to 18.8%.

(Huang et al., 2000) applied a very similar method to the problem of Shanghainese-accented Mandarin. They ran a syllable-recognizer on an accented training set, producing an accented syllable labeling. They then aligned these accented labels with the canonical (standard Mandarin) labels, to produce a set of accented-to-standard syllable transformation pairs. They used 37 such transformation pairs to augment their standard (native) Putonghua dictionary with new pronunciations. In addition, they trained transformation probabilities for each of these transforms to generate a probability for each of these new pronunciations. The resulting expanded lexicon reduced syllable error rate from 23.18% to 19.96%. The lexicon was not as useful when combined with MLLR, however; with a baseline system using 10 sentences of MLLR, the lexicon only reduced syllable error rate from 21.48% to 21.12%.

(Mayfield, 2002) explored the problem of Japanese-accented English that she had already begun to address in (Mayfield Tomokiyo, 2000) and (Mayfield Tomokiyo and Waibel, 2001). Her genre in this case was read speech; she created her database by choosing Japanese speakers with low proficiency in English and having them read aloud from the Children's News Database. In (Mayfield, 2002) she attempted a number of ways to augment the lexicon with Japanese-accented pronunciations. In the first class of methods, she wrote linguistic rules, used them to expand her lexicon, producing a large list of 915,672 pronunciation variants. She then aligned this expanded lexicon to an accented training set. From the aligned data, she either selected frequent pronunciations to directly augment the lexicon, or frequent phone-transforms to apply to the lexicon to create new pronunciations; each of these ways of using the forced alignments was pruned in various ways. In the second class of methods, she used a phone recognizer on the training data to bootstrap the process rather than hand-written rules. She aligned the phone recognizer output with the dictionary output to learn context-independent phone transforms, and applied them, with various prunings, to the lexicon. (Mayfield, 2002) found that none of these methods improved recognition performance. She hypothesized that read speech might be a bad database to use for lexicon adaptation experiments, because speakers often didn't know the words, and so stumbled over them rather than reading them with a standard phonological accent.

(Goronzy, Sahakyan, and Wokure, 2001) looked at German and Italian speakers of English in the ISLE corpus. They phonetically transcribed part of the speech, and used the aligned phonetic transcriptions in two ways. In the first experiment, they added frequently occuring pronunciations to the lexicon. This showed very little improvement in WER. In the second experiment, they used the corpus as a tool for hand-writing phonological rules, rules such as word-final devoicing, /h/-deletion, schwa-epenthesis, and various other phone substitutions. These rules were then used to expand the native-English dictionary. They found that the expanded dictionary actually increased the WER! In a final experiment, however, they instead built oracle-based speaker-specific lexicons. For these, they looked at the test set for each speaker, and from the large set of phonetic rules discussed above just those rules that decreased WER for the speaker. They then applied these rules to build separate speaker-specific dictionaries for each speaker. The resulting oracle dictionaries gave a few percent

absolute WER improvement on each speaker, even after MLLR adaptation.

In summary, lexicon adaptation does appear to offer some help in reducing WER on accented speech, although the reductions are modest and sometimes go away when enough data is available for acoustic adaptation via MLLR. Thus lexicon adaptation may be particularly applicable in situations where speakers are only uttering a single sentence, and acoustic adaptation is not possible. In addition, the work of (Goronzy, Sahakyan, and Wokure, 2001) suggests that speaker-specific pronunciation modeling, when possible, may be a useful tool.

2.1.3 Accent Detection

Various features have been used to build accent detectors. Chen et al. (2001) built an detector which disinguished between four different Chinese accents in Putonghua: native speakers from Shanghai, Guangdong, Beijing, and Taiwan. They trained GMMs for each accent, one for females, one for males. They used standard MFCC features, and 32 component Gaussians. They achieved error rates of 11.7 for females and 15.5 for males. They noted that a weakness of the GMM method is its sensitivity to channel characteristics.

Teixeira et al. (2001) built decision trees to detect the level of accent of Japanese-accented speakers of English. Their features included basic features such as rate of speech, as well as prosodic features like the duration of the primary stressed vowel, durations of words, durations of intra-sentence pauses, maximum F0 excursion, and various normalizations of these features. They assumed correct gender was given. They achieved an error rate of 14.5% compared to human labelers of the accent levels.

(Schultz et al., 2002) suggest that the GMM approach to accent recognition may be too dependent on matched acoustic conditions between training and test. Since in a real environment such conditions are unlikely to match, they propose a new approach for accent detection based on phone strings. Their approach is based on using phonotactic information; phone transition probabilities generated from phone recognizers. The idea is that native speakers will have different phone n-gram probabilities than accented speakers. In order to make the phone n-gram probabilities more fine-grained, they actually use phone recognizers from 6 different languages. The procedure is as follows. First, they created training and test sets of Japanese-accented English and native English. Next, the 6 phone recognizers were run on the 2 training sets, generating 6 sets of phone output and 6 sets of phone N-gram grammars, one set for native English and one set for Japanese-accented English. In decoding, each test sentence is run through the two sets of 6 phone recognizers, generating 6 perplexities. These are interpolated to produce a perplexity for native English and a perplexity for Japanese-accented English; the lower perplexity model is chosen as the accent detected for the test speaker. They achieved 93.7% accuracy at the accent identification task. They also attempted to use this method to label the strength of accent, by labeling each Japanese-accented speaker for their proficiency in English. The phonotactic approach fared worse at this more difficult task of distinguishing between 3 different classes of accents, accuracy on 3-way classification ranging form 34% to 59%.

In summary, previous approaches to accent detection have employed GMMs trained on acoustics, or phonotactic probabilities based on phone recognition. We investigated both of these methods on our Shanghainese-accented Putonghua task.

	Labials	Apicals	Apical Sibilants	Retroflex	Pal.	Vel.
Unasp.	p	t	ts	tʂ	tɕ	k
Asp.	ph	th	tsh	tʂh	tɕh	kh
Nas.	m	n				ŋ
Fric.	f		s	ʂ	ɕ	x
Son.	w	l		ɹ	j	

Table 1: Putonghua Initials, after (Norman, 1988).

ʐ/ɻ	ɤʌ	A	ei	ae	oɷ	ɑo	ən	an	ʌŋ	ɑŋ	ər
i	iɛ	iA			ioɷ	iɑo	in	iɛn	iŋ	iɑŋ	
u	uo	uA	ueɪ	uae			uən	uan	ɷŋ	uɑŋ	
y	yɛ						yɪn	yan	yɷn		

Table 2: Putonghua Finals, after (Norman, 1988).

2.2 Shanghainese and Wu-accented Putonghua

In this section we summarize some of the phonetic and phonological characteristics of Shanghainese and Wu-accented Putonghua. Shanghainese is quite different from Mandarin, particularly in the area of phonology. These differences presumably affect the speech of Shanghainese speakers speaking Putonghua. Key phonological differences are the following. Mandarin, and hence Putonghua, is typical of many Chinese languages in only having two kinds of stops, namely voiceless unaspirated and voiceless aspirated. In contrast Wu dialects are characterized by a three-way contrast, namely unaspirated voiceless, aspirated voiceless and voiced. Mandarin has lost all syllable final consonants except /n/ and /ŋ/, and Wu dialects similarly have lost most final consonants, except /ʔ/, a residue of historical final stops, and final /ŋ/. The two final consonants are restricted in their distribution, only occurring after /i/, /ə/, /a/, /u/, /o/ and /ɔ/ (Norman, 1988, page 201).

Wu dialects have a much wider range of simple vowels than Mandarin, including low front rounded vowels. On the other hand, Mandarin dialects have diphthongs, which many Wu dialects, including Shanghainese, lack. See Tables 1–2 and 3–4 for charts of the segments of Mandarin and Shanghai.

A Wu accent in Putonghua is characterized by two factors. The first is the underlying Wu language, which in the heaviest accented speakers can have a strong influence on the pronunciation of Putonghua: such speakers may substitute Wu vowels for Putonghua ones, use voiced consonants, and produce syllables ending in /ʔ/. The second is the general properties of Southern Mandarin, characterized by a bevy of sound changes including the substitution of retroflex fricatives and affricates (Pinyin *sh*, *ch*, *zh*) with the corresponding non-retroflex sounds (*s*, *c*, *z*), and the neutralization (merger) of final *n* and *ng* to *ng* following *i*. This is characteristic of speakers of Mandarin/Putonghua throughout the South including Taiwan. But, again, these characteristics are not categorical and speakers may exhibit them to greater or lesser extent.

In addition to these Wu accented issues, of course, our corpus exhibits examples of spontaneous or fast speech phenomena that exist in the speech of people from any region. In Putonghua, as in other languages, these include deletions of segments, such as the initials of syllables in prosodically weak environments. There is a common misconception that Chinese syllables are somehow more

	Lab.	Dent.	Alv. Sib.	Pal.	Vel.	Glot.
Unasp. Stops	p	t	ts	tɕ	k	
Asp. Stops	ph	th	tsh	tɕh	kh	
Vcd. Stops	b	d		dʑ	g	
UnVcd. Fric	f		s	ɕ		h
Vcd. Fric	v		z	ʑ		ɦ
Nas.	m	n		ȵ	ŋ	
Liq.	w	l		j		

Table 3: Shanghai Initials, after (Norman, 1988).

	Front	Central	Back
High	i y		u
Upper Mid	e ø		o
Lower Mid	ɛ	ə ɵ	ɔ
Low		a	

Table 4: Shanghai Vowels, after (Norman, 1988).

"robust" than syllables in languages like English. Thus, Huang et al. (2000) have stated that "deletion errors are infrequent in Mandarin because of the strict syllable structure." This is false however: a careful study of even a small corpus such as that reported in (Sproat and Shih, 2001) reveals that deletions are quite common in spontaneous speech. Thus in that corpus, out of 708 lexical segments 113, or 16% are changed, and of these 46 (40%) involve deletions. Similar percentages of deletions are found in our Wu speaker data.

Naturally, for any system to work well on spontaneous speech of any kind, it must be able to deal with any kind of phonetic change, both the kind due to the Wu regional accent as well as these 'fast speech' phenomena. Since this project specifically addresses in the problem of accented speech, however, our attention will be focussed only on the regional changes.

3 The Wu-accented Putonghua Corpus: Data Collection and Transcription

The Wu-accented Putonghua Corpus was designed to give a good cross-section of accented Putonghua spoken by native speakers of Shanghainese (or in a few cases, speakers of other Wu languaages). Data were collected from 100 speakers, 50 male and 50 female, in four locations in Shanghai. The four locations were a junior high school, a government bureau, a factory, and a postal research academy. In each location, a local coordinator was chosen who selected the 25 speakers from that location. Coordinators were given constraints in selecting subjects, so as to achieve a balance of age, education level, and Putonghua skills. (See Appendix A for details on the 100 speakers.)

Both spontaneous speech and read speech was collected from each speaker. The read speech comprised about 3 minutes for each speaker, and speakers read from text prepared by the experimenters. There were 65 different sentences used for the read speech, with each speaker reading

Num of speakers		Male	Female	Total
Age	26-40	27	25	52
	41-50	23	25	48

Table 5: Age distribution of speakers.

Num of speakers		Male	Female	Total
Education	High	41	41	82
	Low	9	9	18

Table 6: Education levels of speakers.

different sentences. A few commonly used Shanghainese words were inserted into the read sentences, so as to have some examples of the pronunciation of such code-switched words. A sample of the read sentences:

泰国国家安全委员会秘书长乍兰说

"Thailand Security Council secretary Zhang Zhalan said"

Spontaneous speech consisted of free form monologues where the speaker was asked to discuss anything they wished in one of the following areas: sports, policy/economy, entertainment, lifestyles or technology. Monologues were recorded in a room with an experimenter. All in all there were two experimenters, neither from Shanghai; one was co-author Li Jing. Occasionally the experimenter would prompt the speaker with questions. The total amount of spontaneous speech for each speaker averaged about 5 minutes. Speech was recorded using two similar head-mounted microphones: SONY OEM headset microphone; ANDREA Anti-Noise NC-61. The experimenters sat in the same room as the speakers, but sat a couple meters away from the microphone.

华寺龙华寺怎么带来的对吧

"Hua Temple — Longhua Temple, how did it come about, right?"

Speakers were classified by experts at the Chinese Academy of Social Sciences (中国社会科学院 CASS) into "Putonghua Level" (PTH Level) and fluency in Putonghua. Putonghua level ranges from 1 (best) to 3 (worst), with subdivisions A (better) and B (worse); all of our speakers fall in the range 2A-3B. Speakers were assigned fluency on a two point scale, i.e. fluent or not fluent.

In sum, the corpus contains 8.3 hours of spontaneous speech and 5 hours of read speech.

Spontaneous speech data was transcribed into 3 main tiers and a fourth miscellaneous tier. The Orthographic tier contains an orthographic transcription of the entire conversation in Chinese characters ('hanzi'). In addition to these normal orthographic conventions of Chinese orthography, 25 special markers were used in this tier to indicate non-lexical phenomena, including breaths, laughter, silence, lip smacks, beeps, noises, and so on.

The Pinyin tier contained the canonical dictionary pronunciation of each word in the conversation, using the Pinyin alphabet and drawn from a standard dictionary. Finally, there was a Phonetic surface form tier, containing a phonetic transcription of the entire conversation. This transcription

was done by phoneticians at the Institute of Linguistics at the Chinese Academy of Social Sciences, led by Aijun Li and Xiaoxia Chen. This phonetic transcription was hand-aligned to the speech waveform using the Praat editor.

Phonetic transcriptions were done in terms of "initial-final" (IF) units, where initials are the initial obstruents of syllables, and finals are the remainder of the syllable, consisting of an optional on-glide, the vowel nucleus, and an optional (nasal) coda. To illustrate this, the syllable *guang* would consist of an initial *g* and final *uang*; for *dou* the initial would be *d* and the final the vowel *ou*; finally for *wang* there would be no initial, just the final *wang* (identical to the final *uang* in *guang*). While it would be perfectly possible to transcribe Putonghua in terms of segments, as is typically done in English, most Chinese speech recognizers, including the recognizers we used in our experiments, are based on IF units. From the point of view of acoustic modeling there may in any case be some advantage in doing this since the longer finals incorporate more context than segmental units would.

The transcription alphabet was created by combining two separate IF sets. One was a set of 61 IFs used to represent standard Putonghua.[1] In addition to these 61 IFs, a large number of other IFs were added to capture the Shanghainese influence on the phonetics. These IFs were drawn from the set of IFs used for the Shanghai Dialect.

In addition, a number of extra diacritics were used to mark more detailed aspects the phonetic transcription, including markers for syllable-final retroflexion ("_r"), voicing due to coarticulation ("_v"), and 3rd tone sandhi. Details on the phonetic transcriptions are given in Appendix B.

It was decided during the later preworkshop preparations that we would focus primarily on spontaneous speech so the bulk of the remaining discussion deals with spontaneous speech only. However, the read speech data has been used in some of the initial linguistic analysis as well as in training models for age and PTH level detection.

4 Preliminary Linguistic Analysis of the Data

Before designing our ASR experiments, we began by conducting a linguistic analysis of the accented speech data. This analysis was conducted by Rebecca Starr and Dan Jurafsky at Stanford.

We first examined the phonological characteristics of the Shanghainese-accented Putonghua data. Shanghainese-accented Putonghua (as a Southern dialect of Mandarin) is well-known to have a number of phonological idiosyncracies. The most well-known is the fronting of retroflex fricatives and affricates, resulting in the following three phonological changes:

- sh → s
- ch → c
- zh → z

Table 7 shows these common pronunciation phenomena.

In addition, Shanghainese Putonghua shows changes in the final nasal consonants. Final [in] and [ing] are merged or in some sort of variation, and final [eng] is pronounced [en].

[1] 60 of these 61 IFs are used also in the Mandarin (Standard Putonghua) recognizers used at Hopkins that we used as our recognition testbed, to be described below. These were the 60 context-independent IFs on which context-dependent models are built.

Standard PTH	Shanghai Accent	
shan (山)	san	'mountain'
chan (蝉)	can	'cicada'
zhuozi (卓子)	zuozi	'table'

Table 7: Southern accent: fronting of [sh]/[ch]/[zh] to s/c/z

- eng → en
- in ↔ ing

Because the nasal vowel differences were subtle and difficult to code accurately, we chose to investigate the three cases of fronted retroflexes, [sh], [zh], and [ch]. We examined every case of [sh], [zh], [ch], and [s], [z], and [c] in the entire corpus. All in all, there were 19,662 tokens of sh/zh/ch. We performed a number of analyses on sh/zh/ch, using the results as a window into Shanghainese accent in general. We discuss the most relevant of the analyses here.

First, we coded each observation of sh/zh/ch/s/z/c for a number of features:

- Did they turn into s/z/c?
- age of speaker
- sex of speaker
- education level of speaker
- phone identity (sh, zh, or ch?)
- phonetic context.

We then performed a series of logistic regression experiments to determine which of these factors affected the fronting of [sh]/[zh]/[ch] to [s]/[z][c].

A first general result of our analysis is that there is massive variation between speakers in their use of standard PTH sh/zh/ch versus Shanghainese-accented s/z/c. Different speakers ranged from 0 to 100% use of the standard phones, as shown in Figure 2.

The massive variation in fronting confirms that accent is a continuous phenomenon, with different accented speakers showing different degrees of change. Even within this single type of sound change (fronting), some speakers showg the change consistently, others never show it, and still others show it to intermediate degrees.

In general, we found that each of the variable we examined played a significant role in predicting the amount of fronting (s/z/c) that occurred. One of the strongest variables was age. Figure 3 plots the degree of fronting (the percentage of cases where, say *sh* is changed into *s*) against age. Older speakers tend to have higher degrees of fronting, i.e., they tend to have a stronger accent. (We also observe in this plot that speakers tend to show somewhat less fronting in reading, which is not surprising given that reading aloud is a normative task.)

We found similar results for education; the more educated a speaker was, the more standard was his/her Putonghua. Or, to rephrase, less educated speakers had a stronger Shanghainese accent. Details are shown in Figure 4.

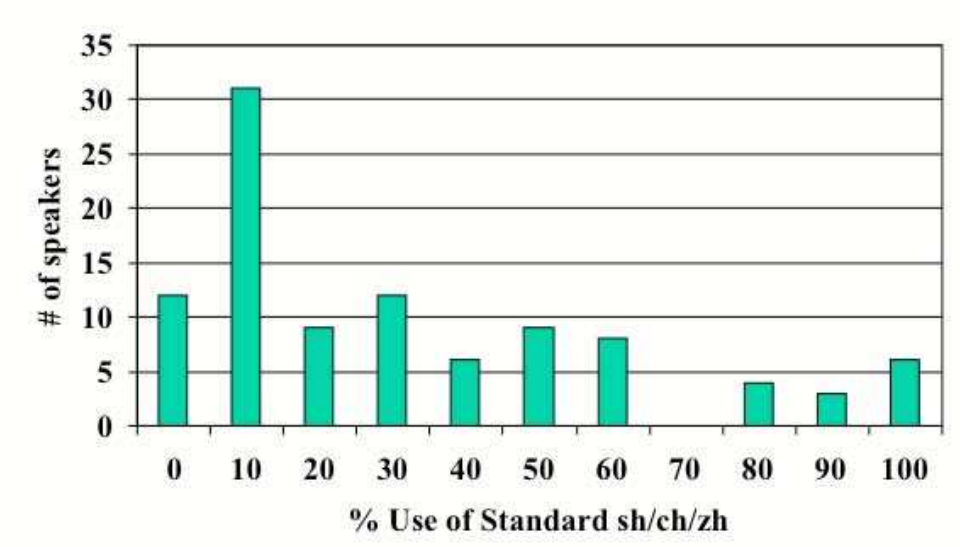

Figure 2: Variation among speakers in their fronting of *sh*, *ch*, *zh* to *s*, *c*, and *z*. The x-axis indicates bins of percentage use of the (northern) standard PTH sh/ch/zh, while the y-axis indicates the number of speakers in each bin.

Women spoke slightly more standardly than man. Details are shown in Figure 5.

We found very little effect of phonetic context on fronting (Southern accent). Only one of the many finals (rimes) in Chinese showed any affect, which was the vowel [an]. Initial sh/ch/zh was more fronted (more Southern) when followed by the final [an]. This is likely to be because [an] is the only front final that can follow sh/ch/zh. Table /reftab:phoneeffect gives an example of more fronting before [an].

	%s		%sh	
shang	725	.67	314	.33
shan	155	.75	51	.25

Table 8: [sh] more likely to become [s] before [an] than [ang].

We also found different values of fronting (Southern accent) for the different phones [sh]/[ch]/[zh] themselves. [sh] was more likely to be fronted (Southern) than [zh] or [ch] Details are shown in Figure 6.

We noted a wide variety of hypercorrections in our data. A hypercorrection is a kind of over-application by a speaker of a rule. The fact that southern-accented speakers use [s/c/z] for standard Putonghua [sh/ch/zh] is known by many Southern speakers. Speakers who know that Southern [s] corresponds to standard PTH [sh], in an attempt to simulate the standard pronunciation, will pronounce [s] as [sh] even in words that have [s] (not [sh]) in standard PTH. In the following sentence, for example, the standard PTH word *su4du*, ‘speed’, is incorrectly pronounced *shu4du4* by a speaker who is hypercorrecting.

速度都是很快的

shu4du4 dou1 shi4 hen3 kuai4 de0

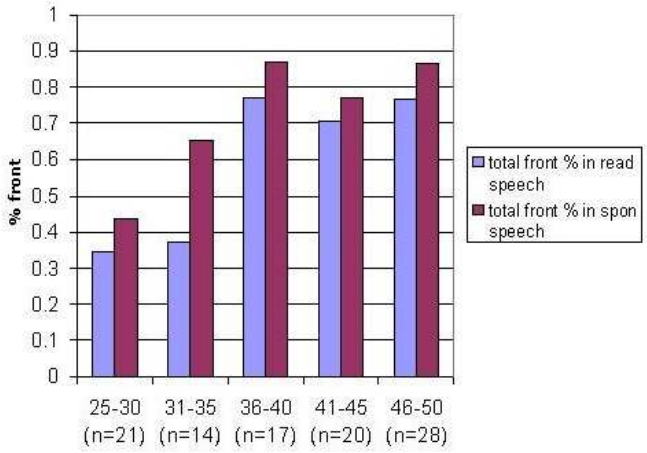

Figure 3: Total fronting of *sh*, *ch*, *zh* as a function of age. Older speakers show more fronting (Southern accent) than younger speakers.

The speed was still quite fast.

We measured hypercorrection in two ways. From a production perspective, we measured the percentage of standard [s], [z], and [c] that are produced as [sh], [zh], and [ch]. From a recognition perspective, we measured the percentage of the [sh], [zh], and [ch] that are produced by the speaker that would be [s], [z], and [c] in standard Putonghua. The example above has 100% production hypercorrection, and 50% recognition hypercorrection. We measured hypercorrection only in the read speech, since a preliminary analysis found that hypercorrection was much more common in read speech than spontaneous speech. Note that this is the opposite of fronting, which occured more often in spotaneous speech than read speech.

Figure 7 shows the percentage of hypercorrection from the production perspective. Figure 8 shows the percentage of hypercorrection from the recognition perspective.

Finally, we looked at the relation between fronting and the hand-labeled PTH fluency score. Figure 9 shows the three fronting rules, in read versus spontaeous speech. For each of the 3 rules, we show the relation between percentage fronting and PTH fluency score. The percentage of [sh] which become [s] in read speech turns out to correlate the most strongly with PTH fluency.

In summary, we reached four conclusions from our analysis. First, there was massive variation between speakers in the amount of accent severity. This suggests that accent modeling needs to be continuous rather than binary, and we will need to model the graded nature of accent severity. Second, age and education are good predictors of standard speech. This suggests that we can use age-type features to predict accent severity. Third, we found that the more speakers used [s] for [sh], the more accented they were. That suggests that the relative counts of [s] versus [sh] might provide us with another measure of accent severity. Finally, Shanghainese accented PTH does have clear phonological transformations from standard PTH. This suggests that traditional pronunciation modeling is worth investigating in this domain.

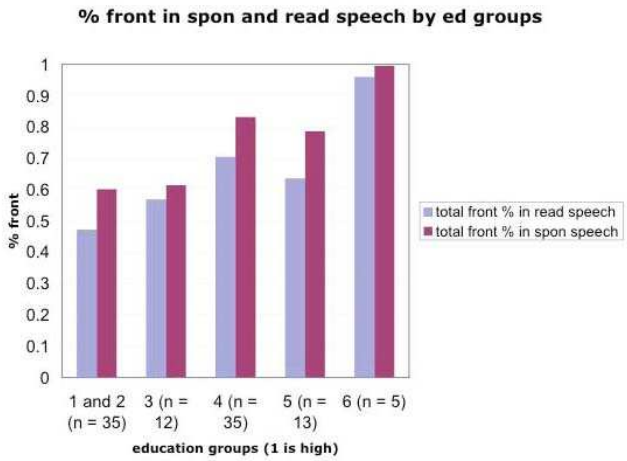

Figure 4: Total fronting of *sh*, *ch*, *zh* as a function of education. Less educated speakers show more fronting (Southern accent) than more educated speakers.

5 Data Division

We selected a total of 20 speakers (10 male, 10 female) for the test speakers, comprising a total of 1.7 hours of speech. The remaining 80 speakers (6.3 hours) were used for development data. The test speakers were selected with the goal of having an evenly balanced set of strongly accented and more standard speakers. The more strongly accented speakers have a PTH level designation of 3, whereas the more standard speakers have a PTH level designation of 2. The status of the test speakers as either more accented or more standard was verified by listening by native speaker project team members. The exact breakdown of the 20 test speakers into their CASS PTH level designations is as follows:

# Speakers	CASS Designation	Our Classification
7	2B	more standard
3	2A	more standard
6	3B	more accented
4	3A	more accented

Further details of the test speakers are given in Table 9. Details of all 100 speakers are given in Appendix A.

6 Baseline system

A word bigram language model is used throughout all the experiments in this paper. The test and training corpora were segmented using a maximum matching algorithm using a fixed dictionary

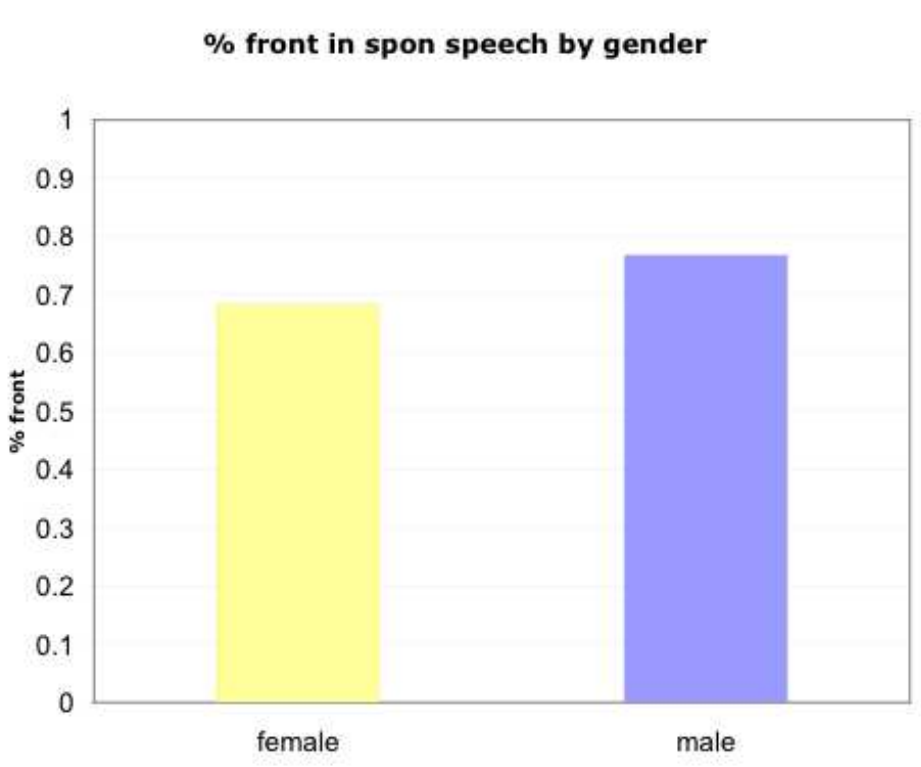

Figure 5: Total fronting of *sh*, *ch*, *zh* as a function of gender. Men show slightly more fronting (Southern accent) than women.

consisting of 50,647 entries developed at Tsinghua University. Language model training corpora consisted of the following conversational Putonghua data with 1.22 million characters:

- Mandarin HUB5 (200 telephone conversations of up to 30 minutes each)
- 100 hours of conversational Putonghua speech collected by Hong Kong University of Science and Technology.
- The transcriptions from the 6.3 hours of training data from our Wu-accented speech corpus.

Standard MFCC-based acoustic models with 14 mixtures per state were constructed using HTK version 3.2 (Young et al., 2002). Two baseline acoustic model training sets were used:

- MBN: 1997 Mandarin Broadcast News corpus (Hub-4NE), consisting of 30 hours of speech from mostly trained speakers.
- WU: 6.3 hours of Wu-accented training data.

The MBN data was chosen since it matches our data in one respect, namely that it is wideband recording. Table 25 shows the baseline results for the two acoustic models; here and elsewhere, we report results in terms of *Character Error Rate* (CER), which is the standard measure of performance in Chinese speech recognition.

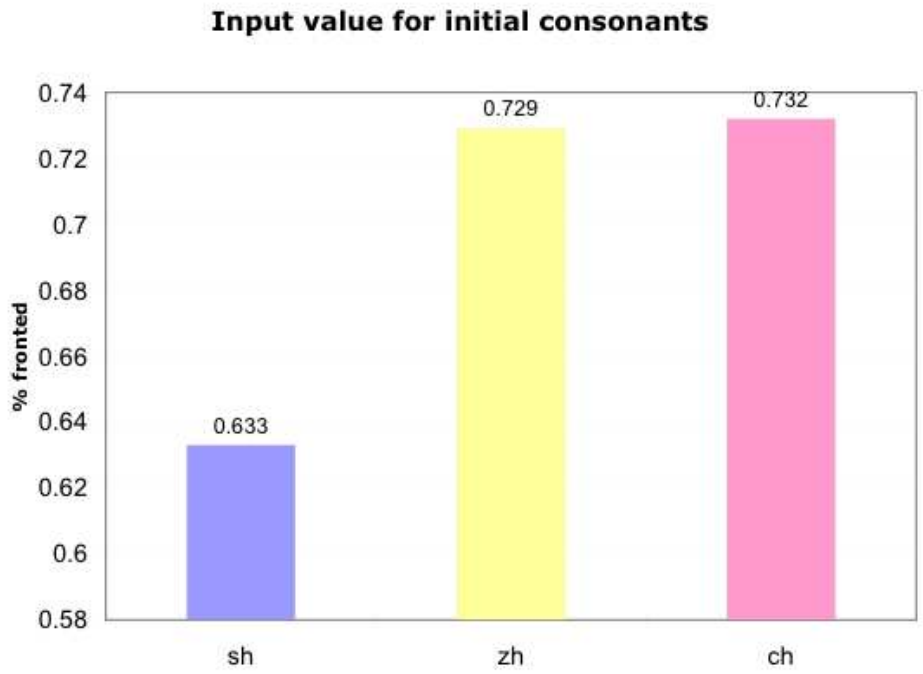

Figure 6: Total fronting of *sh*, *ch*, *zh*. *sh* has less fronting.

6.1 Language Modeling

Language modeling was not a major focus of the work in this project since it was clear from the outset that there was little grammatical difference between the speech of the Wu-accented speakers and what one would expect to find among more standard speakers. One might have imagined that speakers would use Shanghainese grammatical constructions or Shanghainese-influenced lexical items (such as *huānxǐ* for standard *xǐhuān* 'like'), but in fact there were few convincing cases.

We used the AT&T FSM ((Mohri, Pereira, and Riley, 1998), `http://www.research.att.com/sw/tools/fsm`) and GRM ((Mohri, 2001), `http://www.research.att.com/sw/tools/grm`) toolkit to construct a word bigram language model using Katz discounting. The test and training corpora were segmented using a maximum matching algorithm using a fixed dictionary consisting of 50,647 entries developed at Tsinghua University. Training corpora consisted of the following conversational Putonghua data:

- Mandarin HUB5 `http://wave.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC98S69` (200 telephone conversations of up to 30 minutes each)
- 100 hours of conversational Putonghua speech collected by Hong Kong University of Science and Technology.
- The transcriptions from the 6.3 hours of training data from our Wu-accented speech corpus.

Due to implementational differences between different language modeling toolkits and what particular parameters each system supports we did some quick perplexity comparisons between

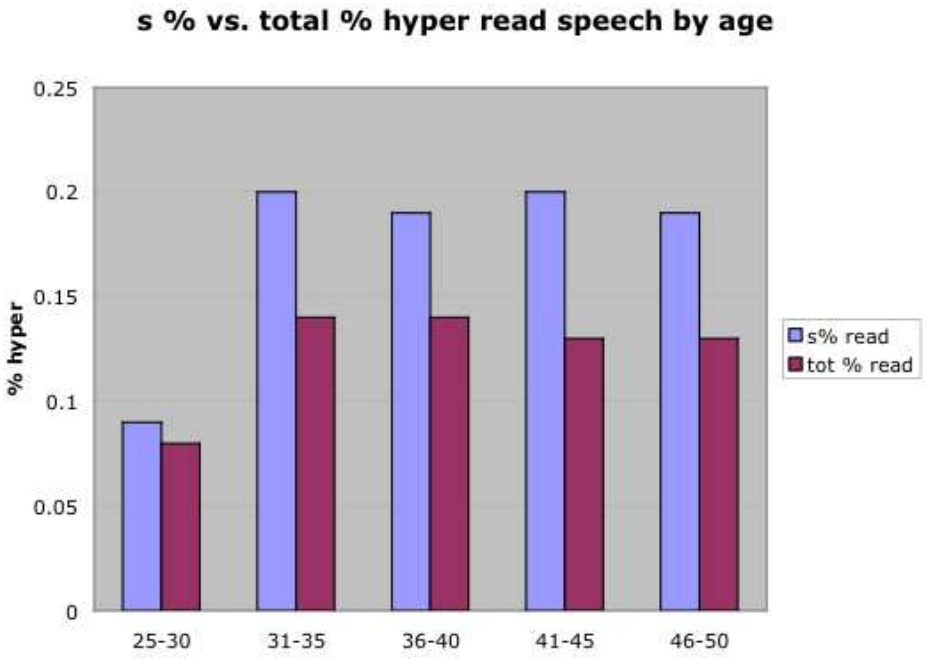

Figure 7: Hypercorrection in read speech: production perspective

the language models produced in the way just described, with equivalent language models built on the same data using the SRI language model toolkit (`http://www.speech.sri.com/projects/srilm/`). The differences between the two were minimal.

Slight differences in perplexity depending upon the dictionaries used for segmentation prompted the work on minimal perplexity Chinese word segmentation that was begun during this workshop and discussed in a later section.

6.2 Acoustic Modeling

Standard MFCC-based acoustic models with 14 mixtures per state were constructed using HTK version 3.2 (Young et al., 2002). Two baseline acoustic model training sets were used:

- MBN: 1997 Mandarin Broadcast News corpus (Hub-4NE) `http://wave.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC98S73`, consisting of 30 hours of speech from mostly trained speakers.
- WUDEVTRAIN: the 6.3 hours of Wu-accented training data.

The MBN data was chosen since it matches our data in one respect, namely that it is wideband recording. However it differs in important respect in that the topics are quite different, the style of language is different (since newsreaders are reading from prepared text) and the kinds of phonetic reductions one finds in conversational speech are much less prevalent.[2]

[2]Experiments were tried with using Mandarin Hub5 telephone speech. This required downsampling and speaker normalization (cepstral means and variances) of the Wu data. In principle this might provide a better baseline than MBN

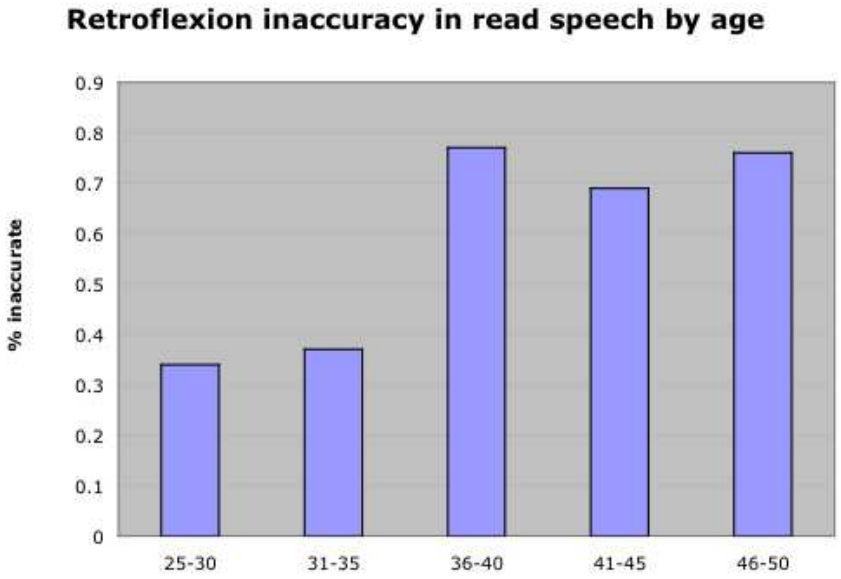

Figure 8: Hypercorrection in read speech: recognition perspective

6.3 AT&T Speech Recognition Tools

The AT&T approach to ASR based on weighted finite state transducers has been described extensively elsewhere—e.g. (Mohri, Pereira, and Riley, 2002)—and will only be briefly summarized here.

The basic idea behind the approach is that the problem of speech recognition can be cast in terms of rational transductions over strings. Given an acoustic model output in terms of weighted sequences of states of an acoustic model, typically representing possible sequences of triphones, one wishes to map between that sequence and a sequence of words, which is the output of the recognizer. This can be broken down into a set of mappings. The first of these, called the *Context* (*C*) transducer maps between HMM state sequences and sequences of phones. The second, the *Lexicon* (*L*) maps between sequences of phones and sequences of possible words. Finally, the *Grammar* (G), is a weighted finite-state acceptor that represents the language model for the task. One combines these three components by *composition* (notated $\circ$) so that during recognition one runs with the transducer $C \circ L \circ G$ (the "CLG" transducer). Note that since weighted finite-state transducers are closed under composition CLG is guaranteed to be a weighted finite-state transducer. Also, since transducers are closed under inversion one can construct the transducers to map in whichever direction seems more natural, since the result can always be inverted; thus it is more natural to think of a lexicon as mapping from words in their standard orthographic representation into (a set of) pronunciations, so one can construct the *L* transducer in that direction and then invert the result. Generally, during construction of the CLG one performs various optimizations including local determinizations and symbol pushing, and one *indexes* the result to allow the input labels to be looked up efficiently: see

since it is conversational speech and both the style and the topics discussed would in principle more closely match our data. In practice however we were unable to show any improvement over the MBN baseline using this approach.

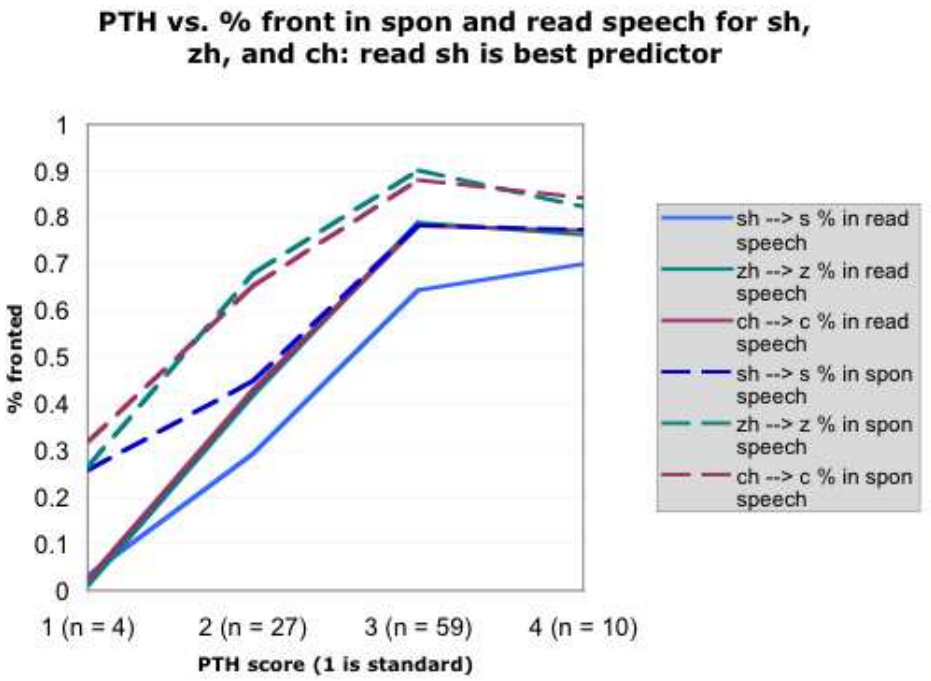

Figure 9: Comparison of fronting percentages with PTH score.

(Mohri, Pereira, and Riley, 2002) for details.

The AT&T recognizer *drecog* (available at `http://www.research.att.com/sw/tools/dcd`) uses the above CLG model during decoding, along with an acoustic model. In the set up at Johns Hopkins HTK acoustic models are used, and are converted at runtime to AT&T BLASR format. The details of the acoustic model and parameters to control aspects of the decoding such as the beam width, language model weight, and so forth are controlled by a parameter file. A typical example is given in Figure 10.

6.4 Baseline Results

The overall results for the MBN and WUDEVTRAIN baselines are given in Table 11. Here and henceforth all results will be reported in terms of *Character Error Rates* (CER).

7 Oracle Experiments

7.1 Lexicon Adaptation Oracles

A good oracle experiment for lexicon modeling is to assume that we know exactly which pronunciations the speaker would use for the words they utter and replace those words in the lexicon with the pronunciations that were actually used. We tried two approaches to this. One was to use the hand transcribed pronunciations and the other was to derive the pronunciations from forced alignment using a dictionary modified to allow for selected sound changes. In the latter case we

Speaker ID	Gender	Age	Education	PTH	Fluency	Rec. Loc.	Rec. Mic.
008	Male	26	UG	2A	Yes	1	2
009	Male	35	UG	2B	Yes	1	1
011	Male	30	UG	2B	Yes	2.1	2
012	Male	34	UG	2B	Yes	2.1	1
016	Male	26	JC	2B	Yes	2.2	1
032	Male	34	UG	3B	Yes	4	1
035	Male	36	JC	3B	Yes	2.2	1
043	Male	44	UG	3A	Yes	4	1
046	Male	50	SHS	3A	Yes	2.1	2
047	Male	40	SHS	3A	No	3	1
053	Female	45	TSS	3B	Yes	3	1
054	Female	45	TSS	2B	Yes	3	2
059	Female	40	SHS	3B	Yes	1	1
061	Female	30	UG	2B	Yes	2.1	2
064	Female	34	UG	2B	Yes	2.1	1
066	Female	26	UG	2A	Yes	2.1	2
067	Female	33	UG	2A	Yes	2.1	2
076	Female	41	UG	3B	Yes	1	1
098	Female	41	SHS	3A	Yes	3	1
099	Female	41	JC	3B	Yes	3	2

Table 9: Details of the test speakers. See Appendix A for an explanation of the codes.

AM Training Corpus	Data Size	# of tied-states	CER (%)
MBN	30 hours	5797	61.0
WU	6.3 hours	1334	44.2

Table 10: Performance of baseline acoustic models (AM) trained on MBN and WU corpus, respectively

focussed on the three fronting rules involving *sh*, *zh* and *ch* and the velarization of *in* to *ing*; the latter were implemented using the lextools package (`http://www.research.att.com/sw/tools/lextools`) and composed with the transducer representing the dictionary; see Figure 11.

In the case of hand transcriptions, all transcriptions for a given test speaker were parsed into word pronunciations, and the word pronunciations were weighted with MLE probability estimates given their frequency. They were then replaced into the original baseline dictionary, converting the probabilities to log probabilities and then scaling so that the most frequent probability had log probability of 0.0; this is necessary since otherwise any word for which we have weighted multiple pronunciation will automatically be penalized relative to other words for which we have a single (free) pronunciation.

In the case of forced alignment, the original dictionary was composed with the rules in Figure 11 and used to build a CLG which was then composed with the actual transcription T for each test utterance ($C \circ L \circ G \circ T$). These combined FST's were then used to "decode" the sentence. The

```
fsms          /home/yzheng/ws04/research/casr/eval/models/hmm144/att_728/SCLG.fsm
model_type    htk
model         /home/yzheng/ws04/research/casr/eval/models/hmm144/att_728/att.mmf
htk_hmmlist   /home/yzheng/ws04/research/casr/eval/models/hmm144/att_728/att.clist
htk_hmmmap    /home/yzheng/ws04/research/casr/eval/models/hmm144/att_728/att.hmmmap
beam            14
gram_mult       14
output          onebest
segment_level   word
lattice_beam    8
```

Figure 10: A typical *drecog* parameters file.

MBN CER	WUDEVTRAIN CER
61.0	44.2

Table 11: Overall baseline results for MBN and WUDEVTRAIN

cheapest path of the resulting lattice contains the pronunciations for the given utterance most favored by the recognizer. These pronunciations were then collected and the single best pronunciation for each word was replaced into the baseline dictionary.

The forced alignment dictionary with a single pronunciation resulted in a 1.4% reduction in CER over the MBN baseline (61%) over the 20 test speakers; speaker by speaker scores are given in Table 12. The hand-transcription-derived dictionaries were tested for five speakers and resulted in either no improvement or a worsening of error rate by as much as 2.3%.

The roughly 1.5% gain for the forced alignment oracle is consistent with previous work, such as (Huang et al., 2000), that tends to show minimal gains for pronunciation modeling. This result is not a true upper bound, since our experiment was biased in two ways against a win from pronunciation modeling. First, we only considered a small set of pronunciation changes, so there would in principle be an opportunity for larger gains if more changes were taken into account, (although we did consider the most important changes). Second, we did not retrain the acoustic models with the new pronunciations, which is known to be crucial for most pronunciation modeling wins. Nonetheless, these results might suggest that the maximum gain from pronunciation modeling might not exceed a few percent.

8 Automatic Identification of Age and PTH Level

As we saw earlier, there is a correlation between degree of accentedness and factors such as age. This therefore raises the possibility that one might detect age, and then use that as a factor in selecting appropriate models. To the extent that age is expected to correlate with strength of accent in any dialect region of China, age detection counts as a domain independent method for accentedness prediction.

Speaker	Oracle CER	Baseline CER
008:	63.9	63.9
009:	62.8	63.8
011:	65.3	70.1
012:	59.0	58.9
016:	67.7	67.7
032:	45.3	48.1
035:	57.9	59.3
043:	57.2	58.6
046:	70.1	71.0
047:	71.7	72.2
053:	81.2	84.3
054:	59.7	59.8
059:	66.4	71.8
061:	50.7	51.6
064:	39.7	40.0
066:	48.6	49.7
067:	50.9	50.9
076:	49.4	50.5
098:	73.1	75.1
099:	70.2	73.6
Total:	59.6	61.0

Table 12: Forced alignment oracle lexicon results.

```
optional

[sh][iii] -> s[ii]   # shi -> si
[zh][iii] -> z[ii]   # zhi -> zi
[ch][iii] -> c[ii]   # chi -> ci

[sh] -> s / _ ([<sigma>] - [iii])
[zh] -> z / _ ([<sigma>] - [iii])
[ch] -> c / _ ([<sigma>] - [iii])

[in] -> [ing]
```

Figure 11: Lextools rules used in forced alignment. Handled are the three cases of fronting and the velarization of final *in*. Note that special rules are needed for the sequences *shi*, *zhi* and *chi* since in addition to the fronting there is an obligatory vowel change.

8.1 Age Detection

In previous work Shafran et al. (2003) investigated the automatic detection of age as one of a number of "voice signatures". In their data, which consisted of calls from AT&T customers the actual age of the speakers was not known, but rather assigned by judges into one of three categories: youth ($<$25), adult (25–50), senior ($>$50). Gaussian Mixture Model classifiers using standard Mel Frequency Cepstral Coefficients plus a normalized f0 were developed. The system performed at 70.2%, where the baseline was 33%.

Our task is harder in that the age range of our speakers is narrower than that of Shafran et al., and in fact corresponds to their "adult" range, with our youngest speaker being 25 and our oldest 50. One assumes that part of the reason that the AT&T work achieved good results was that older speakers often exhibit a significant change in voice quality in their sixties, and the voices of people significantly under 20 often have not completely matured. Since our speakers were within a narrower range we divided them into two groups, the younger speakers being under forty and the older speakers forty and over.

Using the hand annotation for age for the 80 training speakers, we trained three-state HMM's with 80 mixtures for the single emitting state, treating each utterance for each speaker as being an instance of a single "phone", either *older* or *younger*. Two forms of data were used for training and testing:

- MFCC: Standard 39 component MFCC plus energy.
- MFCC+f0: The above, plus normalized f0. This adds three further features, namely f0, Δf0 and $\Delta\Delta$f0. The normalization was computed as in (Ljolje, 2002), whereby: $f0_{norm} = log(f0) - log(f0_{min})$.

During testing the utterances were decoded and the older/younger classification was performed, based on a simple majority of the automatic classifications. Since we have observed statistical differences between read and spontaneous speech, we did separate training on both read and spontaneous speech. We also tested four possible training-testing combinations. The results are given in

	Test: Spontaneous		Test: Read	
	MFCC	MFCC+f0	MFCC	MFCC+f0
Train: Spontaneous	13	14	14	10
Train: Read	13	12	13	14

Table 13: Results for age detection. The baseline for this task is 11/20.

	Test: Spontaneous		Test: Read	
	MFCC	MFCC+f0	MFCC	MFCC+f0
Train: Spontaneous	12	15	11	10
Train: Read	14	15	15	15

Table 14: Results for PTH-level detection. The baseline for this task is 10/20.

Table 13; note that the baseline for this task is 11/20. The best results were obtained for training on spontaneous speech with MFCC+f0 and testing on spontaneous speech, for training on read speech with MFCC+f0 and testing on read speech, and (curiously) training on spontaneous speech with MFCC and testing on read speech.

To the extent that we are able to detect older versus younger speakers this is potentially useful. In theory a general age detector could be built that does not depend upon in-dialect data, which could then be extended to accented speech from other dialect regions. The counter to this presumption is that since we do not know what features the GMM's are using in their classification it is possible that they are picking up on features such as the ratios of /s/ versus /sh/-like sounds, which we know correlate with age, but which also are clearly dialect-region dependent.

8.2 PTH Level Detection

An identical approach to that used for age detection can be applied to the problem of detecting PTH level directly. This is a less useful task from the point of view of building a general accentedness detector since it presumes a corpus hand-labeled with PTH level.

Huang, Chen and Chang (2004) used MFCC-based GMM's to classify 4 varieties of accented Putonghua including speakers from Beijing, Guangzhou, Shanghai and Taiwan. Correct identication ranged from 77.5% for Beijing speakers to 98.5% for Taiwan speakers.

We performed a similar experiment to use GMM models to classify speakers as more accented versus less accented based on the hand assigned PTH levels 2A–B (more standard) versus 3A–B (more accented). Once again we compared straight MFCC and MFCC+f0 and once again we trained on both spontaneous and read speech, testing all four possible combinations of training and testing. Results are given in Table 14. For MFCC alone there is a quite striking difference between spontaneous and read speech in that read speech models seem to be "sharper" and more able to distinguish between PTH levels; more surprisingly, models built on read speech seem to work well on spontaneous speech, though the reverse does not hold. The differences between spontaneous speech and read speech are nullified when f0 is added, but read speech models still work well on spontaneous speech.

One possible explanation for the higher performance of read speech models is that when people are reading they tend to become more standard in their pronunciation — if they are capable of doing

so. People who tend to be fairly standard may become more standard during reading, whereas speakers with heavy accents may simply not be able to overcome their accent even in the more controlled setting of reading aloud.

9 Automatic Speaker Clustering

A more direct approach to grouping speakers into bins is to compute some measures of individual speakers, and then use a clustering algorithm to divide the speakers into groups. In our case we are interested in two groups, more accented and more standard.

Three features that we have shown elsewhere to be related to degree of accentuation for Shanghai speakers are the following:

- $\frac{count(s)}{count(s)+count(sh)}$
- $\frac{count(z)}{count(z)+count(zh)}$
- $\frac{count(c)}{count(c)+count(ch)}$

Since we did not wish to presume a hand transcription of the database we investigated computing these ratios automatically from decoding output. The single best transcription is errorful, but in previous work (Bacchiani et al., 2004; Saraclar and Sproat, 2004) it has been shown that if one computes weights for strings over a lattice rather than over the single best path, one can generally improve one's estimate of the population statistics. The output of *drecog* cannot be interpreted directly as a probability distribution over strings, but the AT&T FSM toolkit provides a *pushing* algorithm (Mohri and Riley, 2001) that moves weights so that the resulting set of weights can be interpreted as (negative log) probabilities. The result of this pushing is that one can reconstruct the probability of an arc, which is to say the set of paths leading through that arc, by semiring timesing the state potential of the origin state of the arc, and the weight on the arc itself. Following (Saraclar and Sproat, 2004) we construct a "count" $C(l|L)$ for a given label l in a lattice L, as:

$$\begin{aligned} C(l|L) &= \sum_{\pi \in L} p(\pi) C(l|\pi) \\ &= \sum_{\pi \in L} \Big(p(\pi) \sum_{a \in \pi} \delta(a,l) \Big) \\ &= \sum_{a \in L} \Big(\delta(a,l) \sum_{\pi \in L : a \in \pi} p(\pi) \Big) \\ &= \sum_{a \in L} f(k[a]) p(a|k[a]) \end{aligned}$$

where $C(l|\pi)$ is the number of times l is seen on path π, $p(\pi)$ is the probability of path π, $f(k[a])$ is the state potential for input state $k[a]$ leading to the arc a, and $\delta(a,l)$ is 1 if arc a has the label l and 0 otherwise. In this way, estimates of the phoneme populations required above could be derived for each lattice and hence for each speaker, and the requisite ratios computed.

The lattices were derived by running the decoder using the MBN baseline acoustic model, and the standard language model. For the training data this is cheating somewhat since the language

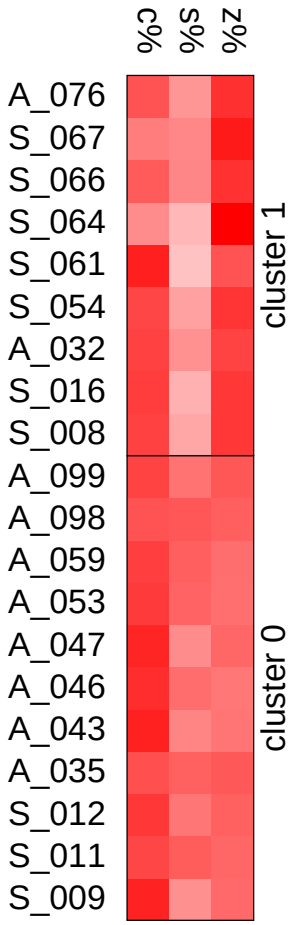

Figure 12: Output of two-way clustering of the test speakers using fronting ratios. For comparison the hand assigned labels of "A" (accented) and "S" (standard) are prepended to the speaker ID's. It will be seen that Cluster 1 is mostly standard and Cluster 2 is mostly accented.

model includes the transcriptions for the training data. However since the CER was 61% for this baseline, it is unlikely that this was a significant benefit.

An additional feature that was considered was age. In the case of the training data we used the actual age of the speaker; in the case of the testing data we used the predicted age from the best age predictor discussed in Section 8.1.

We used the Cluto 2.1.1 (Karypis, 2003) clustering toolkit to decide upon a clustering for both the training and testing data; two conditions were considered, namely using only the fricative/affricate ratios and using those plus age. By default the *vcluster* tool in Cluto uses a repeated bisections method with a cosine distance measure. An example of the two clusters produced can be seen in Figure 12.

10 Adaptation of Acoustic Models

Previous research (Wang, Schultz, and Waibel, 2003) suggests that MLLR can be used on groups of speakers in training set to help adapt acoustic models to foreign accents. However, applications of MLLR in this multi-speaker adaptation environment have been limited to a *single global transform*. Huang et al. (Huang et al., 2000) used MLLR with 65 phone-based transforms on individual test speakers, but they turned off the MLLR in their standard baseline system.

In this section, we explore adaptation techniques in both speaker independent (SI) and speaker dependent (SD) systems. Section 10.1 shows that combining MLLR with multiple transforms and MAP can improve the recognition performance. In order to show that the gain we get from speaker independent adaptation can be further improved when speaker dependent adaptation is used, the result of speaker dependent adaptation experiments is shown in Section 10.2

10.1 Supervised adaptation on training set

We experimented with various supervised adaptation techniques on the training set. Results are show in Table 15. This table shows that *IF-60*, the MLLR with 60 phone-based transforms, is significantly better than *Auto-60* which is the MLLR of 60 transforms by data-driven clustering. By applying MAP on top of both of the MLLRs, the gap is narrowed. We also found that the best combined system is 1.7% absolute better than applying MAP alone.

Baseline (no adaptation)	+ MAP	+ MLLR (Auto-60)
61.0%	45.4 %	51.2%
+ MLLR (IF-60)	+MLLR+MAP(Auto-60)	+MLLR+MAP(IF-60)
47.8%	44.5%	**43.7%**

Table 15: CER (%) Comparison of varies types of adaptation to baseline acoustic models trained on MBN corpus

10.2 Test speaker unsupervised MLLR adaptation

It has been reported (Huang et al., 2000; Mayfield Tomokiyo and Waibel, 2001) that speaker dependent MLLR adaptation is very useful for accented or non-native speech. We performed speaker-dependent adaptation on both MMIF-60 and WU baseline models, where MMIF-60 represents the best model of +MLLR+MAP (IF-60) in Table 15. Two global transforms are used in our experiment, one for the silence model and one for speech models. The results in Table 16 shows that we can get about 3% absolute gain after speaker adapatation.

Table 16 also shows the speaker averaged CER for “more Standard” group and “more Accented” group, which have been defined by fricatives and affricates ratio classifier discussed previously. It can be observed from the table that MMIF-60 favors “more standard” speakers, and WU favors “more accented” speakers. For comparison, the results of speaker independent systems for the same groups of speakers are also listed in Table 16.

10.3 Adaptation to Clusters

Considering the diversity of the speakers, we believe we can adapt to individual speaker clusters in order to improve performance.

	Speaker-independent		Speaker-dependent	
Speaker Group	WU	MMIF-60	WU	MMIF-60
more standard	39.6	**37.5**	36.5	**34.7**
more accented	**49.0**	50.3	**46**	47
Speaker Avg.	41.5	41.7	44.5	44.8

Table 16: Speaker averaged CERs (%) of speaker dependent (SD) and speaker independent (SI) systems

First we directly adapted to the hand separated speaker cluster, and found that performance has decreased comparing to adapting on the entire training set. This is probably caused by insufficient data when adapting only on speaker clusters. The adaptation method used in this experiment is 3 iterations of Auto-60 MLLR and 3 iterations of MAP.

Adapted From	TEST			
	c1	c2	c3	c4
c1 Male (more) Standard Speakers	53.8	52.4	58.4	71.1
c2 Male Accented Speakers	50.8	45.6	53.6	67.5
c3 Female (more) Standard Speakers	62.8	57.7	36.1	54
c4 Female Accented Speakers	63.4	57.7	38.8	52.1
auto-MLLR+MAP	48.1	43.6	37.3	52
WU Baseline	48.7	43.4	38	49

Table 17: Direct Adapt from MBN to the hand separted clusters

We considered the sparse data problem, and then experimented with first adapting on the entire training set, and then adapting again on the speaker clusters, thus using all the data while attempting to shift the weight toward the data that is more similar to the particular test speaker. The adaptation method used in this experiment is 3 iteration of 60-Auto MLLR and 3 iteration of MAP to adapt on the entire training set, and then 3 iterations of 60-Auto MLLR and 3 iterations of MAP of adaptation on the speaker clusters.

Adapted From	TEST			
	c1	c2	c3	c4
c1 Male (more) Standard Speakers	51	46.2	46.9	60.7
c2 Male Accented Speakers	48.6	44.6	47.5	59
c3 Female (more) Standard Speakers	54.8	48.7	36.7	52.9
c4 Female Accented Speakers	55.2	49.9	38.3	50.3

Table 18: Adapt from MBN to the hand separted clusters by two steps

We also experimented on IF transform by substitute Auto-MLLR with 60-IF MLLR, which improved the system performance.

Although the above experiments didn't show any improvement of adaptation on speaker clusters, we think it might be useful to try automatic classifications and use larger clusters. Table 20 shows the experiment results:

In order to compare the hand cluster and the *scz* ratio automatic cluster performance, we merged the 4 hand clusters into 2 clusters and have the follow results. The pronunciation ratio automatic-

Adapted From	TEST			
	c1	c2	c3	c4
c1 Male (more) Standard Speakers	50.3	46.9	40.3	61.9
c2 Male Accented Speakers	49.4	44.1	47.6	59.6
c3 Female (more) Standard Speakers	54.5	49.2	36.5	51.7
c4 Female Accented Speakers	55.1	49.6	38.3	50.7
IF-MLLR+MAP	48.7	42.9	35.6	50.7

Table 19: IF-MLLR + MAP Adaptation from MBN to the hand separted clusters by two steps

Adapted From	TEST			
	scz-c1	scz-c2	scz-c3	scz-c4
scz-c1 more Standard Speakers based on scz ratio	40.7	52.1		
scz-c2 Accented Speakers based on scz ratio	41.8	50.1		
scz-c3 more Standard Speakers based on scz ratio and age			38.1	49.6
scz-c4 Accented Speakers based on scz ratio and age			39.5	48.2
IF-MLLR+MAP	37.9	49.6	35.9	47
WU Baseline	40	48.4	37.9	46.8

Table 20: IF-MLLR + MAP Adaptation from MBN to the automatic separted clusters based on

cluster outperformed even hand separation cluster, which suggest automatic speaker clusters maybe more suited for this particular task. But over all, cluster adaptation performance is still worse than the adaptation on the entire training set.

From the above acoustic adaptation experiments, we find that acoustic modeling is able to capture the pronunciation variability pretty good. Fig. 13 shows the comparison of Gaussian Mean before and after adaptation, where the axes are first and second principal components of the means of the 14 mixtures of middle state of phoneme s and sh.

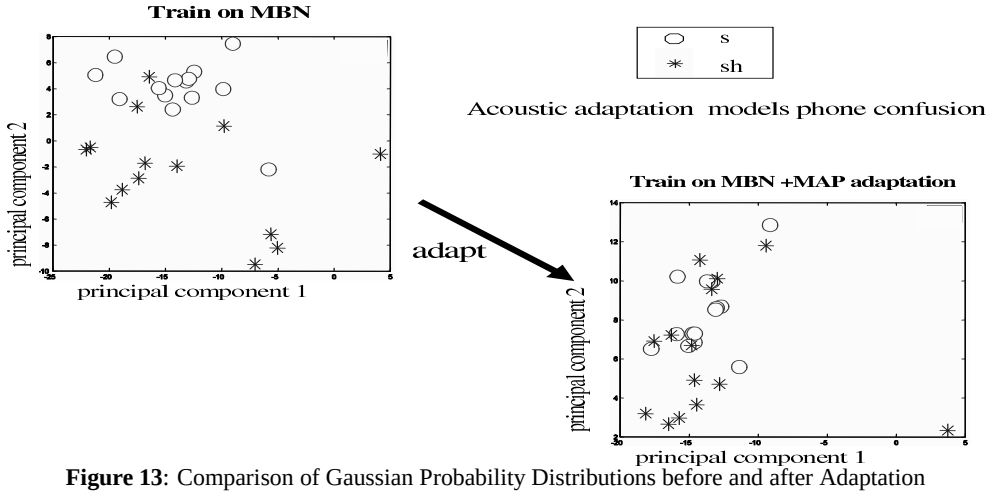

Figure 13: Comparison of Gaussian Probability Distributions before and after Adaptation

As a comparison, we did experiments using multiple pronunciation dictionary and applying force-alignment on the training data to find the best pronunciation before adaptation. Experiment results are shown in Table 22. Even though the results are slightly better than using single pronunciation for the cluster adapted cases, they still cannot outperform the adaptation on the whole training

Adapted From	TEST	
	More Standard	More Accented
More Standard	44.7	52.5
More Accented	44.4	48.8
IF-MLLR+MAP	42.1	47.6
WU Baseline	42.7	46.1

Table 21: IF-MLLR + MAP Adaptation from MBN to the hand separted "more accent" and "more standard" clusters

set.

In conlusion, cluster adaptation cannot outperform the adaptaion using the whole training set, which indicates that the current adaptation algorithm (either MAP or MLLR) cannot deal with the overtraining problem.

11 Study of accent discriminative acoustic features

The results in Table 16 show that there is an approximately 10% (absolute) gap between "more accented" and "more standard" speakers for all the SI and SD models. In this section we present methods for improving the performance of "more accented" speakers so that the gap can be narrowed.

In (Liu and Fung, 1999), Liu and Fung show that besides energy, formant frequency and pitch are also helpful in a task for accent classification. It is reasonable to assume that some acoustic features, such as formant parameters, pitch, word-final stop closure duration etc., might be more discriminative for accented speech. Therefore it may be helpful to add some of these features to the "accented speech recognizer". To test this assumption, we carried out preliminary experiments by appending formant parameters to MFCC features. The formant parameters were estimated automatically using the formant tracking algorithm in (Zheng and Hasegawa-Johnson, 2004).

In our experiment, we choose first three formants ($F_1^3 = [F_1\ F_2\ F_3]$) and their amplitudes ($\eta_1^3 = [\eta_1, \eta_2, \eta_3]$) as the accent related features. The detailed definition and estimation formulas of η are given in (Zheng and Hasegawa-Johnson, 2004). Two acoustic models were trained by appending F_1^3 and η_1^3 to the 39 dimensional MFCC vectors respectively.

The results are given in Table 23. We observed that the model with η_1^3 was able to improve 5 out of the 11 speakers in the "more accented" group; and the model with appended F_1^3 was only able to improve 2 out of the 11 speakers in the "more accented" group. The performance was degraded for speakers in the "more standard" group for both models.

The above experiment shows that formant amplitudes η_1^3 might contain extra information for

Adapted From	TEST			
	scz-c1	scz-c2	scz-c3	scz-c4
scz-c1 more Standard Speakers based on scz ratio	39.3	50.7		
scz-c2 Accented Speakers based on scz ratio	39.8	50		
scz-c3 more Standard Speakers based on scz ratio and age			37.2	48.8
scz-c4 Accented Speakers based on scz ratio and age			38.2	48

Table 22: Multiple Pronunciation Dictionary plus Adaptation

accent discrimination. We therefore constructed a new *accent favorable* model $\mathbf{WU}_\eta$ by finding the best path in the union of the two decoding lattices from the Wu baseline model and the new model with extra feature dimensions η_1^3. As shown in Table 23, compared to the WU baseline model, the overall CER for this group is reduced to 48.2%, and the CERs were reduced for 8 out 11 speakers in the "more accented" group.

A similar experiment was done for speaker dependent system, where two models (WU and $MFCC + \eta_1^3$) were adapted for each individual test speaker and a WU_η was obtained for each test speaker. Compared to the WU baseline model test speaker adaptation, the CERs were reduced for 9 out 11 speakers in the "more accented" group.

	MFCC+F_1^3	MFCC+η_1^3	WU_η
SI	49.4	48.9	**48.2**
SD	-	46.1	**45.6**

Table 23: Average CER (%) of more accented speakers by modeling both MFCC and formant parameters

12 Model Selection

12.1 Model selection based on accentedness

To make use of the prior knowledge of accentedness, we proposed a model-selection algorithm. Suppose that there are M different acoustic models, θ_1, θ_2, ... , θ_M, given observation x, we want to find the best acoustic model according to Eq. 1,

$$\begin{aligned}\theta_{MAP} &= \underset{k=1,2,\cdots,M}{argmax}\ p(\theta_k|x) \\ &= \underset{k=1,2,\cdots,M}{argmax} \sum_a \underbrace{p(\theta_k|a)}_{\theta_k \perp x|a} \quad \underbrace{p(a|x)}_{\text{accentedness classifier}}\end{aligned} \tag{1}$$

where a is the accentedness variable.

For a binary accentedness classification, we have M=2,

$$a = \begin{cases} 1 & \text{if the speaker is "more standard"} \\ 2 & \text{if the speaker is "more accented"} \end{cases}$$

and

$$p(\theta_k|a) = \delta(k-a)$$

To make Eq. 1 work, first, we need a reliable accentedness classifier, as described in the previous section; second, we need to find the acoustic model θ_k, which is most appropriate for the degree of accentedness. In Section 10 and 11, we show how to find two acoustic models which favor different accent groups. And Section 12.2 reports the results of *model-selection* experiment, showing the effectiveness of the accentedness classifier.

12.2 Experiment of model selection

In this section, we use the following model selection strategies:

$$\theta = \begin{cases} \theta_{MMIF-60} & \text{if the speaker is in cluster 1} \\ \theta_{WU} \text{ or } \theta_{WU_\eta} & \text{if the speaker is in cluster 2} \end{cases} \quad (2)$$

Table 24 shows the results of model selection between WU or WU_η and MMIF-60 models based on automatic accent detection results from Section 9. The results shows that by using the ratio of counts of particular fricatives and affricates as the input of accent classifier, we were able to improve the WU baseline by 1% absolute in both SI and SD cases. Furthermore, formant amplitude η is useful to discriminate "accent speakers".

		WU+MMIF-60		WU_η + MMIF-60	
		GMM	SCZ	GMM	SCZ
SI	more accented	-	49	-	**48.2**
SI	speaker avg.	44.4	43.8	44.3	**43.4**
SD	more accented	-	46	-	**45.6**
SD	speaker avg	41.3	40.9	41.2	**40.7**

Table 24: CER (%) for model selection based on the detection of accent, where "WU+MMIF-60" means selection between WU and MMIF-60 models, and "WU_η + MMIF-60" means selection between WU_η and MMIF-60 according to Eq. 2. "SCZ" means model selection based on accent detection using the ratio of counts of particular fricatives and affricates. "GMM" means model selection based on the accent detection of GMM classifier.

12.3 Conclusion

Our research shows that different acoustic models have advantages for different group of speakers. We report the approach of combining accent detection, accent discriminative acoustic features, acoustic adaptation and model selection to the problem of accented Chinese speech recognition. Experimental results show that our proposed approaches achieved 1.0∼1.4% absolute reduction of character error rate over the most state-of-the-art acoustic modeling techniques on Wu-accented Chinese speech.

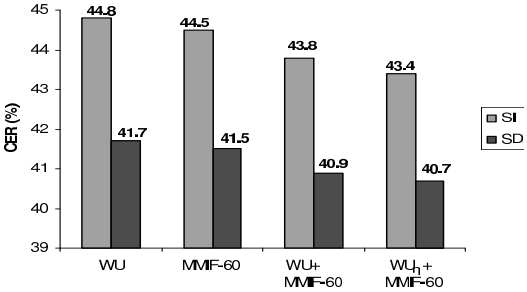

Figure 14: Summary of results

13 Speaker-Clustering-based Hybrid Acoustic Modeling on Wu-Accented Chinese Speech

13.1 Introduction

Acoustic modeling is a crutial component in accented or foreign accented speech recognition. Active research has been carried out in this area during the past few years. The proposed methods vary from simply collecting data in that accent and training a recognizer, to various ways of adapting recognizers trained on unaccented speech. Wang, Schultz, and Waibel (Wang, Schultz, and Waibel, 2003) investigated German-accented English speakers in the VERBMOBIL (conversational meeting planning) task. Tomokiyo and Waibel (Mayfield Tomokiyo and Waibel, 2001) examined the task of recognizing Japanese-accented English in the VERBMOBIL domain. In both of the tasks, it was found that training on non-native speech data, especially when mixed with in-domain native speech data, provides the most obvious gains in performance on accented data. The simplest use of adaptation was merely the direct use of MLLR (Maximum Likelihood Linear Regression) to adapt individually to each test speaker. In (Huang et al., 2000), in order to recognize Shanghainese-accented Putonghua, Huang et al. applied standard speaker MLLR adaptation to a Microsoft Whisper system that had been trained on 100,000 sentences from 500 speakers from the Beijing area. In (Wang, Schultz, and Waibel, 2003)(Mayfield Tomokiyo and Waibel, 2001) , MLLR was adapted not just to the single accented test speaker, but also to a larger number of accented speakers. Research in (Wang, Schultz, and Waibel, 2003)(Mayfield Tomokiyo and Waibel, 2001)(Huang et al., 2000) shows the effectiveness of MLLR or MAP (Maximum A Posteriori) adaptation on accented speech, but it did not report whether combining MLLR and MAP could be helpful for accented ASR.

While some promising results have been published on accented speech recognition using the above approaches, the recognition accuracy on accented speech is still lousy and definitely needs further improvement. Some research issues remain open. First, more sophisticated forms of MLLR or MAP may be applied, such as MLLR using more specific transforms rather than a single global transform. In particular, our research shows that current adaptation schemes have varied performance on different groups of speakers. Second, the effect of combining MLLR and MAP in accented ASR needs to be explored and optimized.

In this workshop, in order to improve current accented speech recognition performance, we will first apply and evaluate acoustic modeling algorithms to various accent-based speaker clusters. We then propose a new Accent-based Hybrid Acoustic Modeling (AHAM) method that applies Maximum Likelihood training and MAP/MLLR adaptation algorithms to more accented speakers and more standard speakers, respectively, while each speaker is classified as "more accented" or "more standard" using the phoneme-based automatic accent detection algorithm presented in the previous section. In our experiments on spontaneous Wu-accented Chinese speech recognition, the proposed AHAM method achieved higher recognition accuracy than conventional single-speaker-cluster-based acoustic modeling algorithms. In particular, the experiments show that more improvement may be achieved if more speech training data from accented speakers is available.

Note that although all experiments in this workshop were performed on WU-accented conversational Chinese speech only, we believe that our proposed approaches will also be helpful for accented speech with other Chinese dialects or in other languages.

13.2 Basic Theory and Problem Statement

During speech recognition, for any acoustic data string $A = \{a_1, a_2, \cdots, a_n\}$, the best hypothesized word string $W = \{w_1, w_2, \cdots, w_m\}$ is conventionally defined as

$$\hat{W} = \underset{W}{argmax} P(A|\bar{\lambda}, W)P(W) \tag{3}$$

where $P(A|\bar{\lambda}, W)$ is the acoustic model and $P(W)$ is the language model. In addition, $\bar{\lambda} = \{\lambda_1, \lambda_2, \cdots, \lambda_n\}$ is the set of statistical acoustic model parameters and is optimized as

$$\bar{\lambda} = \underset{\bar{\lambda}}{argmax}\{\prod_{A} P(A|\bar{\lambda}, W_A)\} \tag{4}$$

In this work, the language model $P(W)$ is out of our concern. Therefore, equation 4 may be simplified as

$$\bar{\lambda} = \underset{\bar{\lambda}}{argmax}\{\prod_{A} P(A|\bar{\lambda})\} \tag{5}$$

Equation 5 represents an ideal case where the acoustic model parameters are optimized over infinite acoustic data. In the reality, the acoustic training data available is always limited and can be denoted as $\phi_{training} = \{A_1, A_2, \cdots, A_k\}$. Accordingly, the acoustic parameter set optimized on $\phi_{training}$ is defined as

$$\bar{\lambda}_{training} = \underset{\bar{\lambda}}{argmax}\{\prod_{A \in \phi_{training}} P(A|\bar{\lambda})\} \tag{6}$$

We hope that $\phi_{training}$ is, although with limited size, fairly representation of all possible acoustic data ϕ (especially the acoustic test data ϕ_{test}), and hence $\bar{\lambda} \approx \lambda_{training}$. Unfortunately, acoustic training data are sometimes very difficult, if possible to collect. As a result, there are many cases where $\phi_{training}$ and ϕ_{test} are significantly different from each other due to some critical mismatches, such as speaking style, noise conditions, channel characteristics and accents, as shown in Figure 15. Under these mismatched situations, $\bar{\lambda} \neq \bar{\lambda}_{training}$ and only a sub-optimal acoustic model parameter set can be obtained and to be used in equation (3). Dramatic decline of recognition accuracy will then be observed, as expected.

Accent is perhaps one of the most critical and also most commonly observed matches in speaker-independent spontaneous speech recognition. In this workshop, experiments are performed on acoustically different test corpora to evaluate the impact of accent mismatch. The acoustic model was optimized on Mandarin Broadcast News (MBN). It was then tested on three acoustically distinct test corpora: MBN, Wu-Accented Chinese read speech, and Wu-Accented spontaneous speech. The recognition results are shown in Table 25. While 19.1% Character Error Rate (CER) was obtained on the matched MBN test set, the CER increased dramatically to 57.4% for Wu-Accented read speech test set and to 61.0% for Wu-Accented spontaneous speech test set. Those are more than 200% jumps in CER! It is obvious that some accent-based acoustic modeling techniques needs to be designed and used to attack this data/accent mismatch problem.

13.3 Conventional Acoustic Modeling for Accented Speech

Without losing generality, assume we have two acoustic training sets: $\phi_{standard}$ and $\phi_{accented}$. $\phi_{standard}$ consists of speech from standard Mandarin speakers and $\phi_{accented}$ consists of speech

Test-set	MBN	Wu-Accented Read Speech	Wu-Accented Spontaneous Speech
CER	19.1%	57.4%	61.0%

Table 25: Recognition performance of acoustic model $\bar{\lambda}_{MBN}$ trained on MBN corpus

from accented speakers. The acoustic test set consists of speech from accented speakers. Furthermore, the size of $\phi_{standard}$, $N(\phi_{standard})$, is usually much greater than the size of $N(\phi_{accented})$, . In our experiments, is five times as big as . Due to this huge difference in data account between $\phi_{standard}$ and $\phi_{accented}$, there are conventionally two different acoustic modeling methods for accented speech:

1. Optimizing acoustic model parameters by maximizing likelihood on accented speech only, i.e.,
$$\bar{\lambda}_{accented} = \underset{\bar{\lambda}}{argmax}\{\prod_{A \in \phi_{accented}} P(A|\bar{\lambda})\} \tag{7}$$
Conventional EM (Expectation-Maximization) algorithm can be used to calculate the above equation.

2. Adaptation of acoustic model parameters from standard-speech-oriented to accented-speech-oriented via MAP (Maximum A Posterior) or MLLR (Maximum Likelihood Linear Regression)
$$\bar{\lambda}_{standard} = \underset{\bar{\lambda}}{argmax}\{\prod_{A \in \phi_{standard}} P(A|\bar{\lambda})\} \tag{8}$$
$$\bar{\lambda}_{standard} = \underset{\bar{\lambda}}{argmax}\{\prod_{A \in \phi_{accented}} P(A|\bar{\lambda}, \bar{\lambda}_{standard})\} \tag{9}$$
Greater details about MAP and MLLR adaptation have already been introduced in the previous chapter.

Since $\lambda_{accented}$ is optimized only on $\phi_{accented}$ while $\bar{\lambda}_{MAP+MLLR}$ is optimized on both $\lambda_{accented}$ and $\lambda_{standard}$, it is common sense that $\bar{\lambda}_{MAP+MLLR}$ is superior to $\lambda_{accented}$ when $N(\phi_{accented})$ is much less than $N(\phi_{standard})$. When $N(\phi_{standard})$ is fixed and $N(\phi_{accented})$ increases, the accuracy of $\lambda_{accented}$ improves much faster than $\bar{\lambda}_{MAP+MLLR}$, and, at one point, $\lambda_{accented}$ will surpass $\bar{\lambda}_{MAP+MLLR}$. Therefore, the optimal acoustic model may be derived from these two models and defined as

$$\bar{\lambda}_1 = \begin{cases} \bar{\lambda}_{MAP+MLLR} & if\ N(\phi_{accented}) < N(\phi_{standard}) \\ \bar{\lambda}_{accented} & if\ N(\phi_{accented}) \geq N(\phi_{standard}) \end{cases} \tag{10}$$

where the value of $N_{threshold}$ may be optimized upon some acoustic held-out sets. Alternatively, the optimal acoustic model may also be a linear combination of the above two models and defined as

$$\bar{\lambda}_2 = w_{MAP+MLLR}\bar{\lambda}_{MAP+MLLR} + w_{accented}\bar{\lambda}_{accented} \tag{11}$$

where $w_{MAP+MLLR}$ and $w_{accented}$ are the probability weights and $w_{MAP+MLLR}$ + $w_{accented} = 1$.

13.4 Accent-based Hybrid Acoustic Modeling (AHAM) via Speaker Clustering

The conventional acoustic modeling approaches described above partition Mandarin speakers into two classes: standard speakers and accented speakers. Accordingly, the acoustic training data is partition into $\phi_{standard}$ and $\phi_{accented}$. However, this binary classification is constrained in practice since our experiments show that the accent of each speaker is a matter of degree rather than binary. Therefore, we propose a new Accent-based Hybrid Acoustic Modeling (AHAM) method by partitioning speakers into multiple categories based on their accentedness. In particular, we will start from the simplest 3-class speaker clustering approach where the Mandarin speakers are clustered into standard speakers, slight-accented speakers and strong-accented speakers. The corresponding training sets are $\phi_{standard}$, $\phi_{slight-accented}$ and $\phi_{accented}$. For any acoustic data string $A = \{a_1, a_2, \cdots, a_n\}$, the best hypothesized word string $W = \{w_1, w_2, \cdots, w_m\}$ based on AHAM is defined as

$$\hat{W} = \begin{cases} \underset{W}{argmax} P(A|\bar{\lambda}_{accented}, W)P(W) & if\ g(A) = S_{strong-accented} \\ \underset{W}{argmax} P(A|\bar{\lambda}_{MAP+MLLR}, W)P(W) & if\ g(A) = S_{slight-accented} \end{cases} \tag{12}$$

where

$$\bar{\lambda}_{accented} = \underset{\bar{\lambda}}{argmax}\{\prod_{A \in \phi_{strong-accented}} P(A|\bar{\lambda})\} \tag{13}$$

$$\bar{\lambda}_{standard} = \underset{\bar{\lambda}}{argmax}\{\prod_{A \in \phi_{standard}} P(A|\bar{\lambda})\} \tag{14}$$

$$\bar{\lambda}_{MAP+MLLR} = \underset{\bar{\lambda}}{argmax}\{\prod_{A \in \phi_{slight-accented}} P(A|\bar{\lambda}, \bar{\lambda}_{standard})\} \tag{15}$$

$g(A)$ is an accent-detection function that classify any unknown speaker into strong-accented or slight-accented speakers. In this workshop, we proposed a phoneme-based accent detection algorithm, which has been introduced and discussed in the previous chapters. One drawback of conventional approaches on accented speech recognition shown in equation (7)-(11) is that it ignores the degree of accentedness among all the speakers. Our proposed AHAM categorizes speakers into multiple groups according to their accentedness and models them differently. More specifically, strong-accented speakers are modeled upon strong-accented speech training data via maximum likelihood criterion, while slight-accented speakers are modeled by adapting acoustic modeling trained on standard speakers to slight-accented speech training data. In this way, we can achieve a tradeoff of using both the information from matched but limited accented speech data and the information from unmatched but large-amount standard pronounced speech data.

13.5 Experiments

13.5.1 Experimental Setup

The conventional acoustic modeling approaches as well as our proposed AHAM method are evaluated in our offline 50k-words Wu-accented spontaneous Mandarin speech recognition experiments. Our test set consists of one-hour, 16-bit, 16kHz-sampled and digitally recorded Wu-accented spontaneous conversations from 10 male and 10 female speakers. For acoustic modeling, two acoustic

training sets were used: MBN training set ($\phi_{standard}$): 1997 Mandarin Broadcast News Corpus (Hub-4NE), which consists of 30 hours read speech from standard Mandarin speakers; WU training set ($\phi_{accented}$): 6.3 hours Wu-accented speech training data from 40 male and 40 female speakers. Standard 39-dimentional MFCC features were computed and used via HTK toolkit version 3.2 (Young et al., 2002). AT&T FSM tools were adopted for language modeling and speech decoding (Mohri, Pereira, and Riley, 2002). All the experimental results will be reported in Character Error Rate (CER), which is a standard measure of performance in Chinese speech recognition.

13.5.2 Experiments of Acoustic Modeling on Accent-based Speaker Clusters

In equation (13), we propose to train accented acoustic model $\bar{\lambda}_{accented}$ on a strong-accented training set. Various strong-accented training sets can be derived using varied accent detection algorithms. ASR experiments were carried out to compare the performance of these accented acoustic models with the baseline acoustic model that is trained on all accented speech data. The experimental results are listed in Table 26. We can see that models trained on strong-accented sub-training-set significantly outperform the model trained on slight-accented sub-training-set, which is a clear evidence of the importance of accentedness in our WU-accented speech recognition task. The best model $\bar{\lambda}_{weighted}$ was achieved when all the accented training data was used with a weighted-emphasis on strong-accented speech over slight-accented speech. However, the resulting CER of 44.78% is still worse than the baseline CER of 44.27%. This is because the 6.3 hours accented training data is so limited that any removal of speakers will deteriorate robust distribution estimation of all the Mandarin pronunciations, even if those speakers are without accents. We believe that when $\phi_{accented}$ is sufficiently large to estimation the distribution of all the Mandarin pronunciations, $\bar{\lambda}_{weighted}$ should be significantly better than $\bar{\lambda}_{baseline}$.

Similar experiments results were observed in acoustic model adaptation of $\bar{\lambda}_{MAP+MLLR}$ using strong-accented sub-training-sets.

Training-set for Acoustic Modeling	Test-set CER
All Wu-accented speakers: $\lambda_{baseline}$	44.27%
Speech from GMM-detected slight-accented speakers	49.67%
Speech from GMM-detected strong-accented speakers	48.06%
Speech from GMM+Age-detected strong -accented speakers	47.31%
Speech from manually-clustered strong-accented speakers	47.65%
All Wu-accented speakers with accent-based weights: $\lambda_{weighted}$	44.78%

Table 26: Performance comparison of acoustic models trained on various training sets

13.5.3 Experiments of AHAM on Accent-based Speaker Clusters

The performance of Accent-based Hybrid Acoustic Modeling (AHAM) method is evaluated and compared with baseline conventional acoustic modeling methods in Table 27. By selecting $\bar{\lambda}_{accented}$ or $\bar{\lambda}_{MAP+MLLR}$ for each speaker based on automatic accent detection, AHAM reduced the CER from 44.27% of $\bar{\lambda}_{accented}$ alone, and 44.37% of $\bar{\lambda}_{MAP+MLLR}$ alone, to 43.39%, representing a 0.88% absolute error rate reduction. If $\bar{\lambda}_{accented}$ and $\bar{\lambda}_{MAP+MLLR}$ can be selected correctly for each speakers as an ideal case, the CER can be further reduced to 43.23%, which is the Lower-bound

of AHAM CER. Although this 2% relative improvement is not very significant, we will show in the next subsection that the potential improvement of AHAM is significant when the accented training data is sufficient.

Acoustic Models	Test-set CER
accented l in equation (13): Baseline 1	44.27%
MLLR MAP+ l in equation (15): Baseline 2	44.37%
AHAM in equation (12)	43.39%
Lower-bound of AHAM	43.23%

Table 27: Performance comparison between baseline acoustic models and our proposed AHAM method

13.5.4 Experiments of AHAM with various amount of accented training data

As mentioned earlier, the amount of accented training data is crucial to the performance of various acoustic models. In order to evaluate the relationship of accented training data size and ASR performance, we randomly partitioned the 6.3 hours Wu-accented training data into several sub-sets by partitioning the speech data of each speaker proportionally. The experimental results are depicted in Figure 16. The chart shows that $\bar{\lambda}_{MAP+MLLR}$ performed best when only 1-hour accented data are available for training. $\bar{\lambda}_{AHAM}$ becomes the best model when the amount of $N(\phi_{accented})$ is about or more than 3 hours. In particular, the performance improvement of $\bar{\lambda}_{AHAM}$ over $\bar{\lambda}_{MAP+MLLR}$ and $\bar{\lambda}_{accented}$ is increasing consistently with the increases of $N(\phi_{accented})$: 0.49% for 4 hours training data, 0.67% for 5 hours training data, and 0.88% for 6 hours training data. As a result, we believe the improvement of $\bar{\lambda}_{AHAM}$ over conventional acoustic models will become bigger and bigger if the amount of accented training data keeps increasing.

Additional experiments were carried out to compare performance of AHAM with the performance of acoustic modeling selection based on MAP (Maximum A Posterior) criterion. The results are shown in Figure 17. While the improvement of AHAM in Figure 17 is smaller compared with the improvement shown in Figure 16 as it is now compared with an enhanced baseline. Nevertheless, the trend of CER reduction by using AHAM is similar and AHAM consistently outperforms its competitor whenever $N(\phi_{accented})$ is greater than 3.5 hours.

13.6 Summary

Acoustic modeling for accented speech recognition is a very important but also challenging task for most state-of-the-art speech recognition systems. Since current acoustic models are mostly trained on standard pronounced speech, models need to be either re-trained on accented speech data or adapted from standard pronunciations to accented pronunciations to achieve optimal ASR performance. Conventional acoustic modeling on accented speech treats speakers as either accented or non-accented. However, it is well known that the accent of each speaker is a matter of degree. Therefore, we propose a new accented hybrid acoustic modeling (AHAM) method that partitions speakers into multiple clusters and treats them differently. The accent of each speaker is detected via a phoneme-based accent detection algorithm proposed in the previous chapter. Base on the accent detection result, an appropriate acoustic model is then selected to decode all the speech for the specific speaker. We thereby take a good advantage of both the limited accented training data

available and large account of standard pronounced training data. Extensive experiments were carried out on speak-independent Wu-accented spontaneous speech recognition. AHAM consistently outperformed baseline acoustic models with a moderate margin. The improvement becomes more significant when more accented speech training data is available. While our current AHAM focuses on 3-class speaker classification, more speaker classes in AHAM will be investigate in our future work as more accented training data become available.

14 Speaker-Adaptive Training

Adaptation schemes investigated in the previous section, using MLLR and MAP, apply a linear transform to an initial speaker-independent model to bring it closer to the test speaker. These models improved performance on Wu-accented speech. Their performance can be further enhanced by reducing the variance of the initial speaker-independent model. Informal perceptual experiments and measurements of fronting of retroflexes through ratios have shown that the degree of accent varies significantly across speakers in the corpus. As a result the speaker independent model trained on the Wu-accented data has a large variance. It is well understood that speaker-adaptive model normalizes the speakers in the model space (Anastasakos et al., 1996), thus produces a compact speaker-independent model. So, we investigated the use speaker-adaptive model training (SAT) on Wu corpus.

In the first experiment, the iterations of speaker-adaptive model training (SAT) were initialized with the models trained on Wu data using maximum likelihood criterion (Wu-ML). Two sets of models were estimated (Wu-ML+SAT-1 and Wu-ML+SAT-2) by two iterations of SAT on the Wu training corpus. All the models were evaluated with and without MLLR adaptation on the test set. In all the experiments in this section, two transforms (silence and non-silence) were used. During test, initially, a single global MLLR transform was estimated, which was then used to obtain two MLLR transforms in 4 iterations. The results are shown in Table 28.

Model	without MLLR	with MLLR
Wu-ML	44.2	41.2
Wu-ML+Wu-SAT-1	43.7	39.4
Wu-ML+Wu-SAT-2	44.5	**39.1**

Table 28: Evaluation of speaker-adaptive model trained on Wu-accented training data.

The results of SAT models without MLLR is shown merely for the sake of contrast, and is not expected to perform better. When the models are evaluated with MLLR adaptation on the test set, the speaker-adaptive models clearly improve the performance of the baseline model, from 41.2 to 39.1. The performance of the Wu-SAT-Iter2 model with MLLR is also better than the best performance (40.7) obtained in the previous section.

Suppose Wu-accented training corpus was not available, and all we had was Mandarin Broadcast News (MBN) corpus, would SAT and MLLR be of help? To investigate this question, we examined the effect of training speaker-adaptive model solely on the MBN corpora. In this case, we initialized the SAT iterations on MBN training corpus with corresponding maximum likelihood models (MBN-ML). The models were evaluated on the Wu-accented corpus, with and without MLLR, as in the

previous experiment. As shown in Table 29, model space normalization on the mismatched training data (MBN corpora), hurts the performance badly on Wu corpus even with MLLR adaptation, and, in this case, use of SAT is not recommended.

Model	without MLLR	with MLLR
MBN-ML	60.9	54.8
MBN-ML+MBN-SAT-1	65.8	56.6
MBN-ML+MBN-SAT-2	67.4	57.6

Table 29: Evaluation of speaker-adaptive model trained on MBN training data.

An improvement on this, would be to train speaker-adaptive model on the Wu corpora, and initialize it with the best MBN-derived model (MBN-ML+Wu-MLLR-MAP) from the previous section. Recall, this model was obtained by 6 iterations of MLLR and 3 iterations of MAP over the Wu training corpus, using 120 transform where a transform modeled the onset and coda of allophones of one of the 60 phonemes. Once again, two sets of models were trained by applying two iterations of SAT (MBN-ML+Wu-MLLR-MAP+Wu-SAT-1/2). As shown in Table 30, SAT on Wu corpus, initialized with MBN-derived models, did not improve performance, and this may be due to mismatch in the model space. These models are better than the models trained only on MBN corpus.

Model	without MLLR	with MLLR
MBN-ML+Wu-MLLR-MAP	43.7	40.7
MBN-ML+Wu-MLLR-MAP+Wu-SAT-1	49.4	43.9
MBN-ML+Wu-MLLR-MAP+Wu-SAT-2	52.8	46.2

Table 30: Evaluation of speaker-adaptive model trained on Wu-accented training data.

In conclusion, our adaptation experiments in this section show that having about 10 hours of indomain acoustic data is worth more than larger amounts of out-of-domain data, especially, when there are significant differences both in acoustic conditions and in speaking styles.

15 Conclusions and Future Work

While we view the results achieved to be preliminary, we believe that we can draw the following conclusions from the work that we performed this summer.

First, we have proposed a new approach to dealing with ASR for accented speech, that depends upon the linguistic common-sense observation that accentedness is not an all-or-nothing proposition, but rather a matter of degree. Our approach thus involves first detecting the degree of accentedness, and then selecting acoustic models based on this degree of accentedness. We demonstrate a 1% overall CER reduction using these techniques. As part of this work we also developed accentedness-specific transforms using supervised MLLR plus MAP on accented training corpus.

Accentedness itself was detected using phone count ratios, which were computed from decoder lattices and usd for unsupervised speaker clustering.

Oracle experiments on pronunciation modeling suggest, consistent with previous reports, that one can only expect modest gains (perhaps 1.5%) from pronunciation modeling or lexicon adaptation alone; however see Appendix D for a report on some further results from lexicon adaptation.

Needless to say, the use of a binary classification into "more" or "less" accented speakers only goes part of the way to addressing the fact that accentedness is a continuous variable. The mathematical model in Section 12.1 is a proposal for future work wherein accentedness could be modeled continuously using arbitrarily fine divisions of accentedness.[3]

One of the side projects that shows some promise (and which will be developed in future work) was the work on minimal perplexity word segmentation, reported on by Philip Bramsen in Appendix E.

[3] On the pronunciation modeling front, a proposal by David Kirsch to do continuous pronunciation modeling modeling was one of the projects chosen for future support in the final day of the workshop. A description of this proposal can be seen in the final presentation on the project website.

Appendix A: Speakers

The following table lists details of the 100 speakers in our dataset. Given are the Speaker ID, Gender, Age, Education Level, Putonghua (PTH) Level, Fluency, Recording Location and Recording Microphone.

The abbreviations used in the Education column are as follows:

JHS	Junior High School
SHS	Senior High School
TSS	Technical Secondary School
PS	Polytechnic
JC	Junior College
UG	Undergraduate
Masters	Masters

Putonghua level and fluency were determined by experts at the Chinese Academy of Social Sciences (CASS). Putonghua level ranges from 1 (best) to 3 (worst), with subdivisions A (better) and B (worse); all of our speakers fall in the range 2A-3B. Speakers were assigned fluency on a two point scale, i.e. fluent or not fluent.

There were four major recording locations [Elaborate on these]:

1	Shanghai Wusi Junior High School
2	Shanghai Education Bureau
3	Factory
4	Academy

The suffixes ".1" and ".2" denote two different rooms in the same location.

Two microphones were used. They were:

1
2

Speaker ID	Gender	Age	Education	PTH	Fluency	Rec. Loc.	Rec. Mic.
001	Male	39	TSS	3A	No	3	1
002	Male	36	TSS	2B	Yes	3	2
003	Male	38	TSS	3A	Yes	3	1
004	Male	49	SHS	3A	Yes	3	1
005	Male	50	JHS	2B	No	3	1
006	Male	50	JHS	3A	Yes	3	2
007	Male	25	TSS	2B	Yes	2.2	1
008	Male	26	UG	2A	Yes	1	2
009	Male	35	UG	2B	Yes	1	1
010	Male	34	SHS	3A	Yes	2.1	2
011	Male	30	UG	2B	Yes	2.1	2
012	Male	34	UG	2B	Yes	2.1	1

013	Male	33	SHS	3A	Yes	3	2
014	Male	39	SHS	3A	Yes	3	1
015	Male	26	JC	3A	Yes	2.2	1
016	Male	26	JC	2B	Yes	2.2	1
017	Male	26	UG	3A	Yes	2.2	2
018	Male	27	SHS	2B	Yes	2.2	2
019	Male	29	UG	3A	Yes	2.2	2
020	Male	45	UG	3A	Yes	1	1
021	Male	45	SHS	3B	Yes	1	2
022	Male	48	UG	3A	Yes	1	1
023	Male	46	SHS	2B	Yes	1	2
024	Male	45	SHS	3A	Yes	3.1	2
025	Male	48	UG	3A	Yes	2.1	1
026	Male	50	SHS	3A	Yes	3	1
027	Male	48	SHS	3A	Yes	3	1
028	Male	41	UG	3A	Yes	2.2	2
029	Male	44	JC	3B	Yes	2.2	2
030	Male	46	SHS	3A	Yes	4	2
031	Male	41	SHS	3A	Yes	4	1
032	Male	34	UG	3B	Yes	4	1
033	Male	47	PS	3A	No	4	1
034	Male	29	SHS	3A	Yes	2.2	2
035	Male	36	JC	3B	Yes	2.2	1
036	Male	47	UG	3A	No	4	2
037	Male	46	JC	3A	Yes	4	1
038	Male	33	UG	3A	Yes	4	2
039	Male	26	JC	2B	Yes	4	1
040	Male	33	UG	3A	Yes	4	2
041	Male	34	JC	3A	Yes	4	2
042	Male	50	UG	3B	No	2.2	2
043	Male	44	UG	3A	Yes	4	1
044	Male	28	UG	2B	Yes	4	2
045	Male	50	TSS	3A	Yes	4	2
046	Male	50	SHS	3A	Yes	2.1	2
047	Male	40	SHS	3A	No	3	1
048	Male	30	JC	3A	Yes	2.2	1
049	Male	49	TSS	3A	Yes	4	1
050	Male	37	Masters	3A	Yes	4	2
051	Female	26	TSS	2B	Yes	3	2
052	Female	40	TSS	3A	Yes	3	1
053	Female	45	TSS	3B	Yes	3	1
054	Female	45	TSS	2B	Yes	3	2
055	Female	38	TSS	3A	No	3	2
056	Female	41	TSS	3A	Yes	4	1

057	Female	48	JHS	3A	Yes	3	1
058	Female	39	UG	3A	Yes	1	2
059	Female	40	SHS	3B	Yes	1	1
060	Female	27	SHS	2B	Yes	1	1
061	Female	30	UG	2B	Yes	2.1	2
062	Female	27	UG	2B	Yes	2.1	1
063	Female	35	SHS	2B	Yes	2.1	2
064	Female	34	UG	2B	Yes	2.1	1
065	Female	26	SHS	3A	Yes	2.1	1
066	Female	26	UG	2A	Yes	2.1	2
067	Female	33	UG	2A	Yes	2.1	2
068	Female	31	UG	2B	Yes	2.1	1
069	Female	26	UG	2A	Yes	2.1	2
070	Female	46	SHS	3A	Yes	1	1
071	Female	46	SHS	3A	Yes	1	2
072	Female	47	SHS	3A	No	1	1
073	Female	48	UG	2B	Yes	1	2
074	Female	50	UG	3A	Yes	1	2
075	Female	47	UG	3A	Yes	1	2
076	Female	41	UG	3B	Yes	1	1
077	Female	45	SHS	3B	Yes	1	2
078	Female	47	SHS	3A	No	1	1
079	Female	45	UG	2B	Yes	3	2
080	Female	43	SHS	3A	No	3	2
081	Female	46	JC	3A	Yes	3	2
082	Female	29	UG	2B	Yes	2.1	1
083	Female	49	JHS	3A	Yes	4	2
084	Female	38	UG	3A	Yes	3	1
085	Female	40	SHS	3A	Yes	3	1
086	Female	45	SHS	2B	No	3	2
087	Female	48	SHS	3A	Yes	3	1
088	Female	38	SHS	2B	Yes	3	2
089	Female	43	JHS	3A	Yes	4	2
090	Female	31	JC	2B	Yes	2.2	1
091	Female	35	UG	2B	Yes	2.2	1
092	Female	45	UG	3A	Yes	4	2
093	Female	45	JC	3A	Yes	4	1
094	Female	49	SHS	3A	Yes	4	1
095	Female	29	SHS	2B	Yes	4	1
096	Female	36	SHS	3A	Yes	3	1
097	Female	40	SHS	3A	Yes	3	2
098	Female	41	SHS	3A	Yes	3	1
099	Female	41	JC	3B	Yes	3	2
100	Female	39	SHS	3A	Yes	3	2

Appendix B: Details of Phonetic Transcriptions

Transcriptions of the Wu dialectal corpus were made using the Praat speech editor `http://www.fon.hum.uva.nl/praat/`; see Figure 18. The transcriptions are in four tiers:

1. HZ: Chinese character tier, giving orthographic transcriptions of the utterances. In addition, the following special symbols are used:
   (a) Non-Chinese string enclosed in {}: meaning English letters or English words;
   (b) Paralinguistic phenomena and noises: see Table 32.
2. PY: Transcription into the standard Pinyin symbol set, with tone. The special symbols described in 1 are used.
3. SEMI: The surface form semi-syllable tier. This is a fairly close phonetic transcription in terms of the initial-final (IF) set defined below. The observed tone of each syllable is attached to the Final of the corresponding syllable. Special symbols are:
   (a) Prefix "+": marks an IF as being inserted.
   (b) Prefix "-": marks an IF as being deleted.
   (c) Prefix "#": indicates a change in the following IF due to dialectal influence. Thus "#s" would mark a standard "sh" changed to "s".
   (d) Prefix "*": indicates a mispronunciation of the following IF.
   (e) Postfix "_v": indicating a voiced consonant due to the co-articulation, different from Wu Dialect specific voice consonants as /bb/ and /ff/.
   (f) Retroflexed "_r": indicating retroflexion.
   (g) Paralinguistic phenomena and noises are only transcribed when there is no speech sound. If the noise is mixed with speech, the speech is transcribed, and the noise is only marked with "?".
   (h) Mandarin 3rd Tone Sandhi.
4. MISC: Miscellaneous non-speech information is provided in this tier.

No	Phenomenon		Label	
			Start	End
1	Para-linguistic	Lengthening (拉长)	LE<	LE>
2		Breathing (吸气、喘气)	BR<	BR>
3		Laughing (笑)	LA<	LA>
4		Crying (哭)	CR<	CR>
5		Coughing (咳嗽)	CO<	CO>
6		Disfluency (不联贯)	DS<	DS >
7		Error (口误)	ER<	ER>
8		Silence (long) (长时间静音)	SI<	SI>
9		Murmur/Uncertain segment (不清发音)	UC<	UC>
10		Modal/Exclamation (语其次/感叹词)	MO<	MO>
11		Smack (咂嘴音)	SM<	SM>
12		Non-Chinese (非汉语)	NC<	NC>
13		Sniffle (吸鼻子)	SN<	SN>
14		Yawn (打哈欠)	YA<	YA>
15		Overlap (重叠发音)	OV<	OV>
16		Interjection (插话)	IN<	IN>
17		Deglutition (吞咽音)	DE<	DE>
18		Hawk (清嗓子)	HA<	HA>
19		Sneezes (打喷嚏)	SE<	SE>
20		Filled Pause (填充停顿)	FP<	FP>
21		Trill (颤音)	TR<	TR>
22		Whisper (耳语)	WH<	WH>
23	Non-linguistics	Noise (噪音)	NS<	NS>
24		Steady Noise (平稳噪音)	TN<	TN>
25		Beep (电话忙音)	BP<	BP>

Table 32: Labels for non-speech events.

Appendix C: OOV Rates on Test Speakers

Spkr	OOV Rates
053	0.057
046	0.038
099	0.026
035	0.023
059	0.017
067	0.017
064	0.014
047	0.013
066	0.013
016	0.0095
012	0.0089
011	0.0086
008	0.0077
043	0.0075
098	0.0075
032	0.0053
054	0.0043
076	0.0024
009	0.0014
061	0.0013

Appendix D: An Alternative Approach to Dialect Adaptation

(Written by Thomas Zheng and Jing Li)

Appendix E: Minimal Perplexity Word Segmentation

(Written by Philip Bramsen)

We present a summary of two approaches to improving language model perplexity on a Chinese corpus by altering the word segmentation. In the first approach, we create 'segmentation models' that mimic the language model probability distribution and try to directly extract the segmentation that is ideal for language modeling. The second approach we used the common Maximum Matching segmentation algorithm and modified the segmentation dictionary to include word concatenations likely to lower the LM perplexity.

15.1 Introduction.

Automatic Speech Recognition systems are typically built off of words. Syllables, letters, or other units more basic than words do not contain information about how their word affiliation affects their pronunciation. Even for Chinese, which has several thousand single character words, words are necessary units because word affiliation still dictates character pronunciation for the majority of words, which are multicharacter. Consequently, language models for speech recognition systems perform best if built off of words.

However, Chinese text lacks word boundaries. Therefore, in building language models for Chinese conversational speech recognition, we come face to face with the need for word segmentation. Various approaches to Chinese word segmentation have been proposed. Quite possibly, the quest for good word segmentation has spawned more research in Chinese speech and language engineering than any other need. For Chinese ASR tasks, the training corpus is first segmented and a language model is built on the segmented corpus. ASR word segmentation tasks require a fixed dictionary, to make character words affiliation clear. For a fixed dictionary, Maximum Matching Segmentation (section 3.2) works just about as well as any segmentation algorithm (Sproat, 2001).

Unfortunately, the segmentation quality of Maximum Matching is arguably dismal, if compared to the "segmentation" of languages which have word boundaries in written text. There are obvious flaws: It is inconsistent. It piles errors upon errors, because after an inaccurate segmentation decision it is often forced to incorrectly segment some a subsequent word into single characters.

There are many ways to approach segmenting Chinese Text. We should not assume beforehand that the "ideal" segmentation is delineated by a majority vote of Mandarin readers. Granted, a simple information retrieval task searching for words from a query might need segmentation that parallels the query, but other tasks are better off with other segmentations. For the purpose of a particular task, a reasonable goal for word segmentation is the hypothetical segmentation which yields the best performance for the task. What is the ideal segmentation for language modeling? Is there a segmentation of Chinese text that would yield the lowest perplexity language model?

We sought to guide Chinese word segmentation toward the best segmentation for the task of language modeling and, more specifically, language modeling for the task of speech recognition. Along the way, we took two routes:

First, we built segmentation models that paralleled the language model we were trying to construct; in short, we attempted to use corpus statistics paralleling language model statistics to directly pick the ideal segmentation. The theory behind this first approach is incomplete, and our results were limited, though informative.

Our second attempt to optimize the segmentation was more fruitful: we sought to tweak the lexicon of a common dictionary-based segmentation algorithm by adding concatenations of words that are likely to push the segmentation in a direction favorable to language modeling. For both approaches, our quality metric was the perplexity of language models built off of the segmentations.

15.2 Optimal Segmentation

Starting from the raw, unsegmented task we sought to directly produce the word boundaries that would bring about the lowest language model perplexity.

15.2.1 Rationale

Consider a string of unsegmented text:

```
A  B  C  D  E  F  G  H  I  J  K  L  M  N
```

Here each English character represents a Chinese character.

There may be a segmentation people generally agree with (although annotators rarely agree as well as we might hope (Sproat, 2001)):

```
A B | C | D E F | G H | I | J | K | L M | N
```

However, perhaps the best segmentation for a language model would be different:

```
A B | C | D | E | F | G H | I J K | L M N
```

Perhaps ‘DEF’ is rare and is better modeled if spelled out (Klakow, 1998). ‘IJK’ may be a phrase and friendlier to the language model as one word. In addition, language models are constructed from word bigrams at the minimum. The last character, ‘N’, might always follow ‘LM’; the bigram model may lose discriminative power if ‘LM’ is distinct from 1N’.

By considering the frequencies of all character sequences (within a reasonable size), we can pick the segmentation of any sentence that corresponds to the “lowest perplexity” segmentation. That is, we can segment any sentence to have the units that yield the lowest entropy segmentation according to the frequencies of all seen words in the training corpus. This might easily be extended to bigrams as it is in language modeling. The rationale behind our model is that the segmentation model might be used to directly derive the best segmentation possible for language modeling on the same corpus.

15.3 Model and Method

We will describe the segmentation model, the method to build it, and methods to apply the model to segmentation and language modeling.

Pseudowords We will describe our model in terms of pseudowords. A pseudoword is any string of characters that is no longer than some bound, k, on the length of the words in the segmentation model.

When counting pseudowords, there are few constraints, besides that the pseudowords must be present in the data. Because of the abundance of pseudowords in a text, we set a bound for the length of pseudowords to rein in the computational complexity. We chose k=4, which is acceptable because very few Chinese words longer. The sentence-start or sentence-end marker were also included as pseudowords. Pseudoword bigram frequencies are likely to be dependent on start or end of sentence location.

Consider the size of the set of pseudowords compared to the number of actual words. The average word occurring in conversation is roughly two characters long. In a string of ten unique Chinese characters which be reasonably seen as five dictionary words, there are 34 pseudowords. (The count ignores sentence start and end markers.) Extend the same rationale to a million character corpus, and there might be roughly 500,000 "real" tokens but 3.4 million pseudoword tokens. However, the bigger challenge is the explosion of word types. We observed 160,000 pseudoword types on our 1 million character training corpus when we allowed all pseudowords shorter than five characters. When we extended the scope of the model to bigrams of pseudowords, the 160,000 observed unigrams snowballed into over a million distinct types of pseudoword bigrams in the training corpus.

Inclusion of all pseudowords regardless of whether they are real words points at significant way that segmentation modeling differs from language modeling. For "thecow" a segmentation model allows: {"t","h",...,"w"; "th","he",..."ow"; "the","hec", ..."cow"; "theco",...; ...} Clearly, nonsensical miscreants are invading our data. This is likely to lead to challenging data scarcity problems. Smoothing techniques developed for language modeling may not be directly applicable to segmentation modeling because we are throwing in totally useless unigrams and bigrams that are often merely pieces of real words and often straddle real word boundaries.

Segmentation Model The segmentation model is the smoothed counts of the pseudowords and bigrams of pseudowords seen on a training corpus. As we used it, the applied segmentation model is when the segmentation model is used to chose the lowest perplexity segmentation of every sentence.

It is helpful to understand our segmentation model with respect to n-gram language models. By comparing them, the intuition behind the segmentation model becomes clearer and certain weaknesses/strengths come to light. With language modeling, the essence of model itself is contained in the smoothed, binned, or otherwise manipulated counts of unigrams and n-grams. This even extends to backoff models, where the smoothed word counts may go through an additional "filter" that substitutes other probabilities where there are unseen contexts or words. At the end, we can map any context (n-gram) to a specific probability.

The segmentation model is similar to the language model because it too is a probability distribution over its basic units. The segmentation model can be seen as a probability distribution that maps any context to a probability or cost. As with language modeling, we can have counts of unigrams or n-grams from the training corpus, smoothing, and backoff models. As with language modeling, model choices all affect the quality of the model. It is reasonable to measure the quality of a segmentation model by its perplexity, just as with language models.

However, unlike in language modeling, the probability distribution does not define a score or

cost for any particular sentence. The segmentation model describes the cost of many particular segmentations of any sentence.

Given the model, how does one select sentence segmentations? We chose to pick the lowest perplexity segmentation for each sentence. It is this decision which caused us significant problems. See section 15.3.1 for further discussion.

Building the Segmentation Model The algorithm first builds the segmentation model, then uses the segmentation model to determine the "best" segmentation of each sentence. This portion of the algorithm yields the "best" segmentations for each sentence.

1. Gather statistics over the entire corpus:
   (a) All pseudoword counts
   (b) All pseudoword bigram counts
   (c) Sentence start and end markers were included in the counts
2. Smooth the counts, apply language modeling techniques
3. Create FSMs
4. One FSM per sentence, encoding the costs of all segmentations.
5. Find the best paths through the FSMs

The algorithm is flexible in step 2, where we applied language modeling techniques such as smoothing or binning the bigrams. However, we did not focus on this. We considered these problems to be very similar to language modeling problems so we focused on the second part of the algorithm: making use of the best segmentations.

The FSMs (finite state machines) encodes up to four states for each possible word boundary in the sentence. Each of the four states represents the cost to reach that word boundary from a previous boundary, one state for each of the possible lengths for the word between the boundaries. Each arc represents the cost of choosing the word that reaches the state, given the previous word. In this way, bigrams are uniquely identified: the state the arc starts at is unique to the pseudoword that reached it, the node the arc reaches is unique to the pseudoword spanned by the arc.

Counting and shaping the model was done in Java. Smoothing was done with SGT.c a common, internet-available simple good turing smoothing program. The FSMs were formed by Java code and the best paths were determined using the AT&T FSM tools (GRM Library, 2004).

Applying the Segmentation Model Having derived the "best" segmentations for each sentence, we tried three different ways to make use of these results.

1. Collect all of the words occurring in the best paths segmentation. Use these words as the dictionary fed to Maximum Matching (or some other segmentation algorithm). Use Maximum Matching and the new dictionary to segment both the training and the test corpus. Build the language model off of this segmentation.

2. Use the segmentation model probabilities derived from the training corpus to find the best paths segmentations of the training and the test corpus. Do nothing more with the segmentation. Build the language model off of the training corpus and test it on the test corpus

3. Pick the dictionary for Maximum Matching by cross validation: Divide the training corpus in ten. Build ten segmentation models, one off of each 9/10ths of the training corpus. Use each 9/10th to find the best paths segmentations of the remaining tenth. Draw words for a fixed dictionary from the best paths segmentations by using the 1/10ths to vote on each word (e.g. if the word occurs on 3 of the best path segmentation 10ths, then include it in the Maximum Matching dictionary).

Baseline Segmentation. When experimenting with these three approaches, we considered the perplexity of the language model built off of the training and test corpus segmented by Maximum Matching to be our baseline. The training corpus was one million characters amounting to one hundred thousand utterances of conversational Chinese transcriptions. The two components of the training corpus were the Mandarin HUB5 corpus and HKUST's 100 hour conversational Chinese corpus. The test corpus consisted of several thousand utterances from test data collected by the Dialectical Chinese Speech Recognition team; these were transcripts of native Shanghainese speakers speaking Mandarin, which is a second language for them. A second baseline was the language model built off of the character segmentation of the training and test corpus. Perplexity measurements took unknown word scores into account and were normalized to the character counts of the test corpus.

Application of the Optimal Segmentation: Feeding a Maximum Matching Dictionary We used the 160,000 word dictionary derived from the best paths segmentation as the dictionary for a Maximum Matching segmentation of the training. While using this technique, we tweaked the segmentation model in several ways (binning the bigrams into frequent-enough and rare, applying smoothing, throwing out low counts). At best, this yielded a language model perplexity almost double that of the baseline and 10%-15% worse than the character bigram language model. The motivation for this approach is that giving Maximum Matching a dictionary pulled from the best paths segmentation might be an improvement over handing it the baseline dictionary. An examination of the best paths segmentation dictionary and the training corpus reveals Maximum Matching chose 70,000 of the 160,000 words available to it. This is roughly five times the vocabulary that the language model built on the baseline segmentation had to face. In short, the language model was facing tremendous data sparsity problems.

Application of the Optimal Segmentation: Using it Directly When we built a language model off the the best paths segmentations of the training and test set the results were still unfavorable. The language model perplexity was six times that of the baseline. The segmented training corpus had over 150,000 "words" present, which is ten times the vocabulary that language model built on the baseline segmentation had to face. This means the language model was facing tremendous data sparsity problems.

Application of the Optimal Segmentation: Cross Validation Word Voting: We used cross validation to determine the best words to add to the Maximum Matching segmentation algorithm's dictionary. This yielded the best results, the language model perplexity pushed down to nearly that of the character segmented baseline. This, however, is still significantly worse than the baseline dictionary segmentation.

15.3.1 Problems

The project has problems with the theoretical underpinnings of its intuitions as well as problems with the implementation.

First, we did not prove that the best path segmentations we end up with are actually what we claim they are: best paths. The mathematics upon which language modeling stands do not directly apply because we are not using word counts, only pseudoword counts. Pseudoword counts are free to repeatedly count the same characters towards multiple pseudowords. Equivalently: the counts may be of overlapping pseudowords. This brings into question the probability distribution we get; just what, exactly, is it?

Second, best path segmentation does not necessarily yield the best path segmentations. The cost of each pseudoword bigram is based on its frequency. However, all cost related issues being equal, the total cost of a segmentation consisting only of four character words is roughly one fourth that of choosing an only single characters segmentation. This problem sometimes lead to preposterous segmentations consisting of primarily long words. We tried factoring the length of the words into the bigram cost (by their length, the square of their length, etc.), but this is an ad hoc measure without justification. In fact, the frequencies of longer words already penalize their length (they are invariably more rare). This issue exposes a weakness of the segmentation model.

Third, best paths segmentation has no global-to-corpus constraints. Best path selection may choose different words for sentences A and B. The segmentation model was built for the entire training corpus and these pseudo words may indeed the best according to the whole corpus model. However, the context seen in any particular character sequence strongly determines what portion of the segmentation model is seen by the best path segmenting step of the the algorithm that makes use of the segmentation algorithm. This means that different sentences may segment using essentially different segmentation models. The enormous size of the segmentation model and the relative distance the optimal segmentation of each sentence may be from every other sentence means that the optimal segmentation may little bearing on the optimal segmentation for language modeling.

15.4 A Heuristic Iterative Approach to Segmentation Dictionary Optimization

During the 2004 CLSP Summer Workshop, some of us on the Chinese Dialectal Speech Recognition Team noted that the perplexity (Manning and Schütze, 2003) of our bigram language model depended on the dictionary used for segmentation.

Figure 15: Mismatches during acoustic modeling

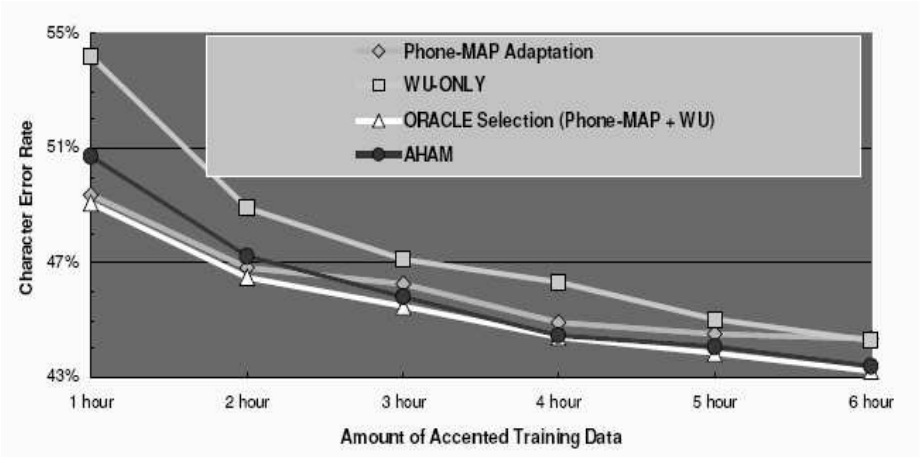

Figure 16: Comparison of various acoustic models with varied account of accented training data

	Perplexity
Characters only	88.1
Baseline Dictionary + Maximum Matching	69.2

Table 33: Comparison of the perplexity of the LM built off of two segmentations.

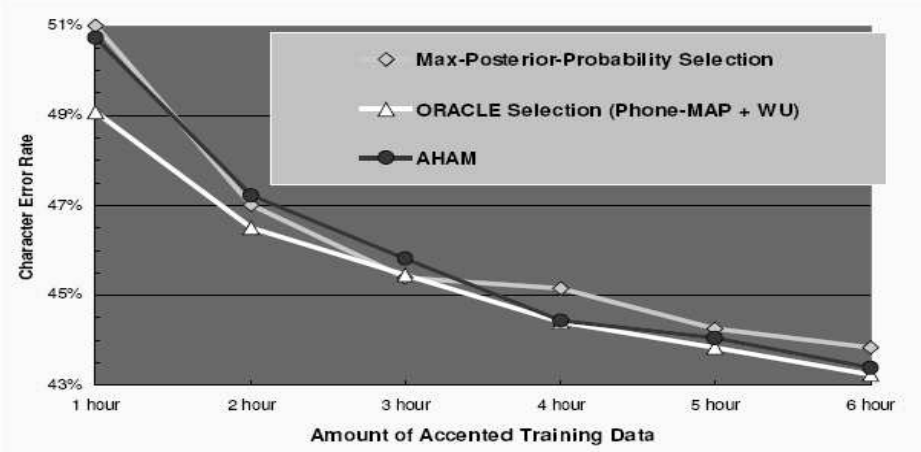

Figure 17: Comparison between AHAM and acoustic model combination using Max-Posterior-Probability Selection.

Building a language model on characters alone is reasonable; we found the average word length for our conversational Chinese training corpus to be a little less than two characters long. The second language model listed was constructed by segmenting our training corpus of one million characters of conversational Chinese with our baseline dictionary of 50K words obtained from Tsinghua University in Beijing. To segment we used Maximum Matching, which has been shown to work about as well as any other algorithm for segmenting Chinese text.

In addition to the two models above, we noted that adding even a small number of words to the segmentation dictionary noticeably changed the perplexity of the resulting language model.

This raises a natural question: Is there a minimum perplexity segmentation? More specifically, is there a minimum perplexity segmentation dictionary?

15.5 Experiment Construction

Nearly all our language models were bigram models built with the AT&T FSM and GRM tools (GRM Library, 2004). We also used the SRI language modeling toolkit to build variable length n-gram models (SRI Language Modeling Toolkit, 2004). For the most part, we depended on Katz-Backoffs and Good-Turing smoothing.

The training corpus was one million characters amounting to one hundred thousand utterances of conversational Chinese transcriptions. The two large components of the training corpus were the Mandarin HUB5 corpus and HKUST's 100 hour conversational Chinese corpus. Another several thousand utterances were from test data collected by the Dialectical Chinese Speech Recognition team; these were transcripts of native Shanghainese speakers speaking Mandarin, which is a second language for them. Perplexity measurements took unknown word scores into account and were normalized to the character count of the test corpus.

We focused on bigram word models because: 1) The word models outperformed the character based models. 2) The bigram models beat out the trigram models.

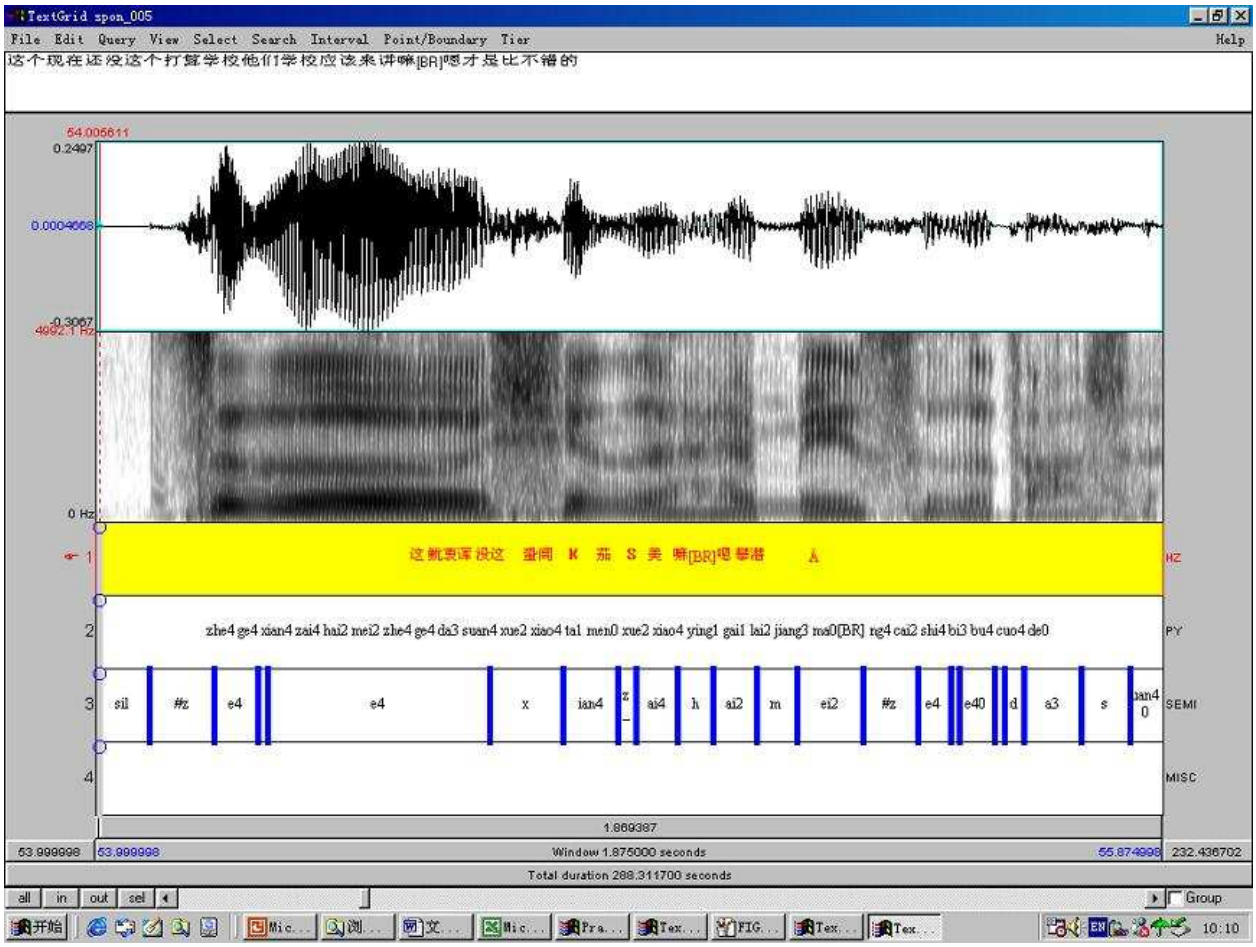

Figure 18: Labeling example with the Praat editor.

The language modeling task was particularly challenging for two reasons:

(a) There is very little usable transcribed conversational Chinese publicly available. At one point, we were desperate enough to search the web for transcriptions of Chinese soap operas.

(b) The training corpus is not an ideal match for the test corpus. Recording situations and topics of conversation were not the same. The year and location of the recordings also affected the content. We did not have the luxury of ensuring these criteria were matched. However, this is a common problem faced by Chinese ASR today.

We did investigate other forms of smoothing, and found similar results. At the very end of the workshop, Dietrich Klakow, a member of another team at the 2004 CLSP Summer Workshop, used his language modeling tools and produced models on our data where the trigram edged out the bigram model.

15.6 Maximum Matching

Maximum Matching (Manning and Schütze, 2003) is one of the simplest segmentation algorithms: Given a string of characters and a lexicon, find the longest word that both starts at the left edge of the string and is in the lexicon. Accept the right edge of that word as a word boundary and continue segmenting the remainder of the string.

The intuition behind Maximum Matching is that strings of characters that are words are unlikely to occur by chance. Taking 5000 characters as a reasonably sized Chinese character set, there are 25 million two character combinations, and 125 billion three character combinations. Add to this the fact that a typical ASR dictionary is constrained to about 50,000 lexical entries, and we can see the probabilities work out in favor of Maximum Matching's success.

For example, suppose English lost its spaces:

```
theirgardenisovergrown
```

Suppose that while losing all our spaces, we managed to retain a dictionary:

```
den, gar, garden, the, their...
```

Maximum Matching would hopefully pick out the correct sentence sentence:

```
their garden is overgrown
```

At the left end of the sentence, their is the longest word. After their, garden is the longest word, and so on.

Maximum Matching does not help resolve segmentation ambiguities, as the following example illustrates:

```
Heateraresteakstrips
```

Which Maximum Matching segments into:

```
heater are steaks trips
```

In this rather contrived example, we see that Maximum Matching incorrectly parses the original sentence, which was "He ate rare steak strips." Instead of dealing with ambiguities, Maximum Matching opts to ignore them, choosing to pick whatever long words it finds first. It only takes one error to botch a sentence or force a portion of the sentence into single characters. Nevertheless, it turns out that in the real world the algorithm works about as well as any other segmentation approach when a fixed dictionary is required.

15.7 Dictionary Optimization - Motivation and Concept

In an effort to reduce the language model perplexity, we attempted to build a better segmentation dictionary than the baseline dictionary we had. We did this by concatenating bigrams of words that tended to occur together to form new dictionary entries.

More precisely, the concatenated bigram w_1w_2 was added if the following criteria were met:

$$\begin{array}{l} C(w_1, w_2) > T_{C(w1,w2)} \\ \frac{C(w_1,w_2)}{C(w_1)} > T_{w_1\%} \\ \frac{C(w_1,w_2)}{C(w_2)} > T_{w_2\%} \end{array}$$

Here $C(.)$ means count-of, $T_{C(w1,w2)}$ is the threshold for how frequently a bigram must occur to be considered, $T_{w_1}\%$ and $T_{w_2}\%$ are thresholds for the percentage of the occurrences of the individual words accounted for by their appearance in the bigram. The first threshold acts to ensure the bigram is frequent enough for the information to be worthwhile. The other thresholds act as a measure of the coocurrance of the words.

Note that the measure bears resemblance to mutual information, which has generally respected value as a measure of cooccurrence. It also bears resemblance to work detailed later in this section where simple counts outperformed more complicated measures like mutual information (Klakow, 1998).

$$\text{MI} = C(w_1, w_2) log\left(\frac{C(w_1,w_2)}{C(w_1)C(w_2)}\right)$$

Some similar approaches have attempted to minimize entropy or the description length of the training corpus (e.g. (de Marcken, 1996)). However, our algorithm (described shortly), attempts to minimize perplexity directly using experimental exploration.

Many of the words we found with this technique were reasonable lexical units or common phrases.

说清楚 “say clearly”
不容易 “not easy”
不知道 “not know”
果子狸 “masked palm civet”
手机 “cell phone”
电子产品 “electronic products”
在学校 “in school”

These bigrams were generated with the thresholds $TC(w1, w2) = 5$ and $Tw1\% = Tw2\% = 0.5$. “Say clearly,” “not easy,” “not know,” and “in school” can all be seen as phrases. “Cell

phone" was actually missing from the baseline dictionary. "Electronic products" is a reasonable lexical entry. "Masked Palm Civet," which goes by the Latin name *Paguna larvata*, is a small 20-inch long mammal with orange, red, or yellow fur, and a stripe down its face. The Mandarin for "Masked Palm Civet" is also the name of a fruit company, which is more likely to have been the topic of the conversation. These results suggest the words found are reasonable.

15.8 Algorithm

The algorithm we used to make use of dictionary optimization ran as follows:

(a) Start with a segmentation, determine new words from training corpus

(b) Re-segment with the new dictionary

(c) Build a language model

(d) Use the LM perplexity as the score of the new segmentation

First, a segmentation starting point is needed. This could be characters, pre-segmented text, or even randomly segmented text. Statistics for the words in the segmented training corpus are collected and every bigram is evaluated by the three criteria mentioned above. Bigrams exceeding all three thresholds are added to the segmentation dictionary.

The new dictionary is used to re-segment the training and test corpus. To evaluate the new dictionary and the segmentation, a language model is built on the segmented text and the language model perplexity is calculated.

The newly segmented text contains new words. Iterating through the algorithm again will unearth new bigrams to contribute to the dictionary. There are two reasons for this: First, bigrams found in one iteration can be components of bigrams in a later iteration. Second, Maximum Matching jostles the segmentation around. The jostling reveals more likely candidates for concatenations, but also reminds us there is no guarantee the words will be noticed by Maximum Matching.

The iterations may continue until language model perplexity ceases to improve.

Perceiving the Parameter Space of the Problem: Previous work on language model optimization via word concatenation apparently has not considered the problem as a parameter space. In our experiments, we discovered that the best results came from allowing the thresholds for word concatenation to change at each iteration.

The parameter space can be seen as a tree. The root node is the original segmentation of the text. One layer down, each node represents a different set of thresholds for word concatenation. There is no reason to believe that the tree must be walked by making exactly the same decision at each node. In fact, we found through experimentation that we could get better results if we looked at the sister nodes at each level and examined all "nearby" nieces and nephews of the best performing node. Another successful experiment involved successive

	Perplexity
Baseline Dictionary	69.19
First Iteration Best (C=5, 3%, 3%)	68.30
Second Iteration Best (C=5, 4%, 4%) + (C=5, 4%, 4%)	68.09
Character-based variable-length (9)-gram model	77.29

Table 34: Perplexity results starting with baseline dictionary.

	Perplexity
Baseline Character Dictionary	88.05
First Iteration Best (C=5, 2%, 2%)	75.03
Second Iteration Best (C=5, 3%, 3%) + (C=5, 3%, 3%)	73.67
Character-based variable-length (9)-gram model	77.29

Table 35: Perplexity results starting with character dictionary.

relaxations of the thresholds at each iteration. This ensured that the more certain words were added first.

In our implementation, the algorithm was heavily supervised by manual decisions at each iteration. We did this to explore the utility of the algorithm in the short time we had at the end of the workshop. However, exploration of the parameter space could be automated. Automation could be quite difficult without more evidence for particular approaches to searching the parameter space. At the minimum however, frameworks such as successively relaxing the thresholds or restricted breadth first search might be useful. Continued research may reveal heuristics to enhance automation.

15.9 Experiments

In the week we worked on this project, we ran several experiments. The early ones gave us some intuitions about the value of the technique; the last few experiments are reported here.

15.9.1 Experiments - Iterating from Baseline Dictionary Segmentation

We started with text segmented by the baseline dictionary and Maximum Matching. The threshold $TC(w1, w2)$ was varied between 2 and 20, and the thresholds Tw1% and Tw2% were varied between 1% and 90%. Due to time constraints, we did not manipulate the latter two independently.

15.9.2 Experiments - Iterating from the Character Segmented Data

Starting from unsegmented text-text segmented into characters-we repeated the above experiment.

15.9.3 Testing the LM with the ASR System

We rescored the lattice of with the lowest perplexity language model we found. This resulted in a small (0.1%) reduction in the character error rate of the total system. However, this is not entirely fair since the language model was selected by using the test corpus used for scoring decoding. On the other hand, word affiliation pronunciations were not added to the ASR system for the new words. We believe further reductions in character error rate are possible.

15.10 Related Work – Improving Language Models with Multiwords

Previous work exists on optimizing language models by concatenating words (Klakow, 1998; Ries, Buo, and Waibel, 1996). The motivation for this work is recognition of the fact that the ideal units for language modeling are not necessarily the words given by typical written text. A language model built on the text of all CLSP Summer Workshop Proposals might be improved if certain common phrases were concatenated into words or rare words were broken into regular components. In speech recognition, rare words might be best understood by looking at their syllables independently. Therefore, much of the research into concatenating words also investigates dividing words.

Another motivation is to mimic text-compression techniques. One approach to text compression is to strive to have symbols for the compressed form all of identical length and basic units of nearly identical frequency. Finding "multiwords" or doing "corpora mapping" means finding common phrases and making them words in the text, but generally only when this creates better word and or bigram frequencies.

It appears the technique of concatenating words became more successful when minimizing the language model perplexity directly became the goal. Typical approaches to LM optimization run as follows:

(a) Acquire corpus statistics

(b) Rank bigrams according to cooccurrence metric

(c) Select a set of words to redefine as phrases in the corpus

(d) Restructure the corpus: map it according to the new phrases found, possible breaking apart other concatenations found in previous iterations or breaking a par

(e) Return to step (1)

Some cooccurrence metrics are (this list is taken pretty much directly from (Klakow, 1998; Ries, Buo, and Waibel, 1996):

(a) mutual information

(b) bigram frequency

(c) backward bigram: $p(w1|w2)$

(d) backward perplexity: $log(p(w1|w2))$ measures the change in the unigram likelihood measure of the training data (Klakow, 1998)

Variations on the techniques include breaking the unknown words into their spellings or syllables. Some variations suggest intentionally considering the word boundary tendencies of the language on which the language modeling is being done (Larson et al., 2000).

15.11 Analysis

In this section I make the case for why dictionary optimization in Chinese segmentation is different from the creation of multiwords in language model optimization. I will also attempt to argue for a new approach to searching for multiwords and split-able words in both tasks.

15.11.1 Successes and Limitations of Previous Approaches

Research into the use of multiwords for language model optimization can be summarized as follows: For bigram models, adding multiwords can reduce the size and perplexity of the model. For trigram and larger models, the advantage afforded by multiwords weakens becomes insignificant as the size of the training corpus grows. Of course, for large corpora, the trigram significantly outperforms the bigram, whether or not the bigram has multiwords.

15.11.2 Why Chinese Segmentation is Different:

Chinese Speech and Language Processing is different from optimizing language model perplexity for "presegmented" languages for two reasons.

First, Chinese text is originally unsegmented, therefore modifying the dictionary by word concatenation and segmenting with the new dictionary is not directly comparable to concatenating words found in English text. In unsegmented Chinese text, information about words is simply not present. Information about the location of words is valuable for language modeling, even though commonly accepted word boundaries not necessarily the truly ideal basic units. This is confirmed by the fact that even if Chinese text is segmented with Maximum Matching, the word based language model built on the segmented text outperforms a model trained on character segmented text.

However, fixed dictionary segmentation of Chinese is clearly clumsy, often giving bizarre segmentations that have obvious inaccuracies. Given ASR's need for a fixed dictionary, we are forced to seek clever ways to optimize the segmentation while continuing to segment with a dictionary.

This strongly suggests that lessons from the successes with multiwords be brought into play in the Chinese text segmentation subfield of NLP. Using word concatenation to iteratively improve the segmentation dictionary or directly improve the segmentation after an algorithm like Maximum Matching might significantly improve Chinese language modeling.

Second, Chinese colloquial speech recognition research faces the challenge of limited data. This means that even given the acknowledged limitations of multiwords for language model optimization for languages with word boundaries, they are likely to be useful in the near future for Chinese ASR. Word concatenation techniques, combined with techniques for breaking

apart unknown words, are useful even for trigram language models built on the 40 million word *Wall Street Journal* Corpus, Chinese ASR has much less data at its disposal.

This second observation argues that determining good ways to accomplish useful phrase concatenation and word splitting techniques to layer on top of word segmentation would benefit Chinese ASR.

These observations suggest that techniques to use optimal dictionaries and post-segmentation multiwords search during Chinese text segmentation would be very useful contributions to speech and language engineering.

15.11.3 Better Approaches to Finding Multiwords

Making language model perplexity the guide for selecting multiwords is more successful than finding multiwords purely by picking words dependant on word cooccurrence scores. However, this means that under this best metric there is no clear way to be certain the multiwords chosen at one iteration will be best for language model optimization in the next iteration.

Even in our few shallow experiments, the best parameters for the first iteration were rarely the best parameters for the later iterations. Moreover, taking less desirable segmentations early on actually led to better segmentations later.

This suggests that the conception of a parameter space introduced in section 2.5 should be exploited to enhance the search for better segmentations.

At this point we can ask what might seem a strange question: Given that we are making language model perplexity our goal, might we use any of several cooccurrence measures to suggest word concatenations at any iteration? Why compare them independently, as is normally done? Why not use several at once? The mathematics becomes ridiculous, but already several purely heuristic approaches that amount to guess and check have proved their worth. Perhaps the different heuristics yield complementary multiword contributions.

Appendix F. Spontaneous Wu-Dialectal Chinese Speech Recognition

16.1 Introduction (Project goal)

There are eight major dialectal regions in addition to Mandarin (Northern China) in China, including Wu (Southern Jiangsu, Zhejiang, and Shanghai), Yue (Guangdong, Hong Kong, Nanning Guangxi), Min (Fujian, Shantou Guangdong, Haikou Hainan, Taipei Taiwan), Hakka (Meixian Guangdong, Hsin-chu Taiwan), Xiang (Hunan), Gan (Jiangxi), Hui (Anhui), and Jin (Shanxi). These dialects can be further divided into more than 40 sub-categories. Although the Chinese dialects share a written language and standard Chinese (Putonghua, or PTH) is widely spoken in most regions, speech is still strongly influenced by the native dialects. This great linguistic diversity poses problems for automatic speech and language technology. Automatic speech recognition relies to a great extent on the consistent pronunciation and usage of words within a language. In Chinese, word usage, pronunciation, and syntax and grammar vary depending on the speaker's dialect. As a result speech recognition systems constructed to process standard Chinese perform poorly for the great majority of the population.

There are many research works focusing on pronunciation variation modeling and accent speech recognition. [Strik 1999] gives a good overview about the pronunciation variations and [Huang 2001] shows the recent research work on accent Chinese speech recognition.

The goal of our summer project is to develop a general framework to model phonetic, lexical, and pronunciation variability in dialectal Chinese automatic speech recognition tasks. The baseline system is a standard Chinese recognizer. The goal of our research is to find suitable methods that employ dialect-related knowledge and training data (in relatively small quantities) to modify the baseline system to obtain a dialectal Chinese recognizer for the specific dialect of interest. For practical reasons during the summer, we focused on one specific dialect, the Wu dialect. And we referred to the Chinese influenced by the native Wu dialect as Wu Dialectal Chinese (or WDC). However the techniques we intend to develop should be broadly applicable.

16.2 Speech Database Collection & Division

16.2.1 Standard Chinese Speech Corpus for Baseline System

The Mandarin Broadcast News (MBN), a read style standard Chinese speech corpus with Chinese initial/final (IF) transcriptions provided by JHU, was used to train the acoustic model for Standard Chinese Recognizer, e.g. our baseline system. MBN contains about 30 hours' high quality wideband speech. Till now, it is the best choice for us to build our baseline system, though the channel, speaking style and accent are totally different.

16.2.2 Wu-Dialectal Chinese (WDC) Speech Corpus

The Shanghai dialect is a sub-dialect belonging to the Wu dialect, one of the main Chinese dialects, and it is often called "standard Wu dialect". So, the Shanghai Putonghua is a good choice to study the common effect of the Wu sub-dialects. The Wu-Dialectal Chinese corpus was carefully designed, collected and transcribed before this workshop. It mainly contains two parts,

the di-IF balanced read speech part and the spontaneous speech part [Li 2003], and the spontaneous speech part has been chosen for use in this summer workshop.

For the spontaneous speech corpus, unlike designing prompting texts for read speech, some topics are pre-defined. Each speaker will be asked to select a topic to talk with another person when collecting the speech data. In this way, the collected speech may be relatively natural and more spoken language phenomena can be covered. In this paper, 5 main topics (Sports, Politics & Economy, Entertainment, Lifestyle, Technologies) and some corresponding sub-topics are designed.

Totally, the spontaneous WDC speech corpus contains 100 Shanghai native speakers' speech data, including 50 males and 50 females, and each speaker has about 5-6 minutes' speech data, but only about 3 minutes has the IF transcriptions. Four-layer transcriptions has been made manually for this corpus, including the Chinese character layer, the canonical Chinese pinyin layer, the surface form Chinese IF layer and the miscellaneous layer.

16.2.3 WDC Speech Corpus Division

Twenty speakers' data (about 1.7 hours), containing ten strongly accented (or More Accent, MA) and ten weakly accented (or More Standard, MS) speakers, are chosen as the *Test* set, while the rest 80 speakers' data (about 6.3 hours) as the Developing Train (*DevTrain*) set. Then, a small data set called Developing Test (*DevTest*) set, containing twenty speakers' data is picked up from the *DevTrain* set. The details of the *Test* set & *DevTest* set are shown in Table 1 and Table 3.

Table 1. Test Set Speaker Info (20 speakers)

Speaker-ID	Gender	Accent	PTH Level	Speaker-ID	Gender	Accent	PTH Level
032	m	MA	3B	008	m	MS	2A
035	m	MA	3B	009	m	MS	2B
043	m	MA	3A	011	m	MS	2B
046	m	MA	3A	012	m	MS	2B
047	m	MA	3A	016	m	MS	2B
053	f	MA	3B	054	f	MS	2B
059	f	MA	3B	061	f	MS	2B
076	f	MA	3B	064	f	MS	2B
098	f	MA	3A	066	f	MS	2A
099	f	MA	3B	067	f	MS	2A

The PTH Level is some different from the Accent Extent, it considers many aspects including accent, fluency, and so on, and is judged by experts. In our experiments, they are consistent to the *Test* set. The *Test* set can also be separated into different sub-sets corresponding to Age, Gender, Education Level, and so on. Table 2 shows the details of the separations.

Table 2. Groups of Speaker in Test Set

Sub-Set		Speaker-ID
Age	**AO (Age Old, >40)**	043,046,047,053,054,059,076,098,099
	AY (Age Young)	008,009,011,012,016,032,035,061,064,066,067
Gender	**GM (Gender Male)**	008,009,011,012,016,032,035,043,046,047
	GF (Gender Female)	053,054,059,061,064,066,067,076,098,099
Education	**EH (Education High)**	008,009,011,012,016,032,035,043,061,064,066,067,076,099
	EL (Education Low)	046,047,053,054,059,098
Accent	**MS (More Standard)**	008,009,011,012,016,054,061,064,066,067
	MA (More Accent)	032,035,043,046,047,053,059,076,098,099

The detailed information of the *DevTest* set used in our experiments is shown in Table 3. The speech data and the Pinyin and IF level transcriptions have been used to adapt the acoustic model and the lexicon.

Table 3. DevTest Set Speaker Info (20 speakers)

Speaker-ID	Gender	PTH Level	Speaker-ID	Gender	PTH Level
001	m	3A	055	f	3A
003	m	3A	060	f	2B
004	m	3A	063	f	2B
013	m	3A	065	f	3A
017	m	3A	069	f	2A
018	m	2B	081	f	3A
026	m	3A	085	f	3A
036	m	3A	090	f	2B
037	m	3A	094	f	3A
050	m	3A	095	f	2B

17 Proposed Framework for WDC recognizer

As mentioned earlier, there are too many different dialects in China, so, it is too difficult and expensive to design and collect a speech corpus big enough to retrain a new acoustic model and build a dialect specific recognizer for each dialect. We are trying to use the WDC data as few as possible so as to verify the feasibility of the idea of developing a framework for deriving a dialectal Chinese recognizer from an existing PTH recognizer with less effort. And we hope the techniques can be easily extended to build other dialectal Chinese speech recognizer.

We want to catch the following varieties to change the PTH recognizer to obtain a dialect specific recognizer:

- Phonetic variability
- Lexical variability
- Pronunciation variability

Our research focused on how to extract the dialect-related knowledge, such as the IF mapping, syllable mapping, cross-dialect synonyms, and so on, but use only a small quantity of or

even lacking adaptation data to improve the recognizer.

During pronunciation modeling, the adaptation of the acoustic model and the lexicon are considered. MLLR is used to adapt the acoustic model, while the extracted IF/Syllable mapping rules are used to refine the lexicon. Different from the traditional adaptation methods, we use the surface-form IF transcriptions to supervise the adaptation instead of the base-form transcriptions. Indeed, these IFs really sound like the surface-form, and are mostly different from the base-form, e.g. phone changes often happen. The base-form based adaptation might be more efficient to modify the parameters so as to model the real thing the speaker wants to speak. However, the model could most possibly be more scattered and would be more confused with other model, such as *sh* and *s*. Essentially, it is combining two or more totally different units together in most cases. This would not happen when using the surface-form based adaptation, but we should give more precise pronunciation lexicon to get the correct recognition results.

In our experiments, we use only the small set, i.e. the *DevTest* set, to improve the recognizer. It contains about 1 hour speech data, including 20 Shanghai speakers' speech and 3 minutes for each speaker.

18 Baseline System

The baseline system is the PTH recognizer. The HTK3.2.1 tool [Young 2002] are used to train and adapt the acoustic model, while the SRILM tools are used to get the word Bi-gram language model. Context-Dependent IFs are chosen to be the speech recognition units (SRUs). Each unit is modeled using left-to-right non-skip 3 or 4 states continuous HMM. Each state has 14 Gaussian mixtures. 39-dimensionaml MFCC coefficients with Cepstral Mean Normalization (CMN) are used as the features. MBN database are used to train the acoustic model. Language model is built on HKUST 100 hour CTS data, plus Hub5, plus Wu-Dialectal Training Data Transcriptions. Totally, 72.17% of Character error rate (CER) is obtained when directly using this PTH recognizer to test the *Test* set. The results are extremely low mainly because of different channel, speaking style and accent.

19 Dialect-Related Knowledge Extraction & Pronunciation Modeling

Our research focuses on acoustic adaptation, extraction and application of the dialect-related knowledge to multi-pronunciation lexicon, which includes:

- Context-Free PTH-IF mapping rules (Lexicon)
- Context-Free WDC-IF mapping rules (AM+Lexicon)
- Syllable-Dependent WDC-IF mapping rules (AM+Lexicon)
- Multi-Pronunciation Expansion (MPE) based on unigram probability (AM+Lexicon)
- Perform rank-based AM rescoring (Future work)

19.1 Context-Free PTH-IF Mapping based PM

It is easy to find that the dialect specific pronunciation variations are mainly occurred in IF layer for Wu-dialectal Chinese. So, we concentrate on IF level pronunciation modeling. Some

common and obvious IF mapping rules provided by linguistic experts are as follows:

- (zh, z) (z, zh)
- (ch, c) (c, ch)
- (sh, s) (s, sh)
- (eng, en) (en, eng)
- (ing, in) (in, ing)
- (r, l)

We can also see the above mappings in Table 4, a sub-set of the Context-Free PTH-IF Mapping Rules (Table 5). It is obvious that expert's knowledge based mapping rules have been covered by the rules extracted from the manual transcriptions. The probability (Prob.) represents the proportion of the phone change from a specific base-form to a specific surface-form.

Table 4. Expert's Knowledge Related Context-Free PTH-IF Mapping Rules Extracted from *DevTest* (Sub-set of Table 5.)

base form	surface form	Prob.(%)
zh	z	73.21
ch	c	81.63
sh	s	54.17
s	sh	10.00
r	l	22.05
eng	en	29.67
en	eng	17.63
ing	in	10.77
in	ing	35.12
iii	ii	50.69
ii	iii	16.67

The Context-Free PTH-IF mapping rules generated from *DevTest* set are shown in Table 5. The first column is the base-form IFs, e.g. the PTH-IFs, and the second column is the surface-form IFs, generated from the original manual IF layer transcriptions. The right-most column is the phone change probability, only the changes exceeding a certain threshold, e.g. the total counts and the probability, can be reserved. Self-mapping should be reserved for presenting the standard pronunciations.

Then the Context-Free PTH-IF mapping rules are used to generate the multi-pronunciation lexicon. The acoustic model is not changed in this experiment. an absolute CER reduction of 0.5% is obtained after using PTH-IF mapping rules.

Table 5. Context-Free PTH-IF Mapping Rules

Base form	Surface form	Prob. (%)	Base form	Surface form	Prob. (%)	Base form	Surface form	Prob. (%)
a	a	88.57	iao	iao	87.69	r	l	22.05

ai	ai	91.29	iao	e	5.03	s	s	84.38
an	an	91.73	ie	ie	94.16	s	sh	10.00
ang	ang	91.58	ii	ii	83.33	sh	s	54.17
ao	ao	94.09	iii	ii	50.69	sh	sh	32.18
b	b	91.75	iii	iii	35.10	t	t	90.64
c	c	85.83	in	in	57.85	u	u	92.72
ch	c	81.63	in	ing	35.12	ua	ua	88.98
ch	ch	11.56	ing	ing	80.33	uai	uai	78.72
d	d	91.37	ing	in	10.77	uan	uan	91.40
e	e	91.99	iong	iong	94.00	uang	uang	86.21
ei	ei	94.12	iou	iou	90.44	uei	uei	92.43
en	en	73.90	j	j	91.38	uen	uen	73.33
en	eng	17.63	k	k	94.55	uo	uo	92.90
eng	eng	64.23	l	l	93.32	v	v	90.20
eng	en	29.67	m	m	96.19	van	van	84.47
er	er	96.36	n	n	93.85	ve	ve	90.08
f	f	90.69	ng	ng	93.02	vn	vn	74.29
g	g	91.39	o	o	84.00	x	x	92.22
h	h	86.23	ong	ong	91.20	z	z	86.50
i	i	92.17	ou	ou	84.20	zh	z	73.21
ia	ia	93.94	p	p	93.99	zh	zh	16.60
ian	ian	94.01	q	q	94.53			
iang	iang	89.66	r	r	65.75			

The recognition results will be shown in section 5.4

19.2 Context-Free WDC-IF Mapping based PM

What we discussed in the previous section is the in-set PTH-IF mappings, that is to say, each mapping is from an PTH IF to another PTH-IF. As a matter of fact, there are some Wu dialect Chinese specific IFs (WDS-IF), such as “io^”, occurring in the speech data. Totally, 13 WDS-IFs are selected corresponding to their frequencies in the *DevTest* set. The WDS-IFs and the PTH-IFs form the WDC-IF set. The context-free mapping rules from PTH-IF to WDC-IF extracted from *DevTest* set are shown in Table 6. The rows with gray background are extended new WDC-IFs.

Table 6. Context-Free WDC-IF Mapping Rules

PTH-IF	WDC-IF	Times	Prob(%)	Is In-Set Mapping?
ai	e>	109	21.29	No
ao	o^	138	47.10	No
c	ch	4	6.45	Yes
ch	c	96	78.40	Yes
en	eng	51	15.64	Yes
eng	en	39	21.91	Yes

er	eer	16	66.67	No
iao	io^	152	51.70	No
ie	ie<	38	22.62	No
iii	ii	262	46.62	Yes
in	ing	72	42.35	Yes
ing	in	8	3.77	Yes
iong	ioong	13	46.43	No
iou	iuu	164	36.12	No
iou	ieu	152	33.48	No
n	ni	87	18.32	No
o	e	2	16.67	Yes
ong	oong	79	41.15	No
ou	eu	164	68.91	No
r	l	46	34.59	Yes
sh	s	494	58.84	Yes
ve	voe	55	55.00	No
ve	voong	4	21.05	No
zh	z	230	64.77	Yes

- 11 mappings from PTH-IF to PTH-IF (excluding the self-mapping rules)
- 13 mappings from PTH-IF to non-PTH-IF, with 13 new IFs introduced as follows, which can be adapted from their corresponding PTH-IF using the *DevTest* Set.

  e>, o^, eer, io^, ie<, ioong, iuu, ieu, ni, oong, eu, voe, voong

19.3 Syllable-Dependent WDC-IF Mapping based PM

We are discussing context-dependent WDC-IF mappings (syllable-dependent IF mapping) in this section. Indeed, in previous sections, some context-dependent mapping rules have been considered, for instance, when Initial "*sh*" in Syllable "*shi*" is changed to "*s*", the Final should not be "*ii*", it should be changed to "*iii*", because the "*i*", "*ii*" and "*iii*" are syllable-dependent. We hope to obtain more precise mapping rules after introducing the context influences.

Syllable-dependent WDC-IF mappings are used to generate the multiple-pronunciation, while the surface-form base MLLR technology is used to improve the acoustic model. After using it, the recognizer becomes worse than other methods. We think the main reason is because we have no enough data to learn syllable-dependent mapping rules.

19.4 Multi-Pronunciation Expansion (MPE) based on Accumulated Unigram Probability

More pronunciations help model pronunciation variations, but also lead to more confusion, there should be tradeoff. In our experiments, Accumulated Unigram Probability (AUP) used as the criterion:

- Only words with higher unigram probabilities will each have multiple pronunciations;
- Words with lower unigram probabilities will each have a single standard pronunciation;

Table 7 shows the AUP that used to reduce the size of multi-pronunciation lexicon. The first column is the designed accumulated word unigram probability, and the other columns shows the

actual accumulated word unigram probability, log-probability threshold and how many words exceeding the threshold, e.g. they can have multi-entries.

Table 7. Accumulated Unigram Probability

Designed	Actual		
Accumulated Prob.	Accumulated Prob.	Log Prob. Threshold	#Words
0.10	0.10782136	-0.967295	1
0.20	0.21344301	-1.719572	5
0.30	0.30682193	-1.940906	12
0.40	0.40467247	-2.258349	25
0.50	0.50225045	-2.442327	47
0.60	0.60066204	-2.757656	87
0.70	0.70055931	-3.153711	176
0.80	0.80018502	-3.594763	416
0.90	0.90050580	-4.240960	1,292
0.92	0.92081776	-4.419749	1,735
0.94	0.94120825	-4.624082	2,427
0.96	0.96064531	-4.867548	3,543
0.98	0.98000226	-5.250755	5,838
1.00	1.00000000	No	15,724

19.5 Primary Results

The primary results are show in Figure 1 and Figure 2. The Figure 1 shows the CER of the baseline system and other proposed methods, including context-free PTH-IF mapping, context-free WDC-IF mapping plus MLLR, for different group of speakers (See Table 2). The MPE method uses the 94% log probability as the threshold. It is obvious that proposed methods can effectively reduce the CER, especially for the EL (Educational Low) speakers.

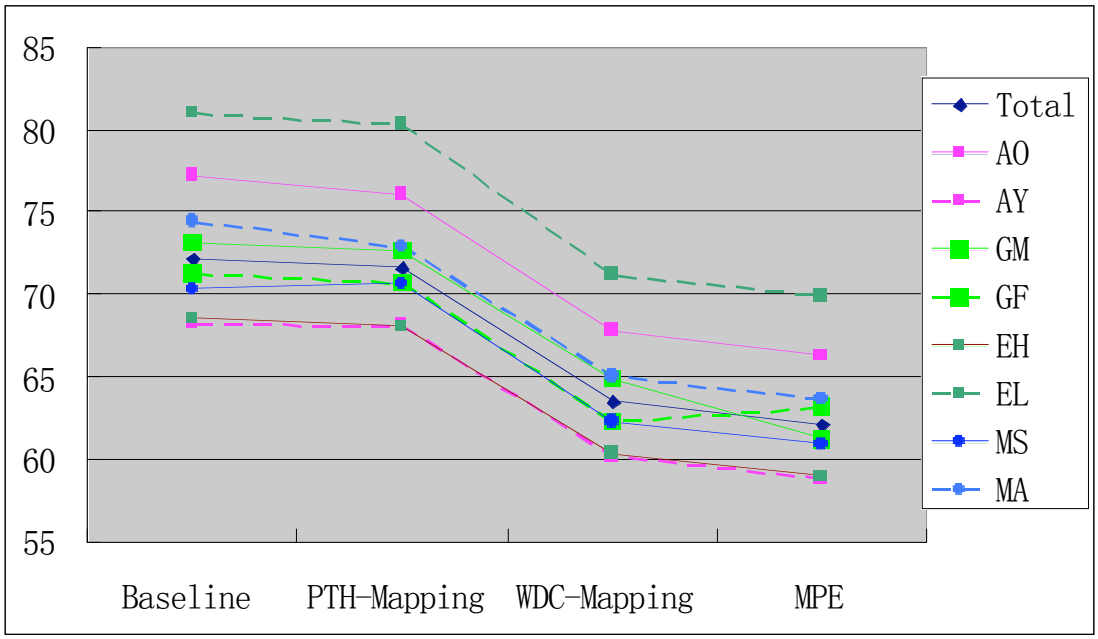

Figure 1. CER curves for different sub-sets & methods

Figure 2 shows the CER curve when using different threshold to reduce the lexicon size for both the base form and the surface form MLLR. 0% means using only single entry, while 100% means using all the possible entries for each word. For n% (0<n<100), only the high frequency words that accumulated the unigram probability up to n% can use the multi-entries, while other words can only use the single pronunciations. We can see that the MPE method is effective to refine the acoustic models, and the surface form MLLR is better than the base form MLLR method when we using only 20 speakers data to adapt the acoustic models.

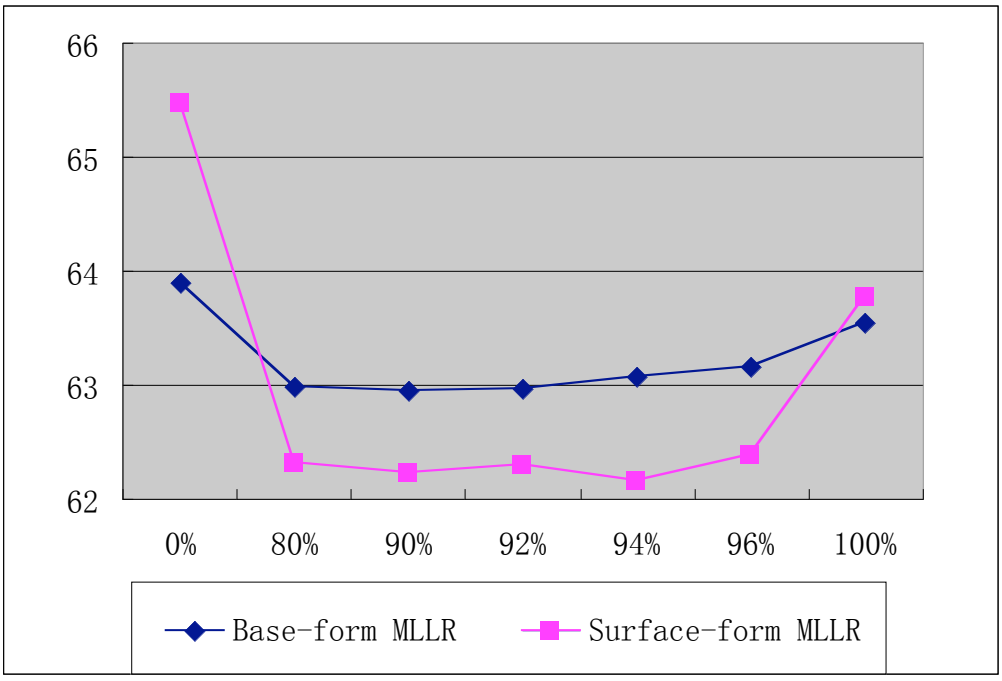

Figure 2. CER and Vocabulary Size curve with MPE

20 Conclusions

An extensible Framework for Dialectal Chinese Speech Recognition is proposed. Our research focuses on how to extract the dialect-related knowledge, such as the Chinese IF mapping, syllable mapping, cross-dialect synonyms, and so on, but use only a small quantity of or even lacking adaptation data to improve the recognizer. The context-free PTH-IF mapping, context-free WDC-IF mapping and syllable-dependent WDC-IF mapping combining with the surface-form based MLLR acoustic adaptation are performed to modify the PTH recognizer to fit the WDC speech recognition task in our experiments. For more efficiency, Multi-Pronunciation Expansion (MPE) based on Unigram Probability is used to reduce the size of lexicon. After these experiments we get such conclusions:

- We can expect that using WDC recognizer to recognize PTH, the performance will degrade, but we would expect it will not decrease too much, because we use the surface form to adapt the acoustic model instead the base-form transcriptions
  By using the WDC recognizer, we get
  - Over 10% CER reduction to recognize WDC;
  - CER increase of only 0.62% to recognize PTH.
- The use of knowledge is useful and effective
- The MPE method is effective to refine the acoustic models, and the surface form MLLR is better than the base form MLLR method when we using only 20 speakers data to adapt the acoustic models
- In this project, there are several problems to solve, including: channel, speaking-style, dialect background, and domain mismatches.
  - It is easier to solve all these problems by simply using the adaptation method;
  - Our method focuses only on the dialect problem;
  - The results using our method could be better if we integrate those methods related to channel, and speaking-style.

21 Future work

Our future plan after this workshop includes

- Continue on the current project, including:
  - Investigating the syllable-dependent mapping
  - Rank-based Rescoring based on IF lattices
- Language Model Adaptation
  - Different word form with same meaning
    - Such as: 喜欢 vs. 欢喜 - like; 做饭 vs. 烧饭 - cook
    - Linguists say the vocabulary similarity rate between Putonghua and Wu dialect is about 60~70%.
  - Different word order
    - 你先走 (you first go) vs. 你走先 (you go first)

References

[1] Chao. Huang, “Accent issue in large vocabulary continuous speech recognition,” Microsoft Research Technical Report, MSR-TR-2001-69, 2001

[2] Jing Li, Fang Zheng, Zhenyu Xiong, and Wenuhu Wu, “Construction of Large-Scale Shanghai Putonghua Speech Corpus for Chinese Speech Recognition”, Oriental-COCOSDA, pp.62-69, Oct. 1-3, Sentosa, Singapore, 2003

[3] Helmer Strik, Catia Cucchiarini, “Modeling pronunciation variation for ASR: A survey of the literature”, Speech communications, p225-246, 1999

[4] Steve Young, Gunner Evermann, Thomas Hain, et al., “The HTK book (for HTK Version 3.2.1)”, http://htk.eng.cam.ac.uk/, 2002

REFERENCES

Anastasakos, T., J. McDonough, R Schwartz, and J. Makhoul. 1996. A compact model for speaker-adaptive training. In *Proceedings of ICSLP 96*, Philadelphia.

Bacchiani, Michiel, Michael Riley, Brian Roark, and Richard Sproat. 2004. MAP stochastic grammar adaptation. Submitted to *Computer Speech and Language*.

Chen, Tao, Chao Huang, Eric Chang, and Jingchun Wang. 2001. Automatic accent identification using gaussian mixture models. In *IEEE Workshop on ASRU*, Italy.

de Marcken, Carl. 1996. *Unsupervised Language Acquisition*. Ph.D. thesis, MIT, Cambridge, MA.

Goronzy, Silke, Marina Sahakyan, and Wolfgang Wokure. 2001. Is non-native pronunciation modelling necessary? In *Proceedings of Eurospeech2001*, Aalborg, Denmark.

GRM Library. 2004. http://www.research.att.com/sw/tools/grm.

He, Xiaodong and Yunxin Zhao. 2003. Fast model selection based speaker adaptation for nonnative speech. *IEEE Transactions on Speech and Audio Processing*, 11(4):298–307.

Huang, Chao, Eric Chang, Jianlai Zhou, and Kai-Fu Lee. 2000. Accent modeling based on pronunciation dictionary adaptation for large vocabulary mandarin speech recognition. In *ICSLP 2000*, pages 818–821, Beijing.

Huang, Chao, Tao Chen, and Eric Chang. 2004. Accent issues in large vocabulary speech recognition. *International Journal of Speech Technology*, 7:141–153.

Humphries, J.J., P.C. Woodland, and D. Pearce. 1997. Using accent-specific pronunciation modelling for robust speech recognition. In *Proc. Eurospeech '97*.

Ikeno, Ayako, Bryan Pellom, Dan Cer, Ashley Thornton, Jason M. Brenier, Dan Jurafsky, Wayne Ward, and William Byrne. 2003. Issues in recognition of spanish-accented spontaneous english. In *Proceedings of IEEE/ISCA Workshop on Spontaneous Speech Processing and Recognition*, Tokyo. IEEE/ISCA.

Karypis, George, 2003. *Cluto: A Clustering Toolkit*. University of Minnesota, Department of Computer Science, Minneapolis, MN. `http://www-users.cs.umn.edu/~karypis/cluto/`.

Klakow, D. 1998. Language-model optimization by mapping of corpora. In *Proceedings of ICASSP*, Seattle, WA.

Kumpf, Karsten and Robin King. 1996. Automatic accent classification of foreign-accented Australian english speech. In *ICSLP-96*, Philadelphia, PA.

Larson, M., D. Willett, J. Koehler, and G Rigoll. 2000. Compound splitting and lexical unit recombination for improved performance of a speech recognition system for German parliamentary speeches. In *Proceedings of ICSLP*, Beijing, China.

Leggetter, C. and C. WoodlandP. 1995. Flexible speaker adaptation using maximum likelihood linear regression. In *Proceedings of Eurospeech'95*, pages 1155–1158, Madrid.

Lincoln, Mike, Stephen Cox, and Simon Ringland. 1998. A comparison of two unsupervised approaches to accent identification. In *ICSLP-98*, Sydney.

Liu, Wai Kat and Pascale Fung. 1999. Fast accent identification and accented speech recognition. In *ICASSP*.

Liu, Wai Kat and Pascale Fung. 2000. Mllr-based accent model adaptation without accented data. In *Proceedings of ICSLP 00*, Beijing.

Liu, Yi and Pascale Fung. 2003a. Modeling partial pronunciation variations for spontaneous Mandarin speech recognition. *Computer Speech and Language*, 17:357–359.

Liu, Yi and Pascale Fung. 2003b. Partial change accent models for accented mandarin speech recognition. In *Proceedings of IEEE Automatic Speech Recognition and Understanding Workshop*, U.S. Virgin Islands.

Livescu, Karen and James Glass. 2003. Lexical modeling of non-native speech for automatic speech recognition. In *ICASSP 2003*, Istanbul.

Ljolje, Andrej. 2002. Speech recognition using fundamental frequency and voicing in acoustic modeling. In *ICSLP-02*, pages 2137–2140, Beijing.

Manning, Chris and Hinrich Schütze. 2003. *Foundations of Statistical Natural Language Processing*. MIT Press.

Mayfield, Laura Tomokiyo. 2002. *Recognizing non-native speech: Characterizing and adapting to non-native usage in speech recognition*. Ph.D. thesis, Carnegie Mellon University.

Mayfield Tomokiyo, Laura. 2000. Lexical and acoustic modeling of non-native speech in LVCSR. In *ICSLP-00*, Beijing.

Mayfield Tomokiyo, Laura and Alex Waibel. 2001. Adaptation methods for non-native speech. In *Proceedings of Multilinguality in Spoken Language Processing*, Aalborg.

Mohri, Mehryar. 2001. Weighted grammar tools: the GRM library. In Jean Claude Junqua and Gertjan van Noord, editors, *Robustness in Language and Speech Technology*. Kluwer Academic Publishers, Dordrecht, pages 165–186.

Mohri, Mehryar, Fernando Pereira, and Michael Riley. 1998. *A Rational Design for a Weighted Finite-State Transducer Library*. Number 1436 in Lecture Notes in Computer Science. Springer.

Mohri, Mehryar, Fernando Pereira, and Michael Riley. 2002. Weighted finite-state transducers in speech recognition. *Computer Speech and Language*, 16(1):69–88.

Mohri, Mehryar and Michael Riley. 2001. A weight pushing algorithm for large vocabulary speech recognition. In *EUROSPEECH'01*, Aalborg, September.

Norman, Jerry. 1988. *Chinese*. Cambridge University Press, Cambridge.

Ramsey, S. Robert. 1989. *The Languages of China*. Princeton University Press, Princeton, NJ.

Ries, K., F. D. Buo, and A. Waibel. 1996. Class phrase models for language modeling. In *Proceedings of ICSLP*, Philadelphia, PA.

Saraclar, Murat and Richard Sproat. 2004. Lattice-based search for spoken utterance retrieval. In Daniel Marcu Susan Dumais and Salim Roukos, editors, *HLT-NAACL 2004: Main Proceedings*, pages 129–136, Boston, Massachusetts, USA, May 2 - May 7. Association for Computational Linguistics.

Schultz, Tanja, Qin Jin, Kornel Laskowski, Alicia Tribble, and Alex Waibel. 2002. Speaker, accent and language identification using multilingual phone strings. In *HLT-2002*, San Diego.

Shafran, Izhak, Michael Riley, and Mehryar Mohri. 2003. Voice signatures. In *Proceedings of IEEE Automatic Speech Recognition and Understanding Workshop*, U.S. Virgin Islands.

Sproat, R. 2001. Corpus-based methods in chinese morphology and phonology. Technical report, LSA. http://compling.ai.uiuc.edu/rws/newindex/publications.html.

Sproat, Richard and Chilin Shih. 2001. Corpus-based approaches to chinese morphology and phonology. Lecture notes for course presented at the 2001 LSA Summer Institute, Santa Barbara, CA. `http://compling.ai.uiuc.edu/rws/newindex/notes.pdf`.

SRI Language Modeling Toolkit. 2004. http://www.speech.sri.com/projects/srilm/.

Teixeira, Carlos, Horacio Franco, Elizabeth Shriberg, Elizabeth Precoda, and Kemal Sönmez. 2001. Evaluation of speaker's degree of nativeness using text-independent prosodic features. In *Proceedings of the Workshop on Multilingual Speech and Language Processing*, Aalborg, Denmark.

Teixeira, Carlos, Isabel Trancoso, and António Serralheiro. 1996. Accent identification. In *ICSLP-96*, Philadelphia, PA.

Wang, Zhirong, Tanja Schultz, and Alex Waibel. 2003. Comparison of acoustic model adaptation techniques on non-native speech. In *ICASSP 2003*, pages 540–543. IEEE.

Young, Steve, Gunnar Evermann, Thomas Hain, Dan Kershaw, Gareth Moore, Julian Odell, Dave Ollason, Dan Povey, Valtcho Valtchev, and Phil Woodland, 2002. *The HTK Book (for HTK Version 3.2.1)*. Cambridge University, Cambridge. `http://htk.eng.cam.ac.uk/`.

Zheng, Yanli and Mark Hasegawa-Johnson. 2004. Stop consonant classification by dynamic formant trajectory. In *ICSLP*, Jeju Island, Korea.

Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant No. 0121285.