國立臺灣大學電機資訊學院電信工程學研究所

博士論文

Graduate Institute of Communication Engineering

College of Electrical Enginnering and Computer Science

National Taiwan University

Doctoral Dissertation

使用跨語言聲學模型及音框層級語言識別來辨識高度不平衡雙語混合課程之整合性架構

An Integrated Framework for Recognizing Highly Imbalanced Bilingual Code-switched Lectures with Cross-language Acoustic Modeling and Frame-level Language Identification

葉青峰

Ching-Feng Yeh

指導教授：李琳山 教授

Advisor: Lin-Shan Lee, Ph.D.

中華民國一百零四年九月

September, 2015

國立臺灣大學博士學位論文
口試委員會審定書

使用跨語言聲學模型及音框層級語言識別來辨識高度不平衡雙語混合課程之整合性架構

An Integrated Framework for Recognizing Highly Imbalanced Bilingual Code-switched Lectures with Cross-language Acoustic Modeling and Frame-level Language Identification

本論文係葉青峰君 (D00942013) 在國立臺灣大學電信工程學研究所完成之博士學位論文，於民國 104 年 8 月 28 日承下列考試委員審查通過及口試及格，特此證明

口試委員：李琳山 (簽名)

(指導教授)

貝蘇章　　張智星

林宗男

李宏毅

系主任、所長 吳宗霖 (簽名)

摘要

本論文探討一種常見的雙語混合語音的辨識：語者所使用的語句中大部分的語音訊號是用主語言（通常是語者的母語）所說，但其中包含小部分的詞或片語是用客語言（通常是語者的第二語言）所說的。在此狀況下，不只因為語言在語句內頻繁切換而造成語音辨識困難，而且客語言的資料量少得多，造成客語言的辨識正確率明顯甚低。本論文提出了一個辨識這種高度不平衡的雙語混合語音的整合性辨識系統架構。這其中包含了在聲學模型上進行不同層級（模型、狀態、高斯）的單位融合做到跨語言語料共享，語音單位的恢復加强以重建融合後的聲學模型，依據單位佔用度排序提供更彈性的跨語言以及語言內的語料共享，以及使用模糊事後機率特徵估測音框層級的語言事後機率等。此外，本論文也將這些方法延伸到今日最成功的用深層類神經網路作為瓶頸特徵抽取器以及隱藏式馬可夫模型狀態模擬器的兩種方法上。我們用一套在真實情境下錄製的語料進行統一條件下的測試，將所有提出方法做了完整的比較。實驗結果顯示本論文所提出的系統架構能夠大幅改善雙語混合語音辨識的正確率。

Abstract

This thesis considers the recognition of a widely observed type of bilingual code-switched speech: the speaker speaks primarily the host language (usually his native language), but with a few words or phrases in the guest language (usually his second language) inserted in many utterances of the host language. In this case, not only the languages are switched back and forth within an utterance so the language identification is difficult, but much less data are available for the guest language, which results in poor recognition accuracy for the guest language part. In this thesis, we propose an integrated overall framework for recognizing such highly imbalanced code-switched speech. This includes unit merging approaches on three levels of acoustic modeling (triphone models, HMM states and Gaussians) for cross-lingual data sharing, unit recovery for reconstructing the identity for units of the two languages after being merged, unit occupancy ranking to offer much more flexible data sharing between units both across languages and within the language based on the accumulated occupancy of the HMM states, and estimation of frame-level language posteriors using Blurred Posteriorgram Features (BPFs) to be used in decoding. In addition, we also evaluated two approaches extending above approaches based on HMMs to the state-of-the-art deep neural networks (DNNs), including using bottleneck features in HMM/GMM and modeling context-dependent HMM states. We present a complete set of experimental results comparing all approaches involved for a real-world application scenario under unified conditions, and show very good improvement achieved with the proposed approaches.

Contents

List of Figures

List of Tables

Chapter 1 Introduction and Overview of the Framework

1.1 Introduction

Conventionally, speech recognition technologies are developed to transcribe utterances in a specific language. But in the globalized world today, many people are capable of speaking more than one languages and actually using more than one languages in their daily lives. As a result, very often the speech signals observed in our daily lives also include more than one languages. This is why substantial effort has been made to try to extend existing speech recognition technologies to consider multilingual scenarios [1–11].

A major concern for multilingual speech processing is the phoneme sets to be used for constructing the system. Very often some phonemes are shared by different languages; some phonemes in different languages are slightly different but similar; and some other phonemes are unique for specific languages. This makes acoustic modeling and the lexicon construction difficult, because the similarities between the phonemes are usually difficult to measure quantitatively by linguistic knowledge. Many approaches such as merging acoustic units on different levels in acoustic modeling [12–19] have been proposed to handle these problems and shown to be very helpful with results in good agreement with the linguistic knowledge. It was also found that the acoustic models for such tasks can be improved by advanced model structures such as subspace Gaussian mixture model [20,21] by jointly modeling the cross-language acoustic information. Discriminative training approaches such as minimum phone error (MPE) [22] training and neural networks were also used for classifying confusing acoustic units in such tasks [23–26]. Approaches such as sub-word unit modeling [27], pronunciation modeling, articulatory features [28,29] and deep neural network for multilingual processing [25,30] were also used and shown to be successful.

In general, bilingual speech can be classified into two categories [1,17,24,28,31–34]. In the first category, the speaker switches language from sentences to sentences. For example, in the sentence, ”It’s fine. 謝謝你(Thank you).”, the first sentence is in English, while the second in Mandarin. In the other category, the languages are switched from words to words within a sentence. For example, in the sentence, ”這個equation很複雜(This equation is very complicated).”, the word ”equation” in English is embedded in a sentence of Mandarin. The latter category is very common for speakers with non-English native languages, especially when they speak very good English and many English words (and phrases) are not necessarily or properly translated into their native languages. So when they speak in their native languages, some English words (or phrases) appear in the utterances. The word ”code-switching” in this thesis refers to this second category and is the focus of this thesis. Such code-switching speech is very frequently observed in large parts of the world, as long as the native languages of the majority of the speakers are not English, such as in Asia. In such cases, English is regarded as the guest language while the native as the host language. In fact, such situation also happens for other major languages other than English, such as Arabic and French for North Africa.

An extra difficulty for the above code-switched bilingual speech is the highly imbalanced data distribution for involved languages [17,24,28,31–34], i.e., much more host language data and very limited guest language data, since only few words or phrases of the guest language are embedded in the sentences of the host language, if code-switching happens. This not only makes acoustic modeling for the guest language difficult, but the recognizer tends to take most speech signals as in the host language. The possible reasons include not only the fact that the acoustic models for the host language units are better trained with more data and therefore better fitted to the signals, but the language model almost always gives higher prior probabilities to the host language words. This difficulty of highly imbalanced data distribution is a major problem considered in this thesis.

Another distinguishing feature of such code-switching environment is the difficulty in language identification [35–40]. In most multilingual tasks, the basic unit for language identification is usually an utterance. However, the unit for language identification in code-switched speech considered here should be smaller, such as segments or frames of signals, since the language may be switched back and forth within an utterance. This much shorter length of considered signal makes the language identification much harder, and is also a major problem considered in this thesis.

An additional problem for this type of code-switching is that the guest language is always spoken by the non-native language speakers. Therefore, these English words are often with strong accents, and the accents vary significantly from speaker to speaker. As a result, the huge quantity of available English data produced by native English speakers usually does not help [41,42]. This will also be verified in the experiments. Moreover, the English words and native language words are fluently spoken by the same speaker within the same utterance. Taking Mandarin-English code-switching as an example, the English words embedded in the Mandarin utterances are very often composed of phonemes sounding like Mandarin phonemes rather than English phonemes because they are spoken by

Mandarin speakers. As a result, the recognizer tends to recognize the English words as a sequence of Chinese characters (each Chinese character corresponds to a syllable of C-V structure).

Although the recognition of the second category of code-switched speech have been very important problems, only limited works have been reported for acoustic modeling, primarily for Mandarin-English [11,13,17,21,23,24,28,31] and Cantonese-English [16,43]. Many works were reported for language modeling for this problem [26,43,44] as well. Due to the difference in the local culture, the genre of the speech and the speaker behavior, the situations previously reported vary in different tasks. For example, the average percentage of the guest language in the utterances is relatively high in Malaysia (37%) [32,42] and Hong Kong (28%) [43] but low in Taiwan (15%) [17], as reported in the respective works. Probably because English is one of the official languages in Malaysia and Hong Kong, so most speakers are more used to speak in English, but the situation is different in Taiwan. So the imbalanced data distribution problem is much more serious in the case of Taiwan. Also, previous works indicated that code-switching happens only between specific POSs in specific structures (e.g. native switching to English for nouns following verbs), which is useful in language modeling but not in acoustic modeling [12–19,34]. Moreover, the only works reported up to the date were on lectures [17,28,43] and daily conversations [26,32]. For course lectures in a specific domain, most English words appeared to be domain-specific terminologies related to the topics of the courses, while almost all function words were in the native language [17,26,32,42]. But the above description does not always fit the case of daily conversation [26,32]. Furthermore, code-switching varies from speaker to speaker [17,24]. This is why people modeled the code-switching characteristics by clustering the speakers with similar behavior [45]. The distinct natures and issues of these different tasks mentioned above make it difficult to compare the works reported for different tasks directly. For example, one of the key issues in this thesis is the imbalanced data distribution (only 15% data are in English) for course lectures in Taiwan [17,24], which may not be serious for the tasks in Malaysia (37% data are in English) [32,42].

Recently, the deep neural networks (DNN) were shown to be able to improve speech recognition performance significantly [25,26,30,46,47]. For multilingual speech processing, various approaches using deep neural networks were also proposed [25,26,30,34], including parameter sharing among different languages [25,30] and rapid adaptation between languages [26]. The most popular form of deep neural network application in acoustic modeling is the context-dependent DNN-HMM (CD-DNN-HMM) [46], in which each context-dependent HMM state is modeled by a node of the output layer of the deep neural networks. In addition, bottleneck features from the deep neural network were also used in HMM/GMM (referred to as BF-HMM/GMM here) [48]. In this thesis, both CD-DNN-HMM and BF-HMM/GMM are considered and tested.

Although some of the above problems have been individually analyzed previously in some way in different tasks, in this thesis, we propose an integrated framework for transcribing highly imbalanced bilingual code-switched speech for a real-world application scenario (course lectures collected in Taiwan). The approaches used in this framework include cross-language acoustic modeling and frame-level language posterior estimation. For cross-language acoustic modeling, we propose unit merging and recovery on three different levels (models, states and Gaussians) [17,31,33], in which both unit similarity and training data availability are considered to take care of the imbalanced data distribution. We further propose the unit recovery techniques in addition for improving the performance. Moreover, we propose approaches to consider the data availability based on the accumulated occupancy of HMM states to realize data sharing both across language and within the same language. For estimation of the frame-level language posteriors, we utilized a neural network with specially designed blurred posteriorgram features (BPFs), and the estimated posteriors are used in decoding [36–39]. Also, we extend these approaches for HMMs to the deep neural networks, which is very popular recently, in both forms of direct HMM state modeling and bottleneck feature extraction for code-switched speech recognition. In summary, in this thesis we present an integrated framework for the task of bilingual code-switching speech recognition putting together different approaches with various considerations and report complete experimental results under unified conditions for a real-world application scenario. In the experiments, both cases of speaker dependent (SD) with sufficient training data and speaker adapted (SA) with very limited adaptation data are considered.

1.2 Baseline System

A bilingual speech recognition system can be built by simply extending each component of a conventional speech recognition system from monolingual to bilingual [17,31,32], as shown in Figure 1.1 . For acoustic modeling, all phonemes of the two languages involved can be combined to form a phoneme set used for acoustic model construction, even though similar phonemes belonging to different languages may be in the phoneme set at the same time. For example, the phoneme set for the bilingual Mandarin-English speech

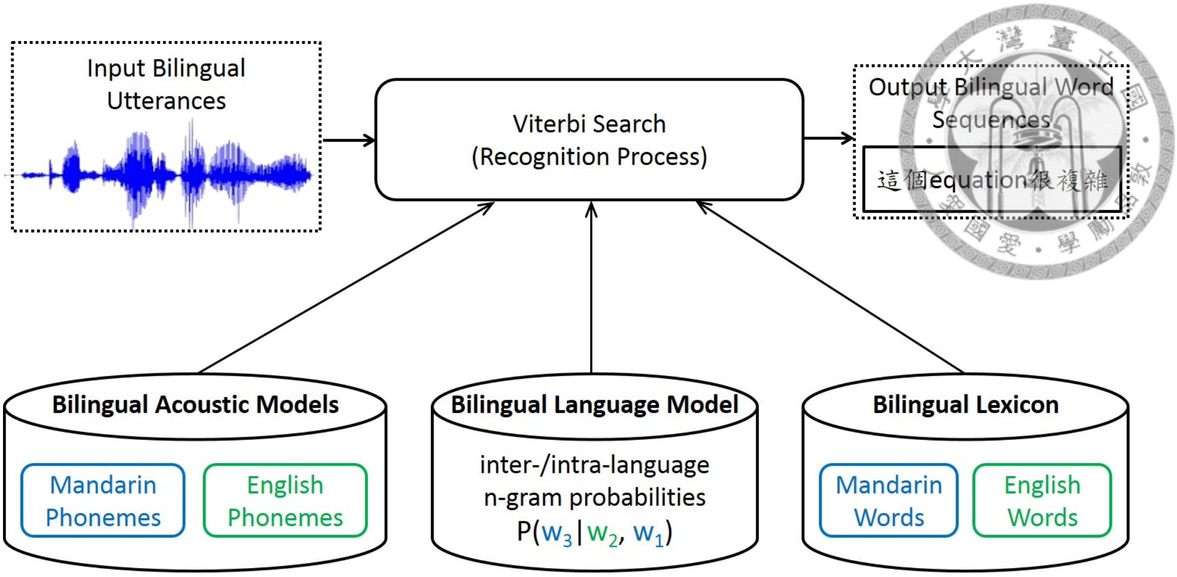

Input Bilingual Utterances
Viterbi Search (Recognition Process)
Output Bilingual Word Sequences
這個equation很複雜
Bilingual Acoustic Models
Mandarin Phonemes
English Phonemes
Bilingual Language Model
inter-/intra-language n-gram probabilities
P(w3|w2, w1)
Bilingual Lexicon
Mandarin Words
English Words

Figure 1.1: *Baseline System by Extending Acoustic Models, Language Model and Lexicon for Recognizing Bilingual Code-switched Speech.*

considered here can simply include all Mandarin phonemes plus all English phonemes. Similarly, the lexicon can be built by including all words needed for the two languages labeled with phoneme sequences in corresponding languages. As for language model, n-gram probabilities based on the bilingual lexicon should cover both inter-lingual and intra-lingual combinations. Such a system is certainly capable of recognizing bilingual speech. We will take such a system as the baseline.

1.3 Overview of the Proposed Framework for Bilingual Speech Recognition

Of course, the baseline system mentioned above does not take the distinct nature of the code-switched bilingual speech as mentioned above into consideration. Here we proposed an integrated framework for such a purpose, with an overall system block diagram

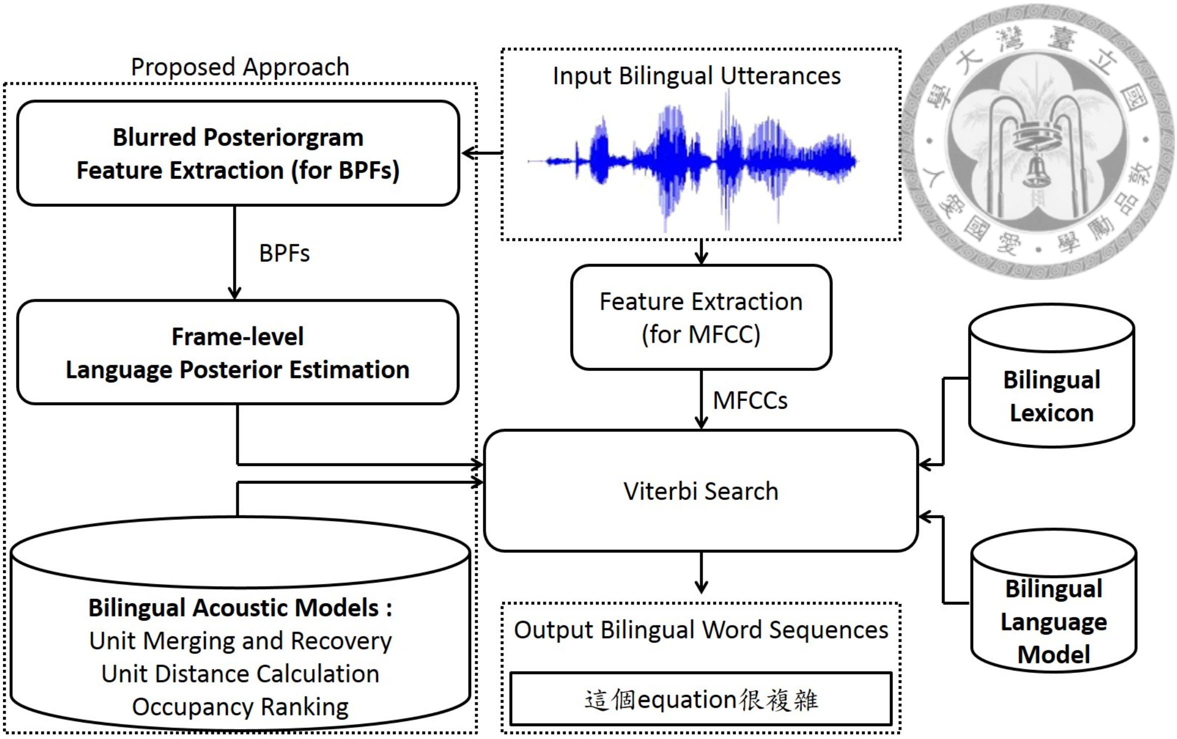

Proposed Approach
Blurred Posteriorgram Feature Extraction (for BPFs)
BPFs
Frame-level Language Posterior Estimation
Bilingual Acoustic Models :
Unit Merging and Recovery
Unit Distance Calculation
Occupancy Ranking
Input Bilingual Utterances
Feature Extraction (for MFCC)
MFCCs
Viterbi Search
Bilingual Lexicon
Bilingual Language Model
Output Bilingual Word Sequences
這個equation很複雜

Figure 1.2: *Proposed Framework for Recognizing Highly Imbalanced Bilingual Code-switched Speech.*

as shown in Figure 1.2. The bilingual lexicon and language model are exactly the same as the baseline mentioned above in Figure 1.1. The acoustic models are improved by the unit merging (Section 3.1) and recovery (Section 3.2) based on unit distance calculation (Section 3.3) and unit occupancy ranking (Section 3.4) proposed in this thesis. In addition, an extra frame-level language posterior estimation (Chapter 5) based on specially designed features referred to as blurred posteriorgram features (BPFs) (Section 5.3) is included. The language posteriors estimated in this way is used in the Viterbi decoding. The conventional acoustic features (MFCC) for recognition are also extracted in addition to the BPFs. Because each speech segment may belong to either language, the acoustic models, word hypotheses and n-gram language models for both languages and the switching between them should all be considered in the Viterbi search. The system finally generates the output code-switched word sequences.

1.4 Chapter Outline

The rest of this thesis is organized as follows. In Chapter 2, the characteristics of the target corpora and baseline experimental setup with results are described. Details of the proposed HMM-based cross-lingual acoustic modeling, including acoustic unit merging (Section 3.1) and recovery (Section 3.2), distance calculation on various levels for unit merging (Section 3.3), and unit ranking based on accumulated occupancy of HMM states (Section 3.4) are described in Chapter 3. Experimental results for those HMM-based cross-lingual acoustic modeling approaches are presented in Chapter 4. The frame-level language posterior estimation and experimental results are described in Chapter 5. Apart from the proposed HMM-based approaches above, the proposed DNN-based cross-lingual acoustic modeling and experimental results are presented in Chapter 6. Finally, concluding remarks are described in Chapter 7.

Chapter 2 Target Corpora and Baseline Experimental Results

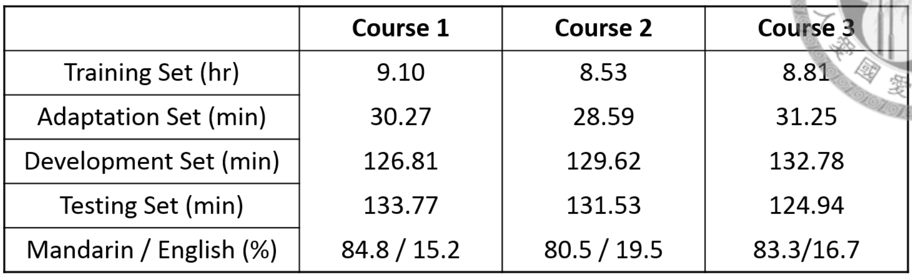
2.1 Target Code-switched Bilingual Corpora

Although the second category of code-switching considered here is very common, the work reported for the acoustic modeling for it is very limited, and it is not easy to find corresponding data set either. Almost all works reported previously use data sets individually collected for specific tasks and therefore these data sets are primarily proprietary [16,17,26,28,32,43]. Also, because of the different nature of the tasks reported previously as mentioned above, the data sets used for these tasks may not be used jointly in a specific work due to the diversity of the characteristics of the data sets. Here we also collected the specific corpora for the purpose of this work.

The corpora used for this work were the recorded lectures of three courses offered in National Taiwan University in spontaneous speech with highly imbalanced Mandarin-English code-switching characteristics (Mandarin as host and English as guest languages) as mentioned above. In these corpora, most English words appeared to be domain-specific terminologies related to the content of the course. Therefore good accuracies for the English words are important.

Courses 1 and 2 were offered by the same instructor, but with completely different contents (therefore different vocabulary and n-grams), while course 3 had contents similar to course 1, but was offered by another instructor. The recording acoustic environments for all the three courses including the microphones and the classrooms were different.

Table 2.1: *Details for the Target Corpora.*



	Course 1	Course 2	Course 3
Training Set (hr)	9.10	8.53	8.81
Adaptation Set (min)	30.27	28.59	31.25
Development Set (min)	126.81	129.62	132.78
Testing Set (min)	133.77	131.53	124.94
Mandarin / English (%)	84.8 / 15.2	80.5 / 19.5	83.3/16.7

So, we primarily treat the courses as three sets of separated recordings and evaluate the proposed methods on them separately. Although the results for two speakers only here seem to be very limited, considering the fact that code-switching behavior is very speaker dependent and the English words were all produced by non-native speakers with accent which varies from speaker to speaker, the results here may serve as good reference for the problem. Of course the results here may not be directly generalized to all speakers for each scenario as mentioned previously. Considering the difficulty of finding data sets for code-switching research, this is currently the best we can get.

The detailed statistics of the corpora are listed in Table 2.1. We see the percentage of English (guest language) for the bilingual corpora is only 15-19%, or roughly 1.5 hours in training and 5 minutes in adaptation data.

2.2 Experimental Environment Setup

The target corpora used for evaluation are already mentioned in the previous section and listed in Table 2.1. For the speaker adaptation scenario, the initial speaker independent (SI) models were trained from two different corpora for the two languages. The Mandarin SI models were trained with the ASTMIC corpus of read speech in Mandarin only with a total length of 31.8 hours. The English models were trained with the EATMIC corpus [49], which was also a read speech corpus in English only but produced by Taiwanese speakers with a total length of 29.7 hours. Note that the test set mentioned above in Section 2.1 was spontaneous in lecture form, while the SI models were trained with read speech here.

The bilingual lexicon used here included English words, Chinese words and all commonly used Chinese characters taken as mono-character Chinese words. Since the words used in the lecture corpora were restricted to a very specific domain of the course, for the guest language of English only a small portion of the normal English vocabulary actually appeared in the corpora. As a result, target-domain related corpora including word frequency counts were used in the selection of the English words in construction of the bilingual lexicon, with some manually picked special terms for the target-domain added to the English part. Extra Chinese words were also generated by segmenting a large Chinese text corpus using PAT-Tree based approaches [31]. There were about 11000 Chinese words and 2500 English words in the lexicon. All words in the development set and testing set were covered in the lexicon. So there were no out-of-vocabulary (OOV) words in the experiments reported here.

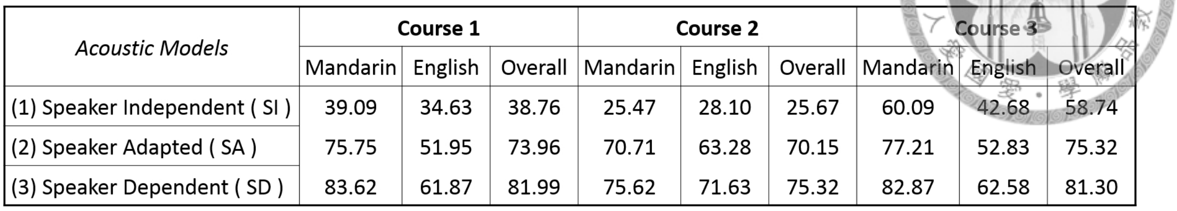
For language modeling, the background model is trained with a combined corpus including Gigaword, Yahoo! News plus some target-domain related corpora such as master thesis in related domains. We used Kneser-Ney trigram language model started with this background model and then adapted with the transcriptions of the training set for the target lectures here. The total numbers of trigrams for courses 1, 2 and 3 were about 65k, 55k and 65k respectively.

The feature extraction and model training processes followed the standard approaches, with the 39 MFCC parameters as features and triphone models with state-clustering by decision trees obtained in Maximum likelihood (MLE) model training [50]. For experiments regarding deep neural networks as mentioned in Chapter 6, only results from course 1 and 2 in the speaker dependent (SD) scenario are reported due to the limited computational resources. The 39 MFCC parameters were concatenated in consecutive 9 frames as features for CD-DNN-HMM. The number of HMM states for HMM/GMM baseline were 2845 and 3172 for courses 1 and 2, respectively. There were 4 hidden layers in each deep neural network, with 2048 nodes in each hidden layer. For bottleneck features of the deep neural networks, the feature dimension was reduced from 2048 to 40 by linear discriminative analysis (LDA).

The way the recognition performance was evaluated followed the earlier work [17, 31,35] and was very similar to the mixed error rate (MER) used for multilingual speech recognition evaluation later on [26,32]. That is, when aligning recognition results with the reference transcriptions, insertions, deletions and substitutions were evaluated respectively for each language and summed up for overall evaluation. The basic unit for alignment and calculation is character for Mandarin and word for English. Individual performance for both Mandarin and English is reported. Since English words are very often the key terms in the code-switched lecture considered here, the accuracy for English part alone is a focus.

In addition to recognition accuracy, significance tests were also performed over the overall results (Considering Mandarin and English jointly) for the proposed approaches as compared to the respective baselines. For example, those with unit merging compared

Table 2.2: *Baseline Results (Accuracies) (%).*



Acoustic Models	Course 1			Course 2			Course 3		
	Mandarin	English	Overall	Mandarin	English	Overall	Mandarin	English	Overall
(1) Speaker Independent (SI)	39.09	34.63	38.76	25.47	28.10	25.67	60.09	42.68	58.74
(2) Speaker Adapted (SA)	75.75	51.95	73.96	70.71	63.28	70.15	77.21	52.83	75.32
(3) Speaker Dependent (SD)	83.62	61.87	81.99	75.62	71.63	75.32	82.87	62.58	81.30

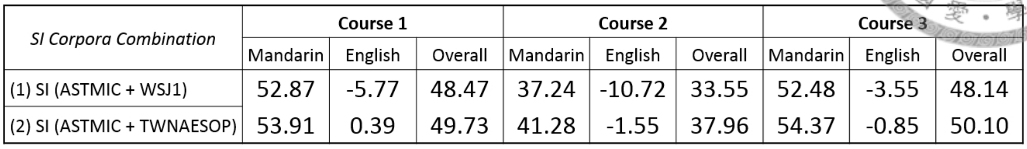
to without unit merging and those with unit recovery compared to without unit recovery. Pair-wise accuracy comparison was used for the p-value test. Results with significant improvements, i.e., those with p-values less than 0.05, are labeled with a superscript symbol ”+” in the results below.

The parameters of proposed approaches were set by obtaining the best performance on the development set in Table 2.1 with grid search. These parameters were then applied to the testing set for experimental results.

2.3 Baseline Results

The recognition accuracies for the baseline system for different sets of acoustic models are listed in Table 2.2. Row 1 are the results for the initial SI models without adaptation data. Both Mandarin and English accuracies were very poor here obviously due to the mismatch between the read speech of SI training corpora and the spontaneous speech of the target lecture corpora. In addition, the English accuracies were especially poor compared to Mandarin (except for course 2), possibly because of the imbalanced prior distribution in the language model. The language model was adapted with the training transcriptions, in which the frequencies of Chinese words were much higher than those of

Table 2.3: *Baseline Results with Different Corpus Used for Building English SI Models (Accuracies) (%).*



SI Corpora Combination	Course 1			Course 2			Course 3		
	Mandarin	English	Overall	Mandarin	English	Overall	Mandarin	English	Overall
(1) SI (ASTMIC + WSJ1)	52.87	-5.77	48.47	37.24	-10.72	33.55	52.48	-3.55	48.14
(2) SI (ASTMIC + TWNAESOP)	53.91	0.39	49.73	41.28	-1.55	37.96	54.37	-0.85	50.10

English words. Therefore English words were more likely to be incorrectly recognized. Row 2 are the results for applying the standard speaker adaptation (SA) cascading MLLR [51] and MAP [52], serving as the baseline of speaker adaptation scenario below. Row 3 are for speaker dependent (SD) models trained with all the training data as listed in Table 2.1 and described in Section 2.1, serving as the baseline for speaker dependent scenario below.

From Table 2.2, it is clear that the recognition accuracies for Mandarin can be significantly improved when the acoustic models were estimated by the target corpora (rows 2, 3 vs. 1). Similarly for English, but the achievable English accuracies were much lower than Mandarin, obviously because the English data in the target corpora were much less. But for speaker independent models trained with Mandarin and English corpora with very similar size (row 1), the performance difference was much smaller. This is the data imbalance issue mentioned in this thesis.

In Table 2.2, the accuracies for SI models (row 1) are quite low. One may wonder the English SI training corpus used here was specially mismatched with the target corpus. But there exists plenty of native speaker English data with Wall Street Journal (WSJ) as one example. To investigate whether other native or non-native English corpora were helpful, we used different English corpora together with the Mandarin corpus ASTMIC to build the SI models. The results are in Table 2.3, as compared to the SI models in row 1 of Table 2.2. Row 1 of Table 2.3 is for WSJ1 [53], which is in read speech and produced by native speakers. Row 2 is for TWNAESOP [54], which is also in read speech but produced by Taiwanese speakers.

We can see from Table 2.3 that for both WSJ1 and TWNAESOP, the English accuracies were very poor. The mismatch between these corpora and the English part of the target corpora was so serious that English acoustic models seemed irrelevant in the decoding process. As shown in the results for WSJ1 (row 1), it is clear that the significant difference between native and non-native English speech made the WSJ1 English corpora almost not helpful at all here. This implied adopting large amount of available English data produced by native speakers may not be a solution for this task. On the other hand, TWNAESOP was produced by Taiwanese speakers (row 2), but also highly mismatched to the target corpus here. In other words, the characteristics for speech produced by non-native speakers may vary in a very wide range and can be highly mismatched to the test speakers also. In comparison, EATMIC (the English data used in row 1 of Table 2.2) is better matched to the task considered here, so we use it as the baseline for comparison below. For training the speaker independent (SI) models in row 1 of Table 2.2 to be used in the following experiments, EATMIC containing data produced by about 400 non-native English speakers was used for training the English part. With the wide variation of characteristics in speech of non-native speakers, the SI models still gave performance for English comparable to that for Mandarin (e.g. 34.63% vs. 39.09% for course 1 in row 1 of Table 2.2).

In addition to directly combining the Mandarin and English corpora for SI model

Table 2.4: *Results of MAP Adaptation Started with the Best Set of SI Models using Different Percentages of a Large Corpus of Native English Data (WSJ1).*

SI Corpora Combination	Course 1			Course 2			Course 3		
	Mandarin	English	Overall	Mandarin	English	Overall	Mandarin	English	Overall
(1) SI (ASTMIC + EATMIC)	39.09	34.63	38.76	25.47	28.10	25.67	60.09	42.68	58.78
(2) SI (ASTMIC + EATMIC) + ADP(WSJ, 25%)	47.46	10.13	44.66	33.42	-3.12	30.68	58.82	8.92	55.08
(3) SI (ASTMIC + EATMIC) + ADP(WSJ, 50%)	50.83	0.35	47.04	36.05	-5.73	32.92	57.03	5.19	53.14
(4) SI (ASTMIC + EATMIC) + ADP(WSJ, 75%)	51.97	-1.19	47.98	37.18	-5.39	33.99	56.95	4.88	53.04
(5) SI (ASTMIC + EATMIC) + ADP(WSJ, 100%)	52.34	-1.83	48.28	37.84	-5.18	34.61	57.12	5.02	53.21

training, it is certainly possible to start with the best set of SI models trained with native Mandarin data and non-native English data (EATMIC) as in row 1 of Table 2.2 and then adapt the models with native English data (WSJ1) to take the advantage of the large quantity of the native data, for example using the well-known MAP adaptation [52]. The results are listed in Table 2.4. In Table 2.4, row 1 are the results of the best set of SI models, directly copied from row 1 in Table 2.2, and rows 2-5 are results with MAP adaptation [52] with different percentages of WSJ1 started with those in row 1. We see that in row 2, the accuracies for English part is seriously degraded with 25% of WSJ1 data (about 17 hours long) used. Note that in the adaptation only those triphones with English central phonemes were updated while those triphones with Mandarin central phonemes remained unchanged. Because the English triphones became much worse, many Mandarin words previously incorrectly recognized as English words were now correctly taken as Mandarin words. This is why the Mandarin accuracy for course 1 and 2 was improved, although Mandarin triphones remained unchanged. This trend continued or remained similar with more native English data used (50%, 75%, 100% as in rows 3-5), with results more or less saturated to values close to row 1 of Table 2.3, or the whole WSJ1 used to generate the SI model. These results showed that the gap between native and non-native English data is large even initialized with better SI models trained with non-native English data.

Chapter 3 HMM-based Cross-lingual Acoustic Modeling

A major problem considered here is the lack of guest language (English) data, the highly imbalanced data distribution, and the fact that the huge quantity of available English data produced by native speakers may not be helpful [1,41]. However, there are much more host language (Mandarin) data and there exist many similar acoustic signal segments between the two languages. Therefore, it is a good idea to try to merge the English units with some similar Mandarin units by putting together the training data for the corresponding similar units and jointly train the corresponding models for both languages. Note that although many Mandarin phonemes sound very different from English phonemes, lower level units (e.g. HMM states or Gaussians) of the two languages may be much more similar than on the phoneme level. This is why merging on lower level units makes sense. Such unit merging [12–19] approaches have been widely used previously and offered good performance improvements. Unit merging considered here can be performed on three levels of HMM models: triphone model level, HMM state level and Gaussian mixture level. The similarity between units can be found by defining proper distance measures on each level.

The concept of unit merging and recovery is illustrated in Figure 3.1, in which unit merging is applied to enhance the English models. However, when merging English units with Mandarin units, for English units the merged models obtained in this way are closer to the corresponding Mandarin units since the latter dominates the data, while for Mandarin units the model purity is actually degraded because of the disturbance by the English

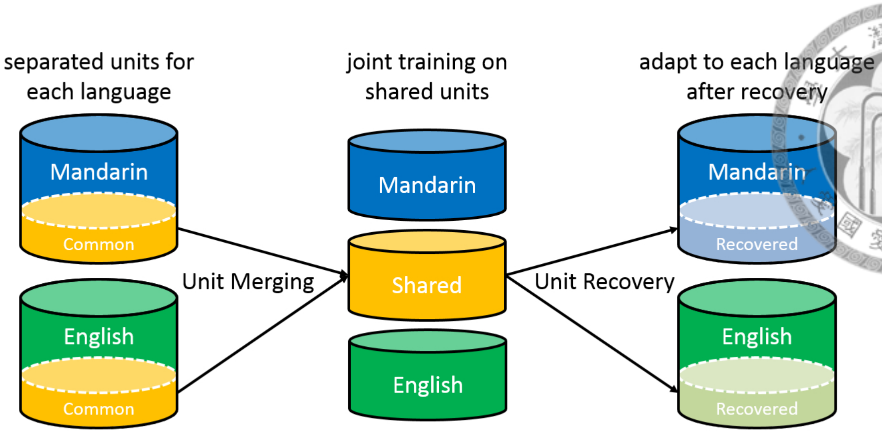


Figure 3.1: *The Concept of Unit Merging and Recovery.*

data. So here we propose in addition the extra process of unit recovery after unit merging, in which the merged units are recovered to each individual language and re-estimated again, so as to reconstruct the identity of the units in the two languages. In addition, we noticed that for both Mandarin and English the quantities of data are quite different for different units. Some units with high frequency have much more data than units with low frequency, regardless of to which language these units belong. We therefore propose to divide the units into weak units (with insufficient data) and strong units (with sufficient data) based on the accumulated occupancy of the HMM states regardless of the language, so weak units can be merged and recovered with strong units regardless of whether they belong to the same or different languages. In this way, data sharing becomes possible both across languages and within each language.

The overall block diagram of the proposed unit merging and recovery is illustrated in Figure 3.2. We begin with a set of full state-tied triphone models trained with the complete bilingual training data and based on the complete bilingual phoneme set. This is in fact the acoustic models used in the baseline system as described in the beginning of Chapter 1, and is referred to as “Acoustic Models (Full)” as shown in the block (A) of

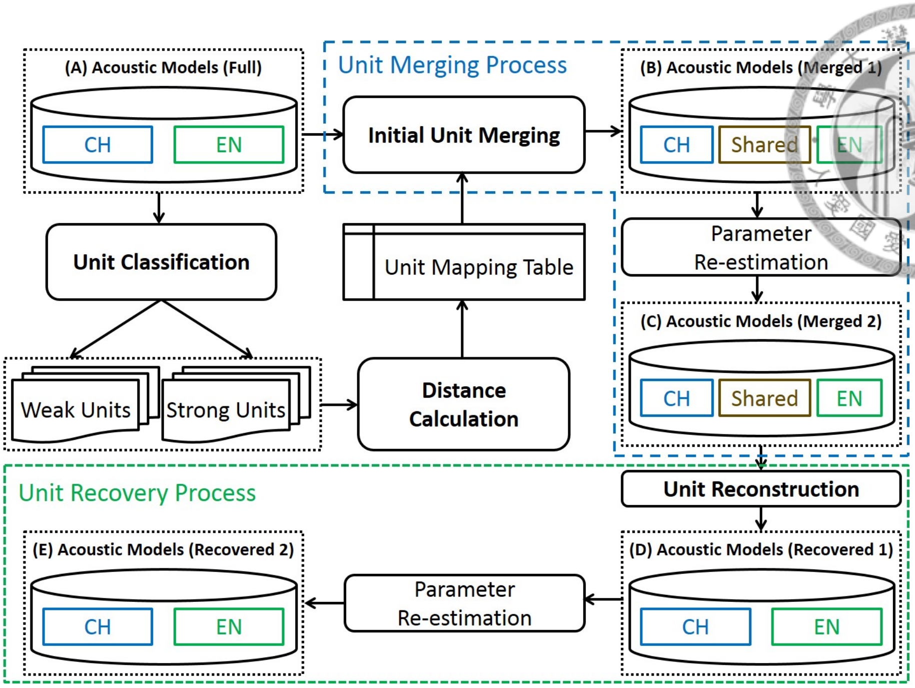


Figure 3.2: *Complete Processes of Acoustic Unit Merging and Recovery.*

Figure 3.2. Here those parts in the figure indicated by “CH” and “EN” represent those triphone models with central phonemes in Mandarin and English, respectively, although phonemes of different languages can appear in the context of the central phoneme on both sides. All acoustic units (model, state or Gaussian) are then collectively divided into weak units and strong units through the unit classification block. A straightforward approach is that all units for guest language or English are weak, while all units for host language or Mandarin are strong, although better principles for classifying the units based on accumulated occupancy for HMM states, referred to as unit occupancy ranking, will be presented later in this chapter.

With the lists of weak units and strong units, distance calculation is performed between each weak unit and all strong units on all levels (model, state and Gaussian). The details of this distance calculation on different levels are explained in the next section. The calculated distances give the mapping table, which tells the closest strong unit with

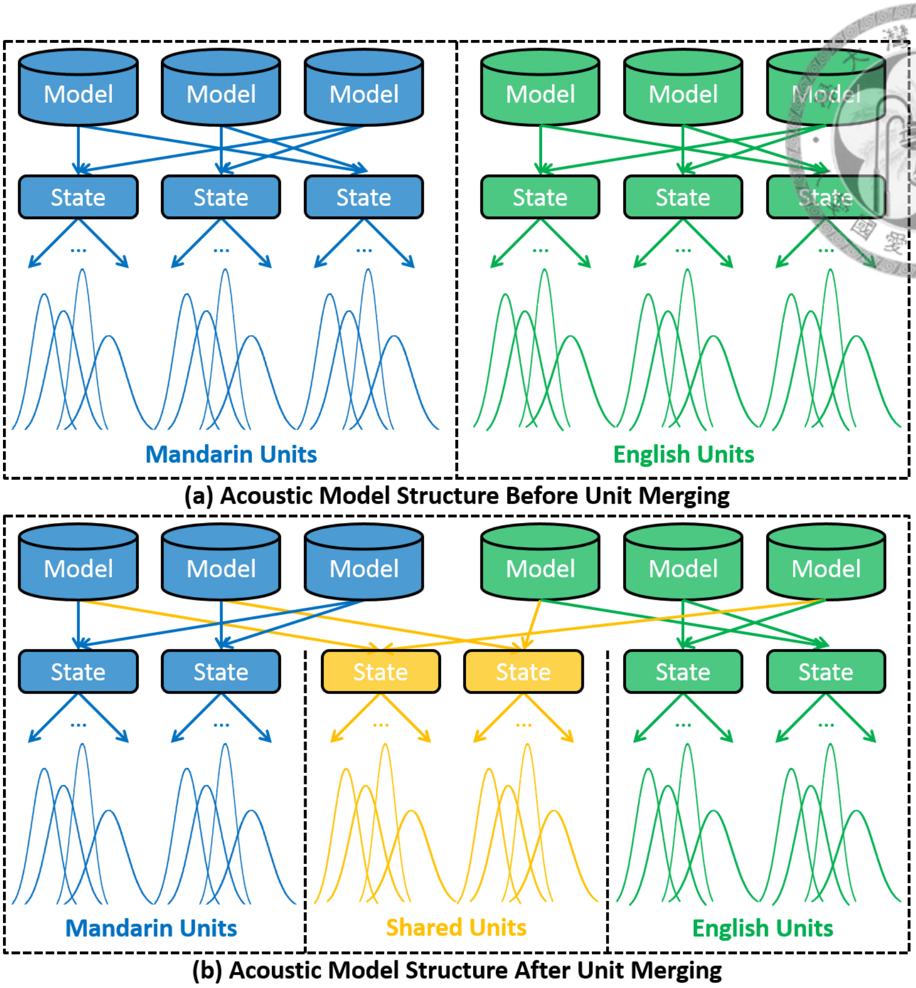


Figure 3.3: *Acoustic Model Structure (a) Before and (b) After Unit Merging on State and Gaussian Levels.*

minimum distance for each weak unit, or the strong unit each weak unit should be merged with. This can be a many-to-one relationship, because several weak units may be merged with the same strong unit. The rest of Figure 3.2 can be divided into two processes, unit merging and unit recovery, both of which will be explained below.

3.1 Unit Merging

The unit merging process first produces a set of “shared units”, as shown in block (B) and labeled as “Acoustic Models (Merged 1)” in Figure 3.2, in which the “shared units” are those produced when each weak unit is merged with the corresponding strong unit of minimum distance. Note that although Mandarin speech and English speech sound quite different, some Mandarin phonemes sound similar to and have similar acoustic characteristics to some English phonemes. In addition, the similarity between units can be higher on lower levels (HMM states and Gaussians). This is why we tried to merge acoustic units on the phoneme level and lower as well.

The parameters for all “shared units” in ”Acoustic Models (Merged 1)” are then re-estimated using the corresponding shared training data in the re-estimation process. All re-estimation processes mentioned here is direct re-training with maximum likelihood estimation in speaker dependent case, and a cascade of MLLR [51] and MAP [52] in speaker adaptation case. This gives the set of acoustic models in the block (C) labeled as ”Acoustic Models (Merge 2)” in Figure 3.2. The results of this merging process on HMM state and Gaussian level are shown in Figure 3.3. Figure 3.3(a) illustrates the units for Mandarin and English in ”Acoustic Models (Full)” of block (A), while Figure 3.3(b) for ”Acoustic Models (Merged 2)” of block (C) with a set of shared units (states and Gaussians). Here a triphone model for Mandarin/English refers to one with the central phoneme belonging to Mandarin/English, although the phonemes in the context of both sides can belong to any language. In Figure 3.3(a), no cross-lingual sharing is allowed, which gives relatively poor modeling for weak or guest language units due to the data imbalance problem. In Figure 3.3(b), after merging, some similar units for Mandarin and

English are merged to form cross-lingual shared units. Note that here shared Gaussians are linked to and used by higher level units (states and models), and are estimated by training data with both Mandarin and English labels. Similarly shared states are linked to models for both languages.

3.2 Unit Recovery

The unit recovery process then follows the unit merging process in Figure 3.2. Although the above unit merging process reduces the impact of insufficient data, the merged units tend to be closer to the strong units (or those for the host language) than the weak units (or those for the guest language), because the former dominates the data. The strong units are also disturbed by the data of weak units, assuming these units are not exactly the same. This limits the achievable likelihood for the corresponding signal segments given the merged units, especially for the weak or English units.

The solution for this problem proposed here is to first reconstruct the merged units for both languages, by copying all parameters from the merged units, and then applying an additional run of parameter re-estimation using the corresponding training data for each respective language. This is illustrated in Figure 3.2, where the unit reconstruction gives the set of ”Acoustic Models (Recovered 1)” in block (D) which does not include the ”shared units” any longer. The parameter re-estimation then gives the final set of ”Acoustic Models (Recovered 2)” in block (E). In the last re-estimation process, parameters of all units for both languages can be estimated toward their own maximum likelihood based on their own labeled data. This last parameter re-estimation gives better models, because the data insufficiency problem is properly taken care of by unit merging (so this last parameter re-estimation is better initialized), and the identity of each individual unit is further recovered afterwards.

3.3 Distance Calculation Between Acoustic Units on Different Levels

In the unit merging and recovery process shown in Figure 3.2, we need to find the closest strong unit with minimum distance for each weak unit, so they can be first merged and then recovered. Such a unit mapping table is based on the distance calculation between two units on all levels, model, state and Gaussian. This problem has been analyzed with good approaches proposed previously [12–15,15–19]. They are briefly summarized in this chapter for completeness purposes.

Knowledge-based approaches such as those based on IPA [55] and SAMPA [56] have been very useful in finding the similarities or distances between higher level units such as phonemes. These approaches are independent of data available or the models used. But the results obtained are not quantitative. For example, the phoneme /a/ in Mandarin is close to /AA/ in English, and the phoneme /b/ in Mandarin is close to /B/ in English. But it is difficult to decide which of the above two pairs have a smaller distance. In addition, distances between lower level units such as HMM states or Gaussians are difficult to estimate with human perception alone.

On the other hand, data-driven approaches for distance calculation rely much more on the available data, but can be used with different models, different speakers and different languages. The major problems of data-driven approaches are the distances obtained become unreliable when available data is insufficient.

For better estimate of inter-lingual unit distances, it has been proposed [12–19,32,33] that data-driven methods used with knowledge-based high level constraints is a good compromise to integrate the above two approaches. In other words, distances calculated in data-driven ways but only within the same acoustic class defined by linguistic knowledge have been shown to be useful and reliable. We follow this direction in this thesis. Both Mandarin and English phoneme sets are divided into 4 classes based on the IPA notations, i.e. the plosives, affricates, voiced consonants and vowels, and data-driven distance between units are calculated only within the same class.

In addition, quite several different data driven approaches have been reported previously to evaluate the distance between HMM models, states and Gaussians [12,13,16,17, 32,33], some are more precise and require much more computation resources and some are relatively simple. It has been shown that for the purposes here some of the more precise approaches did not necessarily offer significantly different results than the relatively simple ones [16,17]. Similar situations were also observed in the preliminary experiments performed in this work, which is why we choose to use relatively simple data-driven approaches to evaluate the distances as summarized below.

3.3.1 Model Level Distance

Phoneme is the minimum unit of sound in a language perceivable by human, while triphone is a better model for phonemes trainable by machines. This is why triphone model merging makes sense [11,12,17,28,32]. Because the training data may be insufficient for many triphones, model-based calculation of similarity is preferred here. Moreover, the

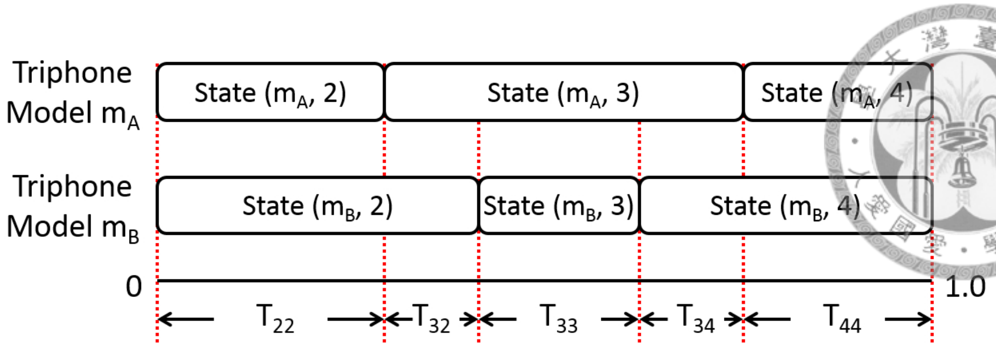


Figure 3.4: *Distance Calculation between Triphone Models m_A and m_B Based on State Alignment.*

length of the signal segments corresponding to each triphone can be very different and so are those of each HMM state within each triphone. Here we tried to align the HMM states of two triphone models so we can evaluate the similarity between them by considering the overlapping of HMM states. Here a model-based distance between two triphone models is defined. First, for each triphone model, an expected state duration $Dur(S_i)$ in number of frames is estimated for each state S_i directly using transition probabilities without considering observation sequences,

$$Dur(s_i) = \sum_{n=1}^{\infty} n[(a_{i,i})^{n-1} a_{i,i+1}] = \frac{a_{i,i+1}}{(1 - a_{i,i})^2}, \tag{3.1}$$

where $a_{i,i}$ is the transition probability from state S_i to state S_j and $[(a_{i,i})^{n-1} a_{i,i+1}]$ is the probability that state S_i lasts for n frames, assuming $a_{i,i} + a_{i,i+1} = 1.0$, or the probability for transiting from S_i to states other than S_i and S_{i+1} is negligible. This expected duration is then further normalized such that the total duration for each triphone model is always 1.0,

$$Dur_n(S_i) = \frac{Dur(S_i)}{\sum_j Dur(S_j)}, \tag{3.2}$$

where the denominator is the summation over all states in the triphone model. This normalized duration can then be used in aligning two triphone models below.

Figure 3.4 is an example demonstrating the alignment for distance evaluation between two triphone models m_A and m_B, each with three states 2, 3 and 4 (assuming states 1 and 5 are entry and exit states). In Figure 3.4, both triphones m_A and m_B have a normalized duration of 1.0, and T_{ij} represents the duration percentage for the overlapped portion for state i of triphone m_A with state j of triphone m_B, and is used as the weight for the distance between the corresponding states. When evaluating the distance between two states, every state is modeled by a single Gaussian, and the distance between two states is defined as the symmetrical KL divergence between the two Gaussians [16,17,21,31]. So the distance $D_M(m_A, m_B)$ between two triphone models m_A and m_B is estimated as,

$$D_M(m_A, m_B) = \sum_{i \in m_A} \sum_{j \in m_B} T_{ij} D_{KL}(G_i, G_j), \tag{3.3}$$

where $D_{KL}(G_i, G_j)$ is the symmetrical KL divergence between the single Gaussians representing the states S_i, S_j in models m_A and m_B. Note that the distances estimated here in (3.3) are certainly not very accurate, but simply serving as an easy reference to be used here. It is well known that the duration estimation by transition probabilities is not very good as in (3.1). Modeling each state with a single Gaussian is not very good either as in (3.3).

3.3.2 State Level Distance

There are certainly limitations in merging triphone models. English and Mandarin are quite different in acoustic nature with quite different phoneme sets. So forced merging of distinct triphone models may not be very smooth. On the other hand, HMM states represent sequential components of phonemes, with statistically steady distribution for acoustically similar feature vectors, usually considered corresponding to a certain stage of vocal tract activities. HMM states cannot be perceived by human, but can be well identified by machine. Although speech production can be very different for many different languages, it is always limited by the physical structure and movement of human vocal tract, which is to a certain degree reasonably represented by the HMM states. So HMM states may be a better unit universal across all languages. This is why states have been used in unit merging [15–17,31,33,57,58], in which the distance between two states is simply the symmetrical KL divergence [17,57], with each state modeled by a single Gaussian,

$$D_S(S_i, S_j) = D_{KL}(G_i, G_j), \tag{3.4}$$

where G_i is the single Gaussian that models the state S_i. Although using a single Gaussian to model each state seems not accurate enough, and there exist several ways to estimate the KL Divergence between Gaussian mixture models [59,60], the computation time needed for these methods was much higher while no significant difference was observed in preliminary experiments. Similar results were also reported earlier [16]. Therefore, in the state level distance calculation here, a single Gaussian is used to model each state. In this way, for every weak (or English) state, a strong (or Mandarin) state with minimum distance can be found.

3.3.3 Gaussian Level Distance

Since Gaussian mixtures represent the fine structure of the HMM states, merging between Gaussians is certainly possible [16,17]. The distance between two Gaussians G_i and G_j is simple, using the symmetrical KL divergence,

$$D_G(G_i, G_j) = D_{KL}(G_i, G_j), \tag{3.5}$$

However, note that in this way two very similar Gaussians simply represent similar local statistical distributions within the feature space which is jointly modeled by many Gaussians. Though the physical interpretation with respect to speech feature distribution is weaker for Gaussians than states or models, Gaussian-level unit merging can be helpful due to the fine structure it represents.

3.4 Unit Occupancy Ranking for Unit Classification

In Figure 3.2, unit classification is first performed to divide the acoustic units into weak (with insufficient data) and strong (with sufficient data) units, and then for each weak unit we find a strong unit with minimum distance to merge with it. In the previous sections 3.1 and 3.3, we simply assume the English units are weak and Mandarin units are strong, but this is not necessarily true, because in each language there are high frequency units and low frequency units. In other words, actually some units of the guest language (English) may have sufficient training data, while some of the host language (Mandarin) may not. This will be verified later on using the methods for evaluating the data sufficiency introduced below. Therefore, taking the distribution of actually available training data into consideration in the unit merging procedure is essential.

In the standard training procedure of HMMs, the accumulated occupancy of each state with respect to the training data can be obtained when running the forward-backward algorithm with the training data and the given model configuration as below.

$$Occ(S_i) = \sum_r \sum_{t=1}^{T_r} \frac{1}{P_r} \alpha_i^r(t) b_i^r(o_t) \beta_i^r(t), \tag{3.6}$$

where S_i is a state, P_r is the posterior probability of utterance r given the observation sequence $o_1...o_{T_r}$ and the corresponding label, $\alpha_i^r(t)$ and $\beta_i^r(t)$ are respectively the forward and backward probabilities for state S_i at time t , and $b_i^r(o_t)$ is the likelihood for the observation o_t given the state S_i, all for utterance r. Here we utilize the above accumulated occupancies for states as a good indicator for the availability of training data for the states, since they are positively related to the quantity of the available training data for the states [21,50,58].

When training the "Acoustic Model (Full)" of block (A) in Figure 3.2 with forward-backward algorithm using all the available training data, a list of accumulated occupancies for each state in the models is obtained. This list is sorted according to the values of accumulated occupancies from low to high. By defining a threshold, all state with accumulated occupancy below the threshold can be defined as "weak states", while those above the threshold as "strong states". For the Gaussian level of units considered in this thesis, the accumulated occupancies for each Gaussian can be estimated using the Gaussian weights in the GMM structure,

$$Occ(G_{ij}) = w_{ij} Occ(S_i), \tag{3.7}$$

where S_i is a state, $Occ(S_i)$ the accumulated occupancy for state S_i given the training corpus as in (3.6), G_{ij} the jth Gaussian in S_i and w_{ij} the weight for G_{ij}. In this way, all Gaussians used in either the host or the guest languages can also be sorted according to this accumulated occupancy to produce a sorted list to define the weak and strong Gaussians.

For the model level, a model is composed of several tied states each with different accumulated occupancies, therefore the available data for training a model is difficult to define. So the unit classification into weak and strong units based on occupancy can be performed on state and Gaussian levels only, but not the model level. Each triphone model is classified as weak or strong simply according to the language its central phoneme belongs to.

In this way, the mapping relationship is no longer limited to cross-lingual manner for state and Gaussian level. With the occupancy ranking for unit classification, not all weak units are for English and not all strong ones are for Mandarin. Mandarin states and Gaussians with accumulated occupancy lower than the threshold may be merged with either Mandarin or English states and Gaussians depending on the calculated distances. Some weak English states or Gaussians hardly finding similar units in Mandarin may also benefit from being merged with similar English units with sufficient data, rather than being forced to be merged with Mandarin units, etc.

Note that in training triphone models for a monolingual task, the state-clustering technique with decision trees [50] has been widely used, which is also capable of handling data sufficiency issue to a good extent. It is possible to adjust the threshold for the state-clustering technique to ensure enough training data for each state, although the threshold adjustment for each individual node in the decision tree can be very complicated. However, in these cases, separated decision trees are constructed for each individual state for each individual central phoneme. Therefore, no data sharing is allowed either across different central phonemes or across different states for the same central phoneme (e.g. the first state and the second state of the triphones with the same central phoneme cannot share data because they are managed by different trees). With the unit merging and recovery techniques proposed above, data sharing becomes possible across different trees but have to be across different languages, i.e., an English unit (Gaussian, state or model) can use Mandarin data for any state in any triphone with any central phoneme, as long as they are close, but they have to be for Mandarin. In contrast, for the unit occupancy ranking considered here, we note that in the bilingual scenario, some units (specially on lower levels such as Gaussian or state) in one language may also be very close to some other units within the same language, so it is reasonable to make sharing between them possible too. With the additional unit occupancy ranking proposed here, we now allow data sharing across all states for all central phonemes, either across languages or within the same language. So an English unit can use data for any state in any triphone with any central phoneme in both languages, as long as they are close. So the data-sharing becomes much more flexible.

3.5 Unit Occupancy Analysis

To find out whether the above unit classification concept is really useful for real data, the histogram for the accumulated occupancies on the Gaussian level for the training set of course 1 listed in Table 2.1 (9.10 hours long) for the corresponding speaker dependent

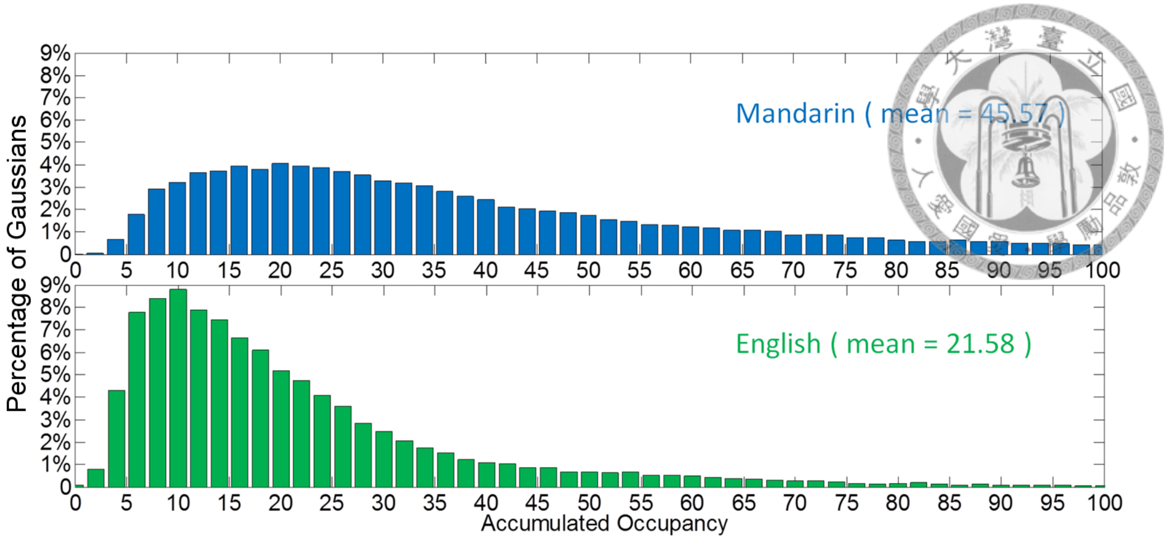


Figure 3.5: *Histogram of Accumulated Occupancies for Mandarin and English Gaussians with the Training Set of Course 1.*

model are shown in Figure 3.5, where the horizontal axis is the accumulated occupancy obtained as in (3.7) and the vertical axis represents the percentage of Gaussians with the corresponding accumulated occupancy. It is clear from this figure that in general the accumulated occupancies of the majority of Gaussians for English (guest language) are lower and those for Mandarin (host language) are higher. But the accumulated occupancies ranged widely for the Gaussians, there certainly exist good numbers of Mandarin Gaussians with lower accumulated occupancies and English Gaussians with higher accumulated occupancies. The results we have here are highly imbalanced due to the nature of the bilingual code-switching speech and highly dependent on the data sets used. However, for code-switching speech as considered here, these results verified that simply assuming all English units are weak and all Mandarin units are strong, as was done previously [17], may not be the best approach.

Chapter 4 Experimental Results for HMM-based Cross-lingual Acoustic Modeling

In this chapter, experimental results using the HMM-based approaches proposed in Chapter 3 are reported and discussed. Results for unit merging (Section 3.1) are listed in Section 4.1, those for unit recovery (Section 3.2) are in Section 4.2, those for unit classification with occupancy ranking (Section 3.4) are in Section 4.3, respectively.

4.1 Unit Merging on Different Levels (without Unit Recovery and Occupancy Ranking)

Experimental results in accuracies for different versions of acoustic models obtained with unit merging only (without unit recovery and occupancy ranking, simply finding a Mandarin unit for each English unit to merge with, labelled ”MRG”) are listed in Table 4.1. Rows 1-5 are for speaker adapted (SA) models while rows 6-11 are for speaker dependent (SD) models. Rows 1 and 6, labeled as ”(Full, ADP)” and ”(Full)” are for the baseline acoustic models directly copied from rows 2 and 3 of Table 2.2, used as baselines in the significance tests. The results for the proposed acoustic unit merging approach on different levels are respectively listed in rows 2 to 5 for SA case and rows 7 to 10 for SD case, with merging on model level in rows 2, 3, 7 and 8 (rows 2 and 7 based on IPA as described below), on state level in rows 4 and 9 and on Gaussian level in rows 5 and 10.

In order to compare the proposed approach with the knowledge-based model merging approach base on IPA table, we built such IPA-based merged models, with results

Table 4.1: *Results of Unit Merging (MRG) on Levels of Model, State and Gaussian (Accuracies) (%).*

Acoustic Models	Course 1				Course 2				Course 3			
	Mandarin	English	Overall	*p*-value	Mandarin	English	Overall	*p*-value	Mandarin	English	Overall	*p*-value
(1) SA (Full, ADP)	75.75	51.95	73.96	--	70.71	63.28	70.15	--	77.21	52.83	75.32	--
(2) SA (MRG, Model-IPA)	74.23	47.15	72.20	1.00	69.52	58.24	68.65	1.00	77.02	48.25	74.79	1.00
(3) SA (MRG, Model)	75.82	52.81	74.09	0.094	70.80	64.66	70.34^{+}	0.032	77.26	52.87	75.37	0.18
(4) SA (MRG, State)	75.88	54.22	74.26^{+}	5.4e-3	70.89	66.84	70.59^{+}	7.6e-3	77.42	54.72	75.66^{+}	6.4e-3
(5) SA (MRG, Gaussian)	75.97	55.87	74.46^{+}	8.2e-4	70.92	67.53	70.67^{+}	1.7e-3	77.64	55.40	75.92^{+}	5.5e-4
(6) SD (Full)	83.62	61.87	81.99	--	75.62	71.63	75.32	--	82.87	62.58	81.30	--
(7) SD (MRG, Model-IPA)	83.90	61.38	82.21^{+}	0.007	75.60	72.88	75.39	0.098	83.15	62.91	81.58	0.02
(8) SD (MRG, Model)	83.71	63.04	82.16^{+}	0.018	75.66	71.83	75.37	0.127	83.02	62.75	81.45^{+}	0.03
(9) SD (MRG, State)	83.98	64.08	82.49^{+}	4.5e-5	75.70	73.70	75.55^{+}	6.2e-3	83.31	64.52	81.86^{+}	5.7e-5
(10) SD (MRG, Gaussian)	84.25	69.00	83.11^{+}	2.7e-13	75.93	75.39	75.89^{+}	1.9e-4	83.81	68.15	82.60^{+}	1.6e-14
(11) SD (Combination)	83.92	63.30	82.37^{+}	9.2e-4	75.76	73.19	75.37	0.103	-	-	-	-

listed in rows 2 and 7 (labeled as ”MRG, Model-IPA”). In this IPA-based model-level merging, 28 English phonemes were directly merged with the corresponding Mandarin phonemes having the same IPA symbols. In row 2 for SA case, we see no improvement was brought by this knowledge-based method. Only very slight improvement can be observed even for SD case as in row 7. A possible explanation is that phoneme similarities vary significantly from speaker to speaker, therefore for the target speakers producing the target corpora, the best unit mapping table is not necessarily the same as the one found by IPA. Mandarin and English are quite different on the phoneme level too.

On the other hand, we can see that the accuracies (particularly for English) were significantly improved by the proposed acoustic unit merging approach in all cases, except for merging on model level for SA case for courses 1 and 3 (with p-values of 0.094 and 0.18 in row 3), and SD case for course 2 (with p-value of 0.127 in row 8).

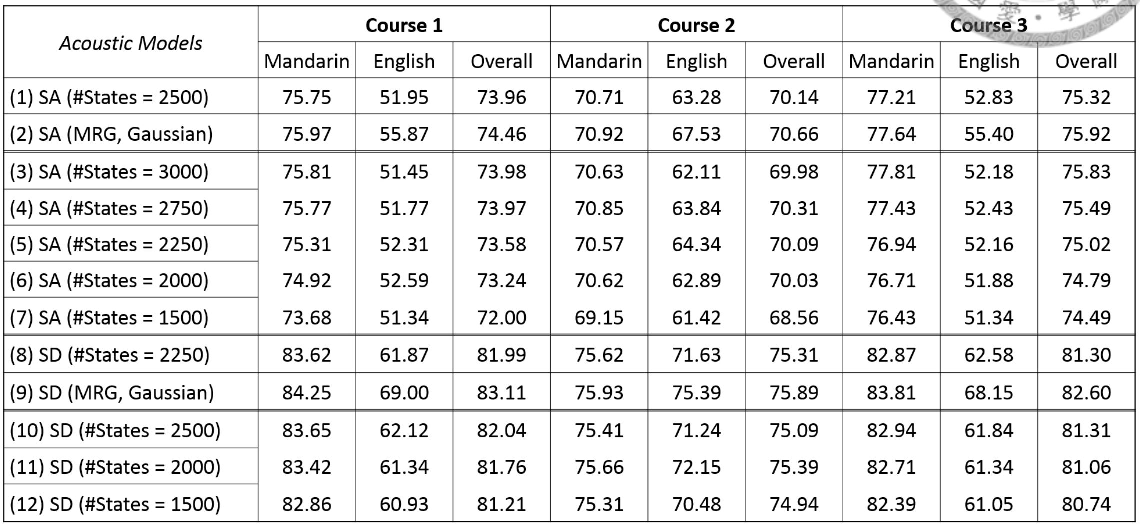
In general, the Gaussian level merging (rows 5 and 10) was better than state level (rows 4 and 9), which is in turn better than model level (rows 2, 3, 7, 8), or the lower (and finer) level merging was better. In addition, note that the performance of Mandarin part was not degraded while the performance of English part was improved in all cases. Comparing the results for speaker adapted models with speaker dependent models (upper to lower halves), the improvements for speaker dependent models were larger, although their English data were less insufficient. This implies more data gives better unit mapping, which is one of the advantages of the data-driven approaches.

In order to verify that the proposed approaches are better than directly adding the data, we performed an extra experiment in SD case with models trained with the combination of the training sets of the two courses 1 and 2. The results are listed in row 11 (labeled as ”Combination”). Since the speakers of the two training sets are the same as mentioned in Chapter 2, the results are reasonably expected to be better than the speaker dependent baseline in row 6. Compared with the best results obtained with the approaches proposed here, unit merging on Gaussian level as in row 10, we see simply adding more data did bring improvements, but much less than unit merging.

An additional concern here is that during the standard training process for the state-tied acoustic models using decision trees, the splitting threshold can be adjusted to modify the size of the acoustic models, based on which the data sharing among different units can be properly managed. With higher thresholds, the total number of HMM states is decreased and each HMM state is share by more triphone models, or more training data are available for each state.

However, in such cases, separate decision trees are constructed for each individual state for each individual central phoneme. Therefore, no data sharing is allowed either

Table 4.2: *Results for Unit Merging Compared with Models Trained with Different Splitting Thresholds.*

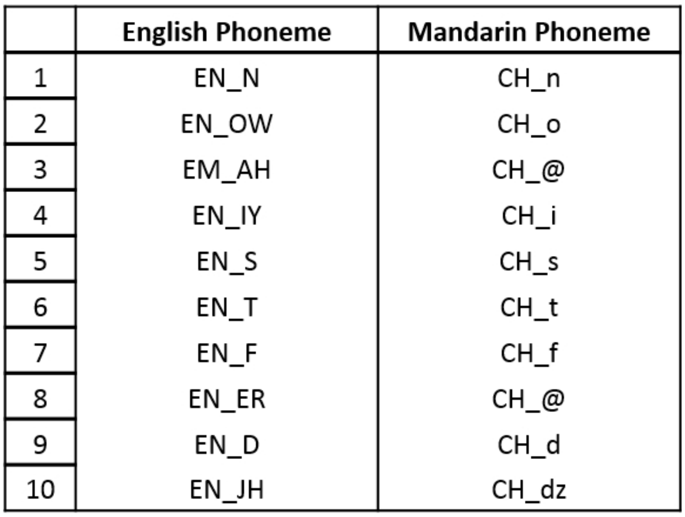


Acoustic Models	Course 1			Course 2			Course 3		
	Mandarin	English	Overall	Mandarin	English	Overall	Mandarin	English	Overall
(1) SA (#States = 2500)	75.75	51.95	73.96	70.71	63.28	70.14	77.21	52.83	75.32
(2) SA (MRG, Gaussian)	75.97	55.87	74.46	70.92	67.53	70.66	77.64	55.40	75.92
(3) SA (#States = 3000)	75.81	51.45	73.98	70.63	62.11	69.98	77.81	52.18	75.83
(4) SA (#States = 2750)	75.77	51.77	73.97	70.85	63.84	70.31	77.43	52.43	75.49
(5) SA (#States = 2250)	75.31	52.31	73.58	70.57	64.34	70.09	76.94	52.16	75.02
(6) SA (#States = 2000)	74.92	52.59	73.24	70.62	62.89	70.03	76.71	51.88	74.79
(7) SA (#States = 1500)	73.68	51.34	72.00	69.15	61.42	68.56	76.43	51.34	74.49
(8) SD (#States = 2250)	83.62	61.87	81.99	75.62	71.63	75.31	82.87	62.58	81.30
(9) SD (MRG, Gaussian)	84.25	69.00	83.11	75.93	75.39	75.89	83.81	68.15	82.60
(10) SD (#States = 2500)	83.65	62.12	82.04	75.41	71.24	75.09	82.94	61.84	81.31
(11) SD (#States = 2000)	83.42	61.34	81.76	75.66	72.15	75.39	82.71	61.34	81.06
(12) SD (#States = 1500)	82.86	60.93	81.21	75.31	70.48	74.94	82.39	61.05	80.74

across different central phonemes or across different states for the same central phoneme, since they are all managed by different trees. With the unit merging proposed here, an English state (or its Gaussian) can share data with any Mandarin state (or its Gaussian) for any central phoneme, as long as they are close. This makes data sharing much more flexible. This was verified by an extra experiment in which the triphone models were trained with different thresholds without unit merging, to be compared with those with unit merging. The results are shown in Table 4.2.

In Table 4.2, rows 1-7 are for SA cases and rows 8-12 for SD cases. Rows 1 and 8 are directly copied from rows 1 and 6 of Table 4.1 without unit merging, serving as the baselines here, except it was marked that 2500 states and 2250 states were used in rows 1 and 8 respectively when generating these results. Rows 2 and 9 are then the best results with unit merging on Gaussian level, which are directly copied from rows 5 and 10 of

Table 4.3: *Monophone-level Mapping Relationship between Mandarin and English for Model-level Merging in SD case and Course 1 with the Proposed Approach.*



	English Phoneme	Mandarin Phoneme
1	EN_N	CH_n
2	EN_OW	CH_o
3	EM_AH	CH_@
4	EN_IY	CH_i
5	EN_S	CH_s
6	EN_T	CH_t
7	EN_F	CH_f
8	EN_ER	CH_@
9	EN_D	CH_d
10	EN_JH	CH_dz

Table 4.1 for comparison. Rows 3-7 and 10-12 are then results with different splitting thresholds ending up with different number of states ranging between 1500 and 3000, all without unit merging. We see that the accuracies varied with different thresholds for different cases, some of which were better than 2500 states in row 1 or 2250 states in row 8. However, the improvements achievable by adjusting the threshold in rows 3-7, 10-12 are much less than that achievable with unit merging on Gaussian level in rows 2 and 9. As discussed above, much more flexible data sharing is offered by unit merging across the languages.

Table 4.3 is an example of monophone-level mapping table for speaker dependent (SD) model-level merging obtained for course 1. Note that because the real model-level merging was performed on triphones, so for each English phoneme in the left column of Table 4.3, different context may lead to different Mandarin phonemes to be merged with. So only the one with the highest count is shown on the right column of Table 4.3.

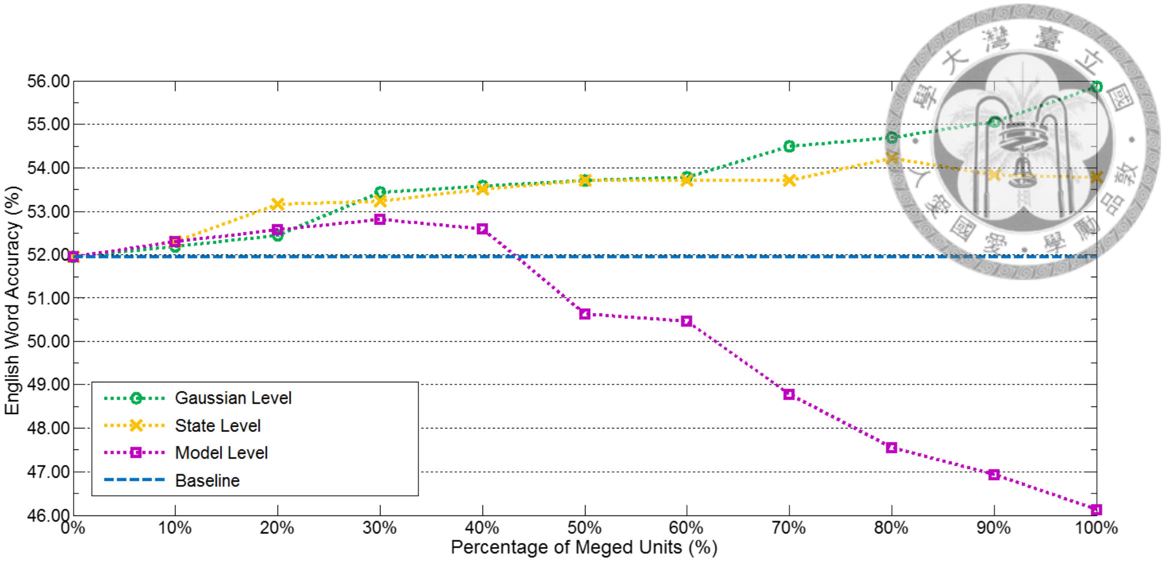


Figure 4.1: *English Word Accuracies for Unit Merging on Different Levels with Different Percentages of Merged Units.*

These mapping pairs are ranked according to the minimum cross-lingual distance, and only the top 10 pairs are listed in Table 4.3. We can see that the mapping relationship was reasonably consistent with the knowledge offered by the IPA table. However, we noticed that the mapping relationship between English units and Mandarin units differed for different speakers (e.g. course 3 vs. course 1). This implies a good amount of training data for the target speaker is necessary for estimating the mapping table, which was about 30 minutes for SA cases and 9 hours for SD cases. Otherwise the mapping may not be accurate enough and the performance may be poor. With the proposed approaches, more detailed mapping relationship for triphones and on lower levels can be obtained in a data-driven way.

In fact, because the mapping pairs are ranked according to the distance, we can choose to merge only a given percentage of English units with the corresponding Mandarin units, but not all. The English word accuracies obtained in this way for different percentages of English units merged on different levels under the speaker adaptation (SA) scenario for course 1 is shown in Figure 4.1, compared to the baseline results without unit merging.

In Figure 4.1 for model level, we can see that the best performance was achieved when 30% of English triphones were merged with Mandarin triphones. When this percentage exceeded 50%, the accuracies became worse than the baseline. This is reasonable since Mandarin and English are quite different languages in nature, thus forced merging of distinct triphones led to performance degradation.

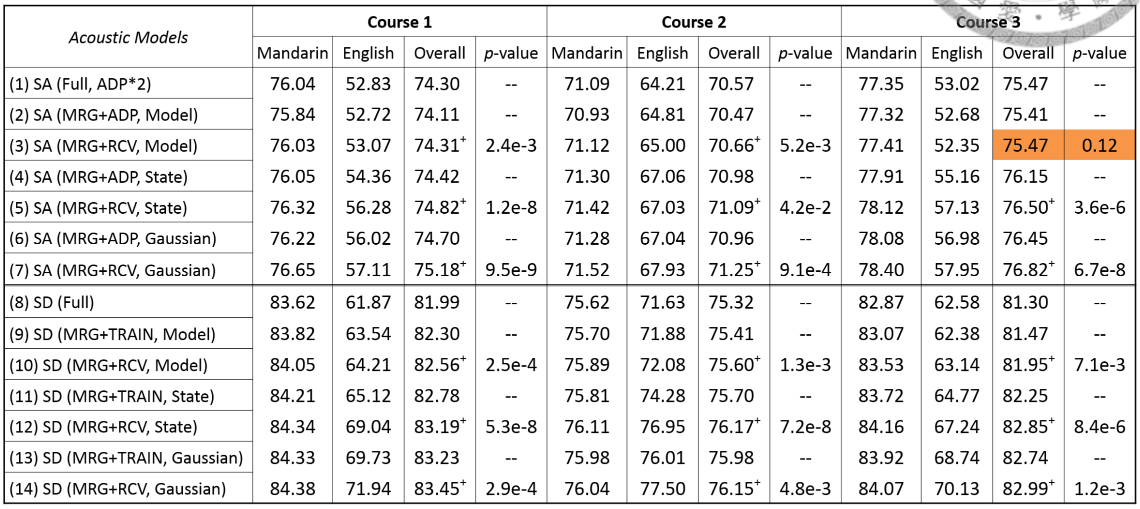
For lower levels or finer units in Figure 4.1, state or Gaussian, the accuracies were continuously improved with higher merging percentage. The results in Table 4.1 are actually the best results obtained with a chosen percentage, i.e., 30% for model level in rows 3, 8, 80% for state level in rows 4, 9, and 100% for Gaussian level in rows 5 and 10.

4.2 Unit Recovery on Different Levels (without Occupancy Ranking)

The results with unit recovery process after the unit merging as shown in Figure 3.2 are listed in Table 4.4 for SA (rows 1 to 7) and SD (rows 8 to 14).

Since there is another parameter re-estimation block in the unit recovery process, the parameters of the models labeled as ”Acoustic Models (Recovered 2)” in block (E) of Figure 3.2 actually had been re-estimated twice, one during unit merging and the other during unit recovery. Therefore, the models obtained in Block (E) should be compared with models with parameters also re-estimated twice. So a set of speaker adapted models

Table 4.4: *Results for Unit Merging (MRG) followed by Unit Recovery (RCV) on Different Levels (Model, State, Gaussian) (Accuracies) (%).*



Acoustic Models	Course 1				Course 2				Course 3			
	Mandarin	English	Overall	*p*-value	Mandarin	English	Overall	*p*-value	Mandarin	English	Overall	*p*-value
(1) SA (Full, ADP*2)	76.04	52.83	74.30	--	71.09	64.21	70.57	--	77.35	53.02	75.47	--
(2) SA (MRG+ADP, Model)	75.84	52.72	74.11	--	70.93	64.81	70.47	--	77.32	52.68	75.41	--
(3) SA (MRG+RCV, Model)	76.03	53.07	74.31^{+}	2.4e-3	71.12	65.00	70.66^{+}	5.2e-3	77.41	52.35	75.47	0.12
(4) SA (MRG+ADP, State)	76.05	54.36	74.42	--	71.30	67.06	70.98	--	77.91	55.16	76.15	--
(5) SA (MRG+RCV, State)	76.32	56.28	74.82^{+}	1.2e-8	71.42	67.03	71.09^{+}	4.2e-2	78.12	57.13	76.50^{+}	3.6e-6
(6) SA (MRG+ADP, Gaussian)	76.22	56.02	74.70	--	71.28	67.04	70.96	--	78.08	56.98	76.45	--
(7) SA (MRG+RCV, Gaussian)	76.65	57.11	75.18^{+}	9.5e-9	71.52	67.93	71.25^{+}	9.1e-4	78.40	57.95	76.82^{+}	6.7e-8
(8) SD (Full)	83.62	61.87	81.99	--	75.62	71.63	75.32	--	82.87	62.58	81.30	--
(9) SD (MRG+TRAIN, Model)	83.82	63.54	82.30	--	75.70	71.88	75.41	--	83.07	62.38	81.47	--
(10) SD (MRG+RCV, Model)	84.05	64.21	82.56^{+}	2.5e-4	75.89	72.08	75.60^{+}	1.3e-3	83.53	63.14	81.95^{+}	7.1e-3
(11) SD (MRG+TRAIN, State)	84.21	65.12	82.78	--	75.81	74.28	75.70	--	83.72	64.77	82.25	--
(12) SD (MRG+RCV, State)	84.34	69.04	83.19^{+}	5.3e-8	76.11	76.95	76.17^{+}	7.2e-8	84.16	67.24	82.85^{+}	8.4e-6
(13) SD (MRG+TRAIN, Gaussian)	84.33	69.73	83.23	--	75.98	76.01	75.98	--	83.92	68.74	82.74	--
(14) SD (MRG+RCV, Gaussian)	84.38	71.94	83.45^{+}	2.9e-4	76.04	77.50	76.15^{+}	4.8e-3	84.07	70.13	82.99^{+}	1.2e-3

obtained with two repeated processes of MLLR followed by MAP was taken as the new baseline in row 1 (labeled as ”ADP*2”). Row 8 is the same as row 3 of Table 2.2. For unit merging on different levels, after the models were obtained at the end of unit merging process before recovery, labeled as ”Acoustic Models (Merged 2)” in block (C) of Figure 3.2, another run of parameter re-estimation was also performed in addition to produce the second sets of baselines in rows 2, 4 and 6 (labeled as ”MRG+ADP”) for SA case and rows 9, 11, 13 (labeled as ”MRG+TRAIN”) for SD case respectively on model, state and Gaussian levels.

The results with unit recovery process performed after unit merging (models in block (E) of Figure 3.2, labeled as ”MRG+RCV”) are listed in rows 3, 5, 7 and rows 10, 12, 14 respectively for SA and SD cases, to be compared with those without recovery but with an addition run of parameter re-estimation in rows 2, 4, 6 and rows 9, 11, 13 respectively. Significant improvements can be observed when comparing with the respective baselines

Table 4.5: *Results When Occupancy Ranking (OCC) was used with Unit Merging (MRG) and Recovery (RCV) on Gaussian Level Only (Accuracies) (%).*

Acoustic Models	Course 1				Course 2				Course 3			
	Mandarin	English	Overall	*p*-value	Mandarin	English	Overall	*p*-value	Mandarin	English	Overall	*p*-value
(1) SA (MRG, Gaussian)	75.97	55.87	74.46	--	70.92	67.53	70.67	--	77.64	55.40	75.92	--
(2) SA (OCC+MRG, Gaussian)	76.02	56.33	74.54^+	0.038	70.99	67.81	70.75	0.082	77.82	55.67	76.11^+	0.019
(3) SA (MRG+RCV, Gaussian)	76.65	57.11	75.18	--	71.52	67.93	71.25	--	78.40	57.95	76.82	--
(4) SA (OCC+MRG+RCV, Gaussian)	76.72	58.06	75.32^+	0.025	71.62	68.43	71.38	0.052	78.35	58.26	76.79	1.00
(5) SD (MRG, Gaussian)	84.25	69.00	83.11	--	75.93	75.39	75.89	--	83.81	68.15	82.60	--
(6) SD (OCC+MRG, Gaussian)	84.63	69.33	83.48^+	1.5e-3	76.08	77.97	76.22^+	2.8e-3	89.92	68.72	82.74^+	8.1e-3
(7) SD (MRG+RCV, Gaussian)	84.38	71.94	83.45	--	76.04	77.50	76.15	--	84.07	70.13	82.99	--
(8) SD (OCC+MRG+RCV, Gaussian)	84.46	72.45	83.56^+	0.017	76.21	78.26	76.36^+	0.009	84.04	70.73	83.01	0.152

in almost all cases with p-values also listed (the only exception was for course 3 and model level for SA in row 3). The parameters of the recovered units are no longer dominated by the data from the host language, while the data insufficiency issue was already properly handled by initializing parameters in unit merging.

4.3 Unit Occupancy Ranking on Gaussian level

We now consider the unit occupancy ranking as discussed in Section 3.4 and the results are listed in Table 4.5. Since the best results of unit merging (and recovery) were obtained on the Gaussian level, we only report results on the Gaussian level.

In Table 4.5, rows 1-4 are for SA and 5-8 for SD. Rows 1, 5 are the best results of unit merging only on Gaussian level, while rows 3, 7 with unit recovery applied in addition. These rows are used as baselines for comparison and rows 2, 4, 6, 8 are respectively for those with unit merging performed with unit occupancy ranking (labeled ”OCC” in addition). Comparing rows 2, 4, 6, 8 to rows 1, 3, 5, 7, we can see that the proposed occupancy ranking approach offered significant improvements to unit merging in most cases regardless of whether unit recovery was performed or not.

Chapter 5 Frame-level Language Posterior Estimates and Experimental Results

In addition to acoustic modeling, here we further propose to estimate the frame-level language posteriors, which is then used in the decoding process. Language identification for the code-switched utterances considered here is much more difficult than the conventional language identification task, because the languages are switched back and forth between words within an utterance [31,33,35]. Since there can be more than one language switching boundaries within an utterance, it is difficult to identify proper signal segments for language identification.

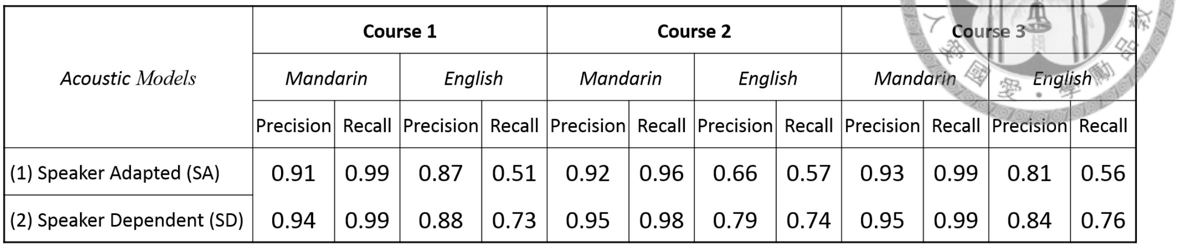
On the other hand, the information regarding which language each frame of signal belongs to is critical here. The recognizer always tends to take every signal segment as belonging to the host language, because not only the acoustic models for the host language are better trained with more data and therefore better fitted to the signals and give higher likelihoods, but the language model almost always gives higher prior probabilities to the host language words.

In code-switched speech considered here, languages are switched between words within an utterance. As a result, the ideal unit for language identification seems to be the word. However, word boundaries in an utterance are not available before recognition, or the word boundary estimates obtained during recognition can be highly unreliable. Therefore, here we proposed to estimate the posteriors for each language frame by frame, and use these posteriors in decoding. But it is certainly very hard to estimate which language a single frame of signal belongs to. Therefore this frame-level language posterior should be estimated based on much longer signal segments than a frame. This leads to the use of neural networks with input features based on longer context, and the newly proposed blurred posteriorgram features (BPFs) extracted from lattices.

5.1 Frame-level Language Identification by Baseline System

As introduced in Section 1.2, the baseline system is already capable of transcribing the bilingual code-switched utterances. By comparing the recognition results with the reference transcriptions frame by frame, we can obtain the frame-level language identification performance of the baseline system. For example, percentage of frames recognized as belonging to a language actually belonging to that language in the reference transcriptions is the precision rate. Such results for both speaker dependent (SD) and speaker adapted (SA) scenarios for the target corpora are shown in Table 5.1. In this table, we can see that precision and recall values of Mandarin are always much better than those of English, especially the recall values. For example, for course 1 with SA models, only 51% of English frames were recognized as belonging to English words while the other 49% were recognized as belonging to Chinese words. The system tends to take most signal segments as a part to a Chinese word, and as a result many English words are recognized as sequences of Chinese characters.

Table 5.1: *Frame-level Language Identification Achieved by the Baseline System.*



Acoustic Models	Course 1				Course 2				Course 3			
	Mandarin		*English*		*Mandarin*		*English*		*Mandarin*		*English*	
	Precision	Recall	Precision	Recall	Precision	Recall	Precision	Recall	Precision	Recall	Precision	Recall
(1) Speaker Adapted (SA)	0.91	0.99	0.87	0.51	0.92	0.96	0.66	0.57	0.93	0.99	0.81	0.56
(2) Speaker Dependent (SD)	0.94	0.99	0.88	0.73	0.95	0.98	0.79	0.74	0.95	0.99	0.84	0.76

5.2 Utilizing Frame-level Language Posterior Estimates in Decoding

Because many frames belonging to the guest language (English) were taken as belonging to the host language (Mandarin), the basic idea proposed here is that we wish to estimate a language posterior for each frame of signals, which can be used to boost the scores for those frames identified as possibly belonging to the guest language during decoding.

Assume the frame-level language posterior estimator generates for each frame of feature vector o_t a posterior probability of belonging to the guest language, $P(G|o_t)$ (and a posterior probability of belonging to the host language $P(H|o_t) = 1 - P(G|o_t)$), the acoustic model score for frame o_t with respect to all states q_j for guest language phoneme HMMs, $P(o_t|q_j)$, can then be boosted into a new score $\hat{P}(o_t|q_j)$ as below,

$$\hat{P}(o_t|q_j) = \begin{cases} P(o_t|q_j) \times [\frac{P(G|o_t)}{1-P(G|o_t)}]^{\alpha} & \text{if } P(G|o_t) > 0.5 \text{ and } q_j \in G \\ P(o_t|q_j) & \text{otherwise} \end{cases} \tag{5.1}$$

where $\hat{P}(o_t|q_j)$ is the score to be used in Viterbi decoding, G is the set of all HMM states for guest language phoneme models, and α is a weight parameter. In other words, if a frame o_t is identified as possibly belonging to the guest language, or $P(G|o_t) > 0.5$, its scores with all states of guest language phoneme models are boosted according to the posterior probability $P(G|o_t)$, otherwise the score is not changed. Because the decoder can choose the host language models very well, no action is needed if $P(G|o_t) < 0.5$. This approach can also be regarded as a multi-stream method [35–38] or in the category of a hybrid system.

The frame-level language posterior estimator producing $P(G|o_t)$ needed here can be implemented in different ways. For example, by neural networks with input features such as MFCCs, possibly based on longer context [36]. It is well known that language identification is easier for large signal segments and more difficult for short signal segments, such as the frame-level identification considered here. Although MFCCs have been useful for such tasks before, in this work, the bilingual speakers for the code-switched speech tend to pronounce guest (non-native) language words using host (native) language phonemes, and the MFCC features for the two languages are actually very similar. In addition, considering the two languages are pronounced by the same speaker, MFCCs may not necessarily be useful for the problem here [35]. MFCC features are extracted from a short time window, therefore contain only very limited language information. We therefore propose to use the blurred posteriorgram features (BPFs) extracted from decoded lattices as presented below.

5.3 Blurred Posteriorgram Features (BPFs)

As shown in Figure 5.1, each utterance is first decoded into a phoneme lattice with a first-pass recognition using the baseline system. With this phoneme lattice an N-dimensional

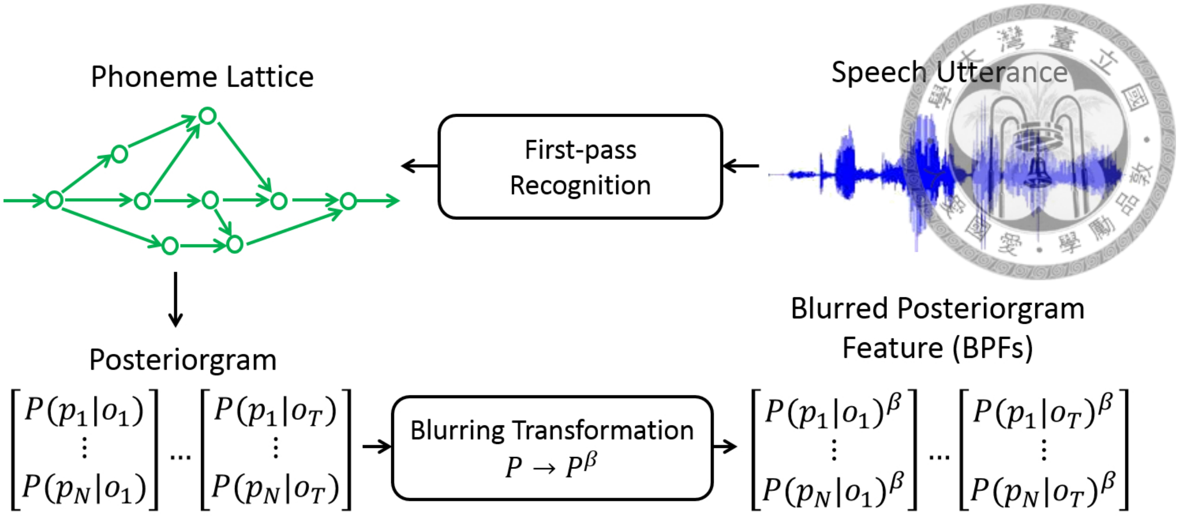

Phoneme Lattice
Speech Utterance
First-pass Recognition
Blurred Posteriorgram Feature (BPFs)
Posteriorgram
$\begin{bmatrix} P(p_1|o_1) \\ \vdots \\ P(p_N|o_1) \end{bmatrix} \cdots \begin{bmatrix} P(p_1|o_T) \\ \vdots \\ P(p_N|o_T) \end{bmatrix}$
Blurring Transformation
$P \rightarrow P^\beta$
$\begin{bmatrix} P(p_1|o_1)^\beta \\ \vdots \\ P(p_N|o_1)^\beta \end{bmatrix} \cdots \begin{bmatrix} P(p_1|o_T)^\beta \\ \vdots \\ P(p_N|o_T)^\beta \end{bmatrix}$

Figure 5.1: *Extraction of Blurred Posteriorgram Features (BPFs).*

posteriorgram vector $\vec{P}_t = \{P(p_i|o_t), i = 1, 2, ..., N\}$ can be obtained using forward-backward algorithm for each frame o_t, where p_i is a phoneme in either the host or guest languages, N is the total number of phonemes for the two languages involved, and $P(p_i|o_t) = 0$ for those phonemes p_i not appearing in the lattice at time t. The problem here is that very often guest language phonemes are decoded as host language phonemes, or $P(p_i|o_t)$ is usually relatively lower for guest language phonemes p_i even if o_t belongs to a guest language phoneme. So we wish to transform these posterior probabilities $P(p_i|o_t)$ into new posteriors $P'(p_i|o_t)$ in such a way that $P(p_i|o_t)$ is significantly increased if it is very small (so very possibly o_t belongs to the guest language), but the ordering for the posteriors should not be reversed in the new posteriors, i.e., $P'(p_i|o_t) > P'(p_j|o_t)$ if $P(p_i|o_t) > P(p_j|o_t)$. The latter requirement implies this transformation function from $P(p_i|o_t)$ to $P'(p_i|o_t)$ should be increasing monotonically.

There can be many ways to do this transformation, but an easy way to do it is in (5.2),

$$P'(p_i|o_t) = P(p_i|o_t)^{\beta}, 0 < \beta \leqslant 1, \tag{5.2}$$

where β is the ”blurring factor”, much smaller than 1 and close to 0. The concept of (5.2) is shown in Figure 5.2 for a few selected values of β. In Figure 5.2, we see that when $P(p_i|o_t)$ is small, $P'(p_i|o_t)$ is increased significantly (e.g. when $P(p_i|o_t) = 0.1$, $P'(p_i|o_t)$ is close to 0.8 for $\beta = 0.1$ and 0.9 for $\beta = 0.01$); for larger $P(p_i|o_t)$ it is also increased but by a smaller quantity (e.g. when $P(p_i|o_t) = 0.6$, $P'(p_i|o_t)$ is close to 0.95 for $\beta = 0.1$ and 0.97 for $\beta = 0.01$). So the ordering for the posteriors remains unchanged. This implies $P'(p_i|o_t)$ is monotonically increasing for increasing $P(p_i|o_t)$, while all posterior probabilities $P(p_i|o_t)$ are moved towards unity in a non-linear manner as in Figure 5.2. More importantly, such a function achieves the goal that $P(p_i|o_t)$ is significantly increased if it is small (or possibly o_t belongs to the guest language, so we can boost $P(p_i|o_t)$), while only slightly increased if it is large. For all frames, regardless of belonging to guest or host languages, the posterior probabilities are boosted greatly or slightly through the blurring transform, but the ordering is still preserved since the monotonically increasing nature. It is still unknown which language each frame belongs to, but the blurred posteriorgram tends to be better recognized by the neural network regarding the language it belongs. The value of β in (5.2) can be estimated by a development data set.

In other words, the blurring transform defined above is to properly enhance the posterior probability distribution $P(p_i|o_t)$ which is usually highly biased towards the host language phonemes, while preserve the ordering among all posterior probabilities by a monotonically increasing mapping function. Certainly it is possible to design other mapping functions achieving similar goals, while the one in (5.2) is simply an easy example.

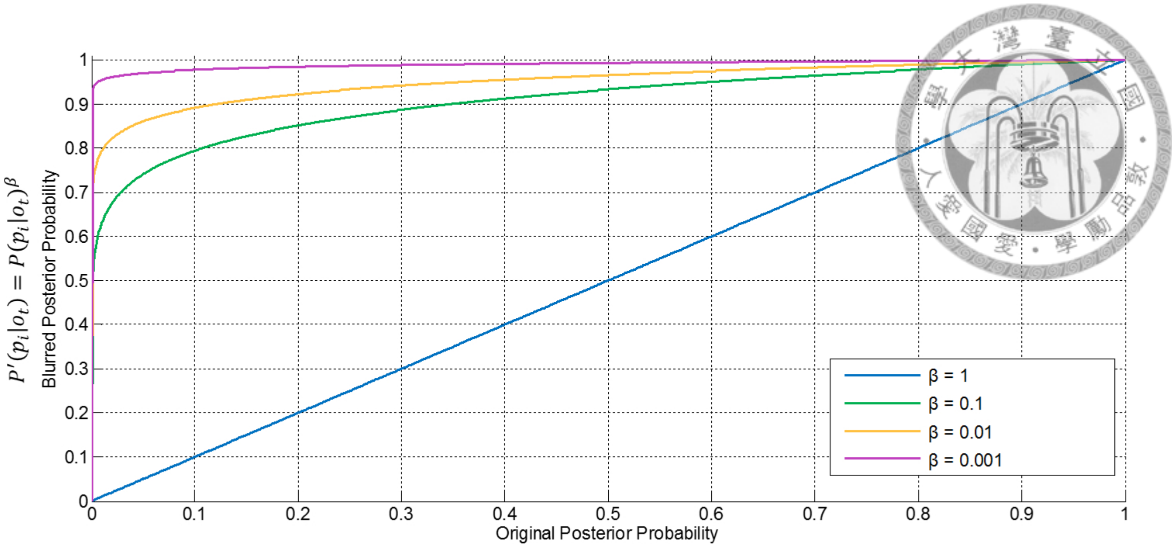
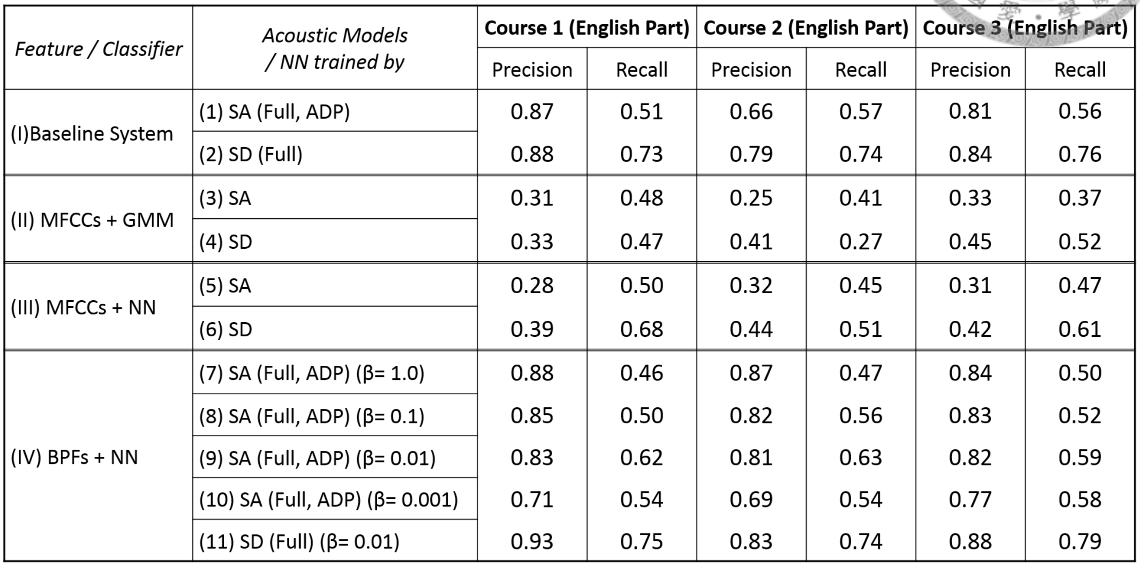


Figure 5.2: *The Blurring Transformation for Posterior Probabilities.*

These enhanced posterior probabilities are referred to as blurred posteriorgram features (BPFs), and used as the input to a neural network for generating an estimate for the language posterior $P(G|o_t)$ and $P(H|o_t)$ to be used in (5.1) with two training targets: guest or host language. Note that the blurred posteriorgram features (BPFs) are generated from lattices from the first-pass recognition, so it contains not only acoustic information such as those in MFCCs, but also information from acoustic models, language model and lexicon. It is also a frame-level feature but extracted based on the signals in the whole utterance. Furthermore, because the posteriorgram $P(p_i|o_t)$ acquired directly from the lattice without blurring are strongly biased towards the host language by the first-pass recognition system, therefore the blurring transform is applied here to properly take care of the bias. These are why we believe the proposed BPFs carries stronger information for estimating the language posteriors $P(G|o_t)$ and $P(H|o_t)$ when compared with conventional features such as MFCC, as will be verified by the experimental results to be reported below.

Table 5.2: *Frame-level Language Identification Results with Different Input Features and Different Classifiers.*



Feature / Classifier	*Acoustic Models / NN trained by*	Course 1 (English Part)		Course 2 (English Part)		Course 3 (English Part)	
		Precision	Recall	Precision	Recall	Precision	Recall
(I)Baseline System	(1) SA (Full, ADP)	0.87	0.51	0.66	0.57	0.81	0.56
	(2) SD (Full)	0.88	0.73	0.79	0.74	0.84	0.76
(II) MFCCs + GMM	(3) SA	0.31	0.48	0.25	0.41	0.33	0.37
	(4) SD	0.33	0.47	0.41	0.27	0.45	0.52
(III) MFCCs + NN	(5) SA	0.28	0.50	0.32	0.45	0.31	0.47
	(6) SD	0.39	0.68	0.44	0.51	0.42	0.61
(IV) BPFs + NN	(7) SA (Full, ADP) (β= 1.0)	0.88	0.46	0.87	0.47	0.84	0.50
	(8) SA (Full, ADP) (β= 0.1)	0.85	0.50	0.82	0.56	0.83	0.52
	(9) SA (Full, ADP) (β= 0.01)	0.83	0.62	0.81	0.63	0.82	0.59
	(10) SA (Full, ADP) (β= 0.001)	0.71	0.54	0.69	0.54	0.77	0.58
	(11) SD (Full) (β= 0.01)	0.93	0.75	0.83	0.74	0.88	0.79

5.4 Frame-level Language Identification Analysis

In Section 5.1, we analyze the frame-level language identification recall/precision for the baseline recognition system. In addition we propose to estimate the language posteriors using BPFs with a neural network. Because MFCCs were used for this purpose previously [32,36], several of such approaches are also compared here. We first used MFCC features with a GMM classifier as was done previously [32], and then we replaced the GMM classifier by a neural network classifier. Finally the MFCC features are replaced by BPFs as the input to the neural network classifier as proposed here. The neural network classifier used in the experiment had one hidden layer with 1024 hidden nodes.

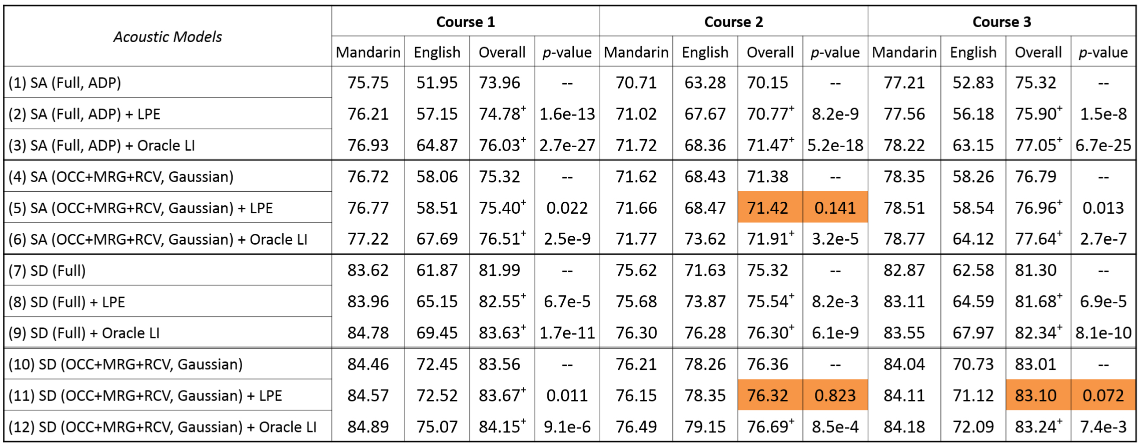
The frame-level precision/recall rates are listed in Table 5.2 for English part only. Part (I) (rows 1, 2) are for the baseline system, directly copied from Table 5.1. Part

(II) (rows 3,4) are then the results of frame-level language identification obtained using 39-dimensional MFCCs with a GMM classifier [32], and part (III) (rows 5,6) are those obtained when a neural network classifier was used to replace the GMM classifier and MFCCs with longer context [36] (4 preceding and 4 following frames) were used as the frame-level input. Part (IV) (rows 7-11) are then the results using BPFs proposed here as the input features for the neural network. In all parts (II)(III)(IV) the GMM or the neural network classifiers trained with the adaptation/training sets listed in Table 2.1 of Chapter 2 are respectively referred to as SA/SD, and in part (IV) SA/SD further indicate the BPFs were obtained with phoneme lattices produced by the SA/SD baseline systems as in rows 1, 2 of part (I).

We can see from parts (II) and (III) from Table 5.2 that it is difficult to identify the language using MFCCs at least for this task, regardless of the input context length or the type of the classifiers. Although the recalls obtained were close to the baseline system (rows 3-6 vs. rows 1,2), the precisions were very low. It is also clear that the neural network outperformed the GMM classifier in more cases for this task (rows 5,6 vs. rows 3,4). However, even with a longer input context and a stronger classifier such as the neural network, the performance using MFCCs is still not good enough. Clearly, it is not easy to identify the language simply based on several frame of MFCCs, especially for this task in which both languages were produced by the same speaker in the same utterance. Part (IV) (rows 7-11) then includes results of using BPFs proposed here with different values of β used in (9), from which we selected $\beta = 0.01$ for the following experiments.

Note that as shown in Figure 5.2 and discussed in Section 5.3, different values of β actually gave very different nonlinear transformations, or the posterior probabilities

Table 5.3: *Results for Using the Language Posterior Estimate (LPE) or Oracle Language Identification (Oracle LI) in Decoding for Systems with Cross-language Acoustic Modeling (OCC+MRG+RCV, Gaussian) (Accuracies) (%).*



Acoustic Models	Course 1				Course 2				Course 3			
	Mandarin	English	Overall	*p*-value	Mandarin	English	Overall	*p*-value	Mandarin	English	Overall	*p*-value
(1) SA (Full, ADP)	75.75	51.95	73.96	--	70.71	63.28	70.15	--	77.21	52.83	75.32	--
(2) SA (Full, ADP) + LPE	76.21	57.15	74.78^{+}	1.6e-13	71.02	67.67	70.77^{+}	8.2e-9	77.56	56.18	75.90^{+}	1.5e-8
(3) SA (Full, ADP) + Oracle LI	76.93	64.87	76.03^{+}	2.7e-27	71.72	68.36	71.47^{+}	5.2e-18	78.22	63.15	77.05^{+}	6.7e-25
(4) SA (OCC+MRG+RCV, Gaussian)	76.72	58.06	75.32	--	71.62	68.43	71.38	--	78.35	58.26	76.79	--
(5) SA (OCC+MRG+RCV, Gaussian) + LPE	76.77	58.51	75.40^{+}	0.022	71.66	68.47	71.42	0.141	78.51	58.54	76.96^{+}	0.013
(6) SA (OCC+MRG+RCV, Gaussian) + Oracle LI	77.22	67.69	76.51^{+}	2.5e-9	71.77	73.62	71.91^{+}	3.2e-5	78.77	64.12	77.64^{+}	2.7e-7
(7) SD (Full)	83.62	61.87	81.99	--	75.62	71.63	75.32	--	82.87	62.58	81.30	--
(8) SD (Full) + LPE	83.96	65.15	82.55^{+}	6.7e-5	75.68	73.87	75.54^{+}	8.2e-3	83.11	64.59	81.68^{+}	6.9e-5
(9) SD (Full) + Oracle LI	84.78	69.45	83.63^{+}	1.7e-11	76.30	76.28	76.30^{+}	6.1e-9	83.55	67.97	82.34^{+}	8.1e-10
(10) SD (OCC+MRG+RCV, Gaussian)	84.46	72.45	83.56	--	76.21	78.26	76.36	--	84.04	70.73	83.01	--
(11) SD (OCC+MRG+RCV, Gaussian) + LPE	84.57	72.52	83.67^{+}	0.011	76.15	78.35	76.32	0.823	84.11	71.12	83.10	0.072
(12) SD (OCC+MRG+RCV, Gaussian) + Oracle LI	84.89	75.07	84.15^{+}	9.1e-6	76.49	79.15	76.69^{+}	8.5e-4	84.18	72.09	83.24^{+}	7.4e-3

$P(p_i|o_t)$ were boosted in quite different ways. The value of β here was tuned using the development set of course 1, but this value was also applied to the other two courses as well. As a result, the value β can be data-dependent, but it did not vary much. Here we see with properly chosen value of β, improved recalls were achievable with precision either improved or slightly degraded (rows 7-11 vs. rows 1,2).

5.5 Experimental Results

Here we tested the proposed language posterior estimates with blurred posteriorgram features (BPFs) along with the HMM-based acoustic models obtained with the approaches described in Chapter 3, including acoustic unit merging and recovery on Gaussian level with occupancy ranking for unit classification. The results are listed in Table 5.3.

In Table 5.3, row 1 is for the SA baseline. Row 2 is the same except with the proposed language posterior estimate with BPFs used in decoding in addition (labelled as ”LPE”, language posterior estimation). The boosting factor $\alpha = 1.0$ in (5.1) was set empirically and applied throughout the experiments. We see the English accuracy was improved significantly, while the Mandarin accuracy was improved too (row 2 vs. 1). As a result, the overall performance was improved significantly with good p-values. Row 3 is the same as row 2 except assuming oracle guest language identification obtained with forced alignment with the reference transcriptions, serving as the upper bound (labelled as ”Oracle LI”, oracle language identification). We can see that there is much room for further improvement. Rows 4, 5, 6 are exactly the same as rows 1, 2, 3, except the acoustic models used here were obtained with the best approaches proposed here, which is unit merging and recovery with unit occupancy ranking on the Gaussian level. We can see the same trends as in rows 1, 2, 3, except the p-value for course 2 in row 5 exceeded 0.05. This shows the approaches proposed here are equally useful for different acoustic models, and the improvements are additive and complementary to each other. Though the improvements brought by jointly using the two different sets of approaches (cross-lingual acoustic modeling and language posterior estimate) is relatively limited compared to either individual one in this task, this improvement should depend on the performance of the individual methods for the task considered. For example, if very good language identification can be accomplished, as the ”Oracle LI” results showed, jointly using the two sets of approaches should be able to offer significant improvements hardly achievable by improving acoustic models alone.

Rows 7-12 are similar results as rows 1-6, except with the speaker dependent (SD) models, offering exactly the same observations. So the proposed approaches are useful for increased training data as well, not limited to data insufficiency scenarios. We also noticed that the improvement is more significant for weak acoustic models, which is reasonable.

Table 5.3 also serves as a brief summary of the results obtained using the proposed HMM-based cross-lingual acoustic modeling and frame-level language posterior estimate, when rows 3, 6, 9, 12 for oracle language identification are ignored, with rows 1, 2, 4, 5 for SA and rows 7, 8, 10, 11 for SD. Rows 1, 7 are baselines, rows 4, 10 for the best set of acoustic models (unit merging and recovery with occupancy ranking on Gaussian level) alone, rows 2, 8 for language posterior estimate alone, and rows 5, 11 for using the best acoustic models with language posterior estimation simultaneously.

The best acoustic models in rows 4 and 10 gave 11.76%, 8.03%, 10.28% relative improvements for English part for courses 1, 2, 3 respectively for SA case and 17.10%, 9.26%, 13.02% for SD case. More improvements were obtained in SD case because the larger data size gave more precise unit merging relationships. Rows 2 and 8 with the proposed language posterior estimates alone gave 10.00%, 6.94% and 6.34% relative improvements for SA case and 5.30%, 3.13% and 3.21% for SD case. The improvements for SD case were slightly less, probably because the SD models were already capable of identifying the languages better. The system using language posterior estimates in decoding with the best acoustic models in rows 5, 11 gave 12.63%, 8.20%, 10.81% relative improvement for SA case and 17.21%, 9.38%, 13.65% for SD case.

Chapter 6 DNN-based Cross-lingual Acoustic Modeling and Experimental Results

6.1 DNN-based Acoustic Modeling

As a classifier, deep neural network (DNN) has been proven to significantly outperform the conventional neural network with less hidden layers [25,26,30,46,47]. By pre-training with restricted Boltzmann machine (RBM), more hidden layers or deeper network structure can be trained sequentially. In acoustic modeling, context-dependent DNN-HMM (CD-DNN-HMM) is the most popular form of DNN application. Different from conventional HMM-GMM structure, each context-dependent HMM state is modeled by a node of the output layer of the DNN instead of a GMM [46]. During the recognition process, the acoustic features are fed to the input layer of the DNN as observation sequences, and the likelihood between an observed feature and a HMM state is given by the output value of the node modeling the HMM state divided by the correspdoning HMM state prior, as shown in Figure 6.1. Generally, the acoustic features accepted at the input layer were concatenated in consecutive frames with various choices, such as spectrogram, mel-filter bank outputs and MFCCs. During the training process of DNN, each hidden layer was pre-trained by a restricted Boltzmann machine (RBM) in an unsupervised manner with input being the output of the previous layer. Compared with the conventional HMM-GMM systems, DNN systems give better performance with higher computaional requirements in both training and decoding process.

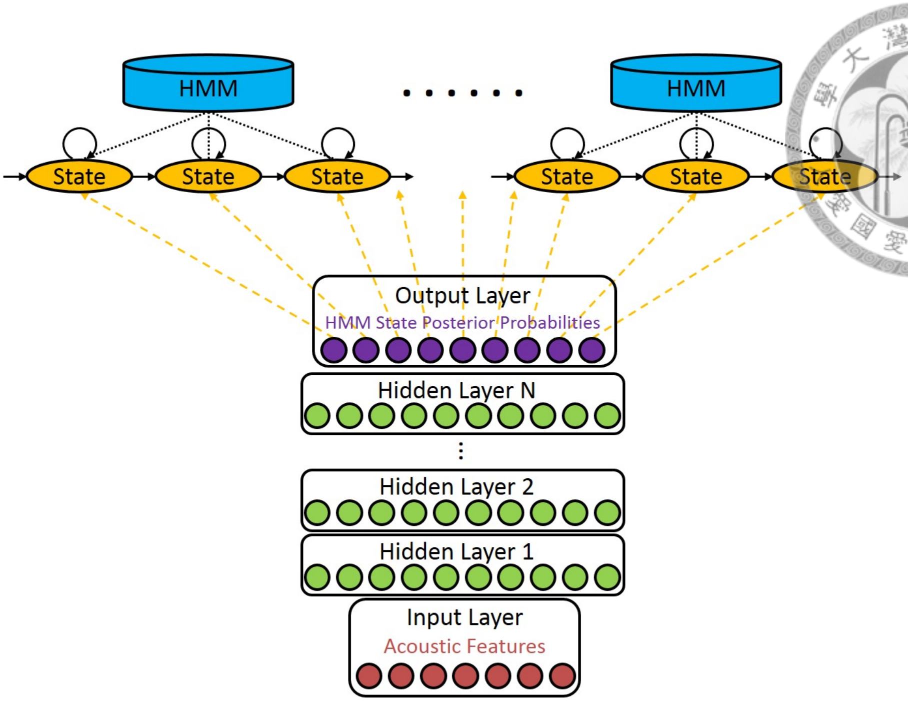


Figure 6.1: *Context-dependent Deep Neural Network Hidden Markov Model (CD-DNN-HMM) for Acoustic Modeling.*

6.2 Code-switched CD-DNN-HMM

Similar to the case of monolingual speech tasks, CD-DNN-HMM can be used in code-switched bilingual speech systems as well. For code-switched bilingual speech, the system should be able to handle both languages simultaneously. As a results, in code-switched CD-DNN-HMM, the output layer consists of HMM states for all Mandarin and English triphones. In these triphones, the central phonemes include all Mandarin phonemes plus English phonemes, and all cross-language context dependency conditions are considered. Modeling HMM states for different languages by individual networks may lead to poor results due to the very limited data for the guest language in the code-switched corpora. We therefore adopt the concept of multilingual DNN recently proposed [30], in which all layers except the output layer were jointly trained by all data of

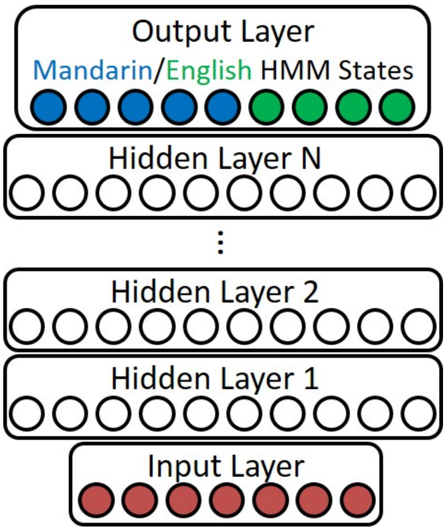


Figure 6.2: *Context-dependent Deep Neural Network for Acoustic Modeling in Code-switched Speech Recognition.*

both languages. The only difference is that here the output layer including HMM states for the two languages with all code-switching context dependencies are jointly obtained, while in the multilingual DNN [30] the HMM states for each individual language were separately obtained.

6.3 Code-switched BF-HMM/GMM

In addition to CD-DNN-HMM, in which the nodes in the output layer of DNN are designed for modeling HMM states, another popular form of neural network application is using the network as a feature extractor [61]. This approach originated from the auto-encoding theory of neural networks. By transforming features sequentially from one layer to another, better features could be obtained. Conventionally, the output of the last hidden layer is extracted, with a size much smaller than other hidden layers for generating more compact features. However, it is difficult to decide the size of the last hidden layer, and it

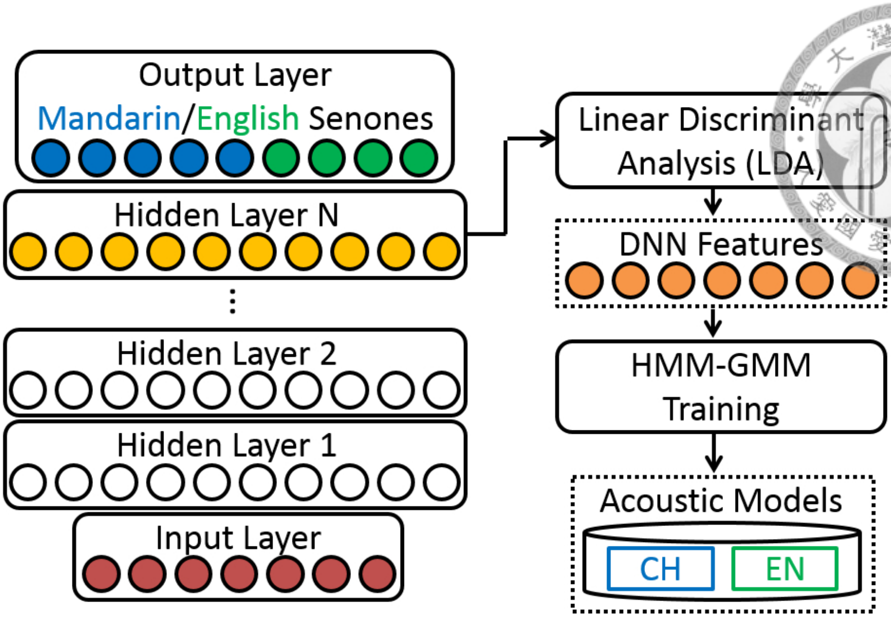


Figure 6.3: *Bottleneck Feature for Code-switched BF-HMM/GMM.*

takes time to retrain the DNN whenever the size of bottleneck feature is changed. Therefore, fixing the size of the last hidden layer but using a dimension reduction procedure was proposed to extract the DNN-based bottleneck features [48].

The performance of such a system is reported to be comparable to CD-DNN-HMM with the same data, and such a structure is completely compatible to the conventional acoustic modeling framework including many powerful techniques such as MLLR, MAP, MPE and MMI, as well as many approaches for cross-lingual acoustic modeling such as unit merging and recovery [1] as presented in Chapter 3. In this work, we use linear discriminant analysis (LDA) for the dimension reduction mentioned above as shown in the upper part of Figure 6.3 to reduce the dimensionality from the size of last hidden layer (typically thousands) to the size for feature vector dimensionality for HMM/GMM (typically tens). These bottleneck features are then used to train the acoustic models for the two languages.

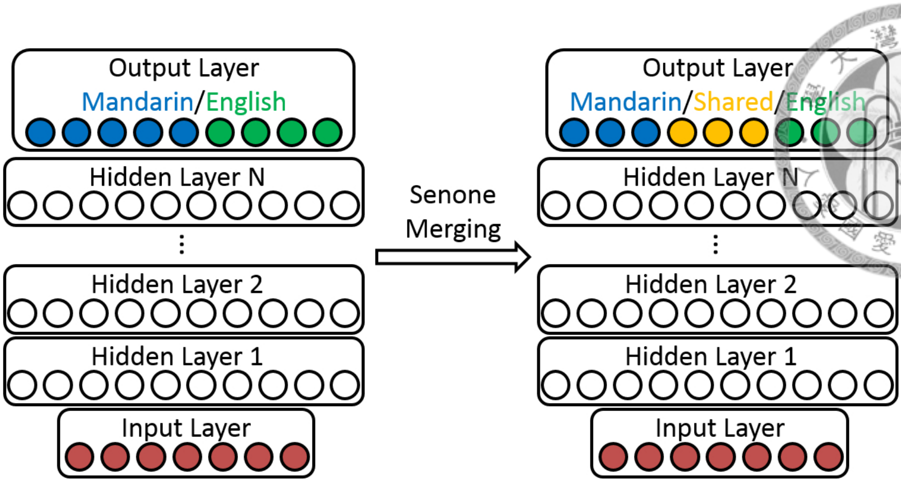


Figure 6.4: *Unit Merging on State Level for CD-DNN-HMM.*

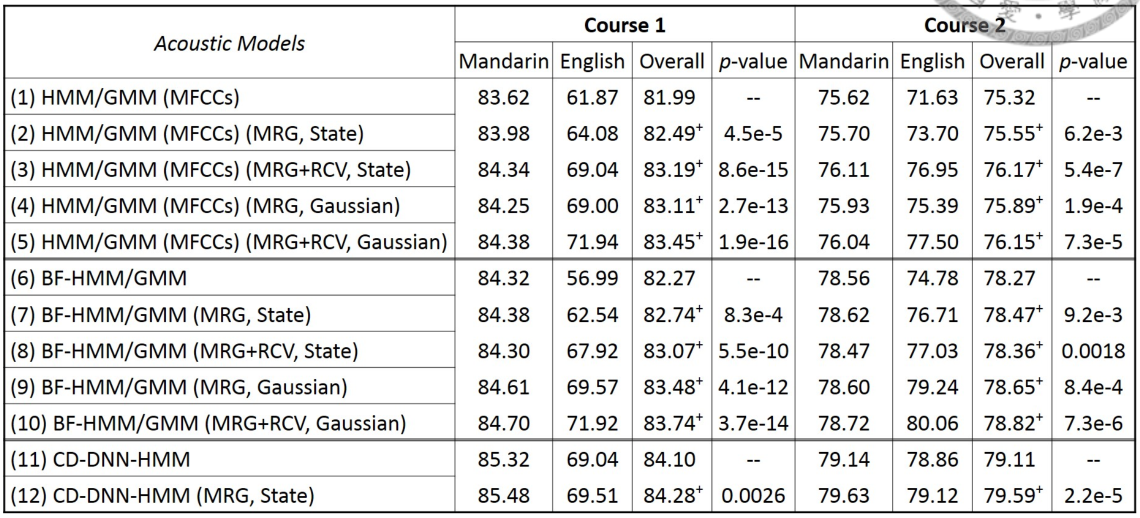
6.4 Unit Merging on State Level for CD-DNN-HMM

Similar to the HMM-based cross-lingual acoustic modeling approaches (Chapter 3), acoustic unit merging is certainly feasible for CD-DNN-HMM systems. Unit merging for CD-DNN-HMM can be achieved on HMM state level by replacing the HMM states in the output layer by the corresponding merged set of HMM statess (some nodes are shared across languages) before DNN training as in Figure 6.4. Note that in conventional HMM/GMM, each unit (HMM state or Gaussian) is to model the local distribution for the specific unit. In contrast, for CD-DNN-HMM here the parameters are shared by all target HMM states and trained by all training data. Therefore, the data sparseness and imbalance problem may not be as serious here.

6.5 Experimental Results

The experimental results regarding deep neural network approaches for both courses 1 and 2 are listed in Table 6.1. Rows 1-5 are for HMM/GMM with conventional MFCCs, directly copied from results Table 4.1 and Table 4.4, with row 1 for the conventional

Table 6.1: *Experimental Results for HMM/GMM systems with Different Features and CD-DNN-HMM systems*



Acoustic Models	Course 1				Course 2			
	Mandarin	English	Overall	*p*-value	Mandarin	English	Overall	*p*-value
(1) HMM/GMM (MFCCs)	83.62	61.87	81.99	--	75.62	71.63	75.32	--
(2) HMM/GMM (MFCCs) (MRG, State)	83.98	64.08	82.49$^+$	4.5e-5	75.70	73.70	75.55$^+$	6.2e-3
(3) HMM/GMM (MFCCs) (MRG+RCV, State)	84.34	69.04	83.19$^+$	8.6e-15	76.11	76.95	76.17$^+$	5.4e-7
(4) HMM/GMM (MFCCs) (MRG, Gaussian)	84.25	69.00	83.11$^+$	2.7e-13	75.93	75.39	75.89$^+$	1.9e-4
(5) HMM/GMM (MFCCs) (MRG+RCV, Gaussian)	84.38	71.94	83.45$^+$	1.9e-16	76.04	77.50	76.15$^+$	7.3e-5
(6) BF-HMM/GMM	84.32	56.99	82.27	--	78.56	74.78	78.27	--
(7) BF-HMM/GMM (MRG, State)	84.38	62.54	82.74$^+$	8.3e-4	78.62	76.71	78.47$^+$	9.2e-3
(8) BF-HMM/GMM (MRG+RCV, State)	84.30	67.92	83.07$^+$	5.5e-10	78.47	77.03	78.36$^+$	0.0018
(9) BF-HMM/GMM (MRG, Gaussian)	84.61	69.57	83.48$^+$	4.1e-12	78.60	79.24	78.65$^+$	8.4e-4
(10) BF-HMM/GMM (MRG+RCV, Gaussian)	84.70	71.92	83.74$^+$	3.7e-14	78.72	80.06	78.82$^+$	7.3e-6
(11) CD-DNN-HMM	85.32	69.04	84.10	--	79.14	78.86	79.11	--
(12) CD-DNN-HMM (MRG, State)	85.48	69.51	84.28$^+$	0.0026	79.63	79.12	79.59$^+$	2.2e-5

HMM/GMM baseline, row 2 for merging on HMM state level and row 3 with unit recovery in addition, similarly for rows 4,5 except on Gaussian level. Rows 6-10 are in similar setup for BF-HMM/GMM except MFCCs as training features were replaced by DNN bottleneck features. Also, the bottleneck features were extracted from MFCCs in concatenated frames, therefore may carry additional context information. Rows 11,12 are for CD-DNN-HMM with row 12 for HMM state merging in the output layer of the deep neural network.

First consider rows 1, 6, 11 without unit merging, we can see that regardless of the model structure and the features used, the English accuracies were always significantly lower than Mandarin due to the data imbalance problem for the code-switching bilingual speech. By comparing rows 1 and 11, we see the CD-DNN-HMM greatly outperformed the HMM/GMM baseline using the same MFCC features without any unit merging or recovery. The BF-HMM/GMM with DNN bottleneck features in row 6 was somewhere in between in most cases.

Now compare rows 2, 7, 12 to rows 1, 6, 11. First we see the English accuracy was significantly improved (rows 2, 7 vs. 1, 6) by unit merging on HMM state level due to data sharing regardless of using MFCCs or DNN bottleneck features. However, for CD-DNN-HMM, the improvement brought by unit merging on HMM state level is relatively limited (rows 12 vs. 11). A possible explanation for this is that for DNN all parameters are shared by all target classes, so the data sparseness issue is not as serious as in HMM/GMM (rows 1,2,6,7), for which the parameters are to model the local distributions for the specific HMM states. Furthermore, by checking rows 3,8 in addition, we see that performing an extra pass of unit recovery did bring improvement for HMM/GMM (rows 3, 8 vs. 2, 7) for either MFCCs or DNN bottleneck features. For Gaussian level merging and recovery (rows 4, 5, 9, 10), we can see the trend is very similar to that for senone level merging and recovery, except with better performance due to the finer structure of the Gaussian level (rows 4, 5, 9, 10 vs. 2, 3, 7, 8). Furthermore, the DNN bottleneck features always outperformed MFCCs in most cases (rows 7-10 vs. 2-5.)

Comparing BF-HMM/GMM with CD-DNN-HMM, we see the best BF-HMM/GMM (merging and recovery on Gaussian level in row 10) achieved better English accuracy while the best CD-DNN-HMM (merging on senone level in row 12) achieved better Mandarin and overall accuracies (rows 10 vs. 12). This is important since Gaussian level merging is feasible only for BF-HMM/GMM and English accuracy is emphasized in this task. Therefore, for the two approaches of using DNN in HMM state modeling (CD-DNN-HMM) or for feature extraction (BF-HMM/GMM) considered in the work for the

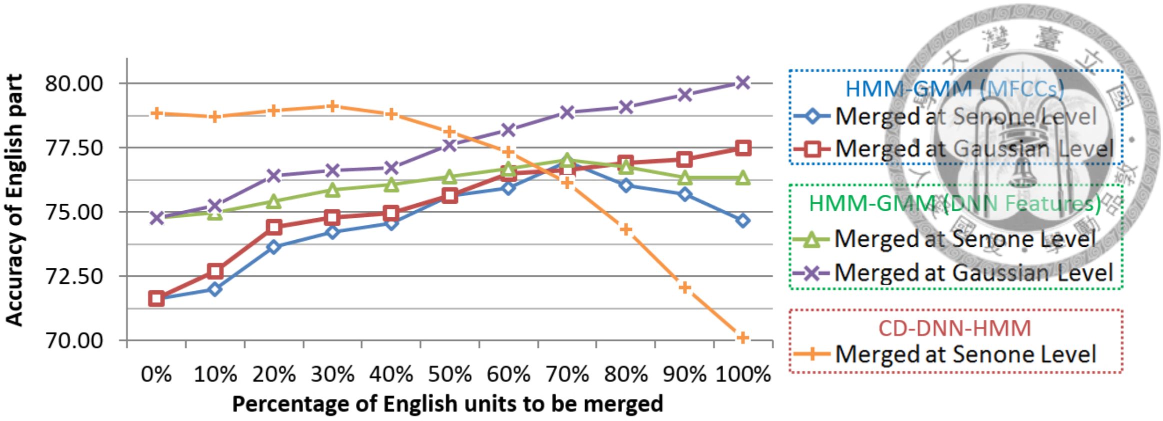


Figure 6.5: *English Accuracy with Different Merging Percentage of Course 1 for Different Systems.*

specific task, DNN in modeling gave better overall performance while DNN for feature extraction gave better English accuracy.

For HMM/GMM system with MFCCs as training features, while performing unit merging, we can choose to merge only a selected percentage of English units with the corresponding Mandarin units starting with those pairs with minimum distances, but not all. The English accuracies obtained in this way for different percentages of English units merged (but not recovered) on HMM state and Gaussian levels for HMM/GMM (MFCCs) baseline, BF-HMM/GMM and CD-DNN-HMM for course 1 is shown in Figure 6.5. The best values on each curve correspond to the numbers in rows 2,4,7,9,12 in Table 6.1, respectively. We can see that the English accuracy in general increased when more English units were merged on either HMM state or Gaussian levels for HMM/GMM regardless of using MFCCs or DNN bottleneck features, although in some cases too high percentage of merging may not be good. However, for CD-DNN-HMM, the improvement achievable with unit merging on HMM state level was very limited (best at 30 %), and the performance degraded seriously when too many HMM states were merged. This is consistent with the previous explanation that parameters in DNN are jointly trained by all data, so do not benefit too much from data-sharing, and in fact the modeling ability of DNN was degraded when too few HMM states are present in the output layer.

Chapter 7 Conclusion

Recognition of speech with code-switching occurring frequently within utterances is an important problem for the globalized world today. The difficulties include not only the lack of the guest language data and the language identification to be performed over very short segments of speech, but the fact that the English (guest language) is usually spoken by a non-native speaker within an utterance of his native language (host language), so very often taken as in the host language. In this thesis, we present an integrated framework for recognizing such highly imbalanced bilingual code-switched utterances on top of the previously proposed unit merging approaches on three levels: model, state and Gaussian. This includes unit recovery after being merged, unit occupancy ranking for much more flexible data sharing both inter-language and intra-language, and frame-level language posterior estimates to be used in decoding. In addition, we proposed to utilize the deep neural networks (DNN), including CD-DNN-HMM and BF-HMM/GMM, with the unit merging and recovery approaches. We also present a complete set of experimental results comparing all approaches involved for a real-world application scenario under unified conditions. The experimental results verified that the concepts behind the proposed framework are all useful and can offer improved recognition accuracy, i.e., the acoustic units should be properly recovered after being merged, the data sharing should be much more flexible than simply using state-tied triphones and across the languages, and good estimates of frame-level language posteriors can help in decoding.

Reference

[1] Ching-Feng Yeh and Lin-Shan Lee, “An improved framework for recognizing highly imbalanced bilingual code-switched lectures with cross-language acoustic modeling and frame-level language identification,” *IEEE Transactions on Audio, Speech, and Language Processing*, 2015.

[2] Tanja Schultz and Alex Waibel, “Multilingual and crosslingual speech recognition,” in *DARPA Workshop on Broadcast News Transcription and Understanding*, 1998.

[3] Tanja Schultz and Alex Waibel, “Language independent and language adaptive acoustic modeling for speech recognition,” *Speech Communication*, 2001.

[4] Hui Lin, Li Deng, Jasha Droppo, Dong Yu, and Alex Acero, “Learning methods in multilingual speech recognition,” in *NIPS*, 2008.

[5] L. Lamel, M. Adda-decker, and J.L. Gauvain, “Issues in large vocabulary, multilingual speech recognition,” in *Europ. Conf. on Speech Communication and Technology*, 1995, pp. 185–188.

[6] Li Deng, “Integrated-multilingual speech recognition using universal phonological features in a functional speech production model,” in *ICASSP*, 1997.

[7] Alex Waibel, Hagen Soltau, Tanja Schultz, Thomas Schaaf, and Florian Metze, *Speech-to-speech Translation.*

[8] S.J. Young, M. Adda-Dekker, X. Aubert, C. Dugast, J.L. Gauvain, D.J. Kershaw, L. Lamel, D.A. Leeuwen, D. Pyea, A.J. Robinson, H.J.M. Steeneken, and P.C.

Woodland, “Multilingual large vocabulary speech recognition: the european sqale project,” in *Computer Speech & Language*, 1997.

[9] Tanja Schultz, Ngoc Thang Vu, and Tim Schlippe, “Globalphone: A multilingual text & speech database in 20 languages,” in *ICASSP*, 2013.

[10] Zoltán Tüske, David Nolden, Ralf Schlüter, and Hermann Ney, “Multilingual mrasta features for low-resource keyword search and speech recognition systems,” in *ICASSP*, 2014.

[11] Jie Li, Rong Zheng, and Bo Xu, “Investigation of cross-lingual bottleneck features in hybrid asr systems,” in *Interspeech*, 2014.

[12] B. Mark and E. Barnard, “Phone clustering using bhattacharyya distance,” in *ICSLP*, 1996.

[13] Yanmin Qian and Jia Liu, “Phone modeling and combining discriminative training for mandarin-english bilingual speech recognition,” in *ICASSP*, 2010.

[14] Anne-Katrin Kienappel, Dieter Geller, and Rolf Bippus, “Cross-language transfer of multilingual phoneme models,” in *Automatic Speech Recognition*, 2000.

[15] J. Kohler, “Language adaptation of multilingual phone models for vocabulary independent speech recognition tasks,” in *Acoustics, Speech and Signal Processing*, 1998.

[16] Houwei Cao, Tan Lee, and P.C. Ching, “Cross-lingual speaker adaptation via gaussian component mapping,” in *Interspeech*, 2010.

[17] Ching-Feng Yeh, Chao-Yu Huang, and Lin-Shan Lee, "Bilingual acoustic model adaptation by unit merging on different levels and cross-level integration," in *Interspeech*, 2011.

[18] R. Bayeh, S. Lin, G. Chollet, and C. Mokbel, "Towards multilingual speech recognition using data driven source/target acoustical units association," in *ICASSP*, 2004.

[19] Edward Lebese, Jonas Manamela, and Nalson Gasela, "Towards a multilingual recognition system based on phone-clustering scheme for decoding local languages," in *SATNAC*, 2012.

[20] Lukas Burget, Petr Schwarz, Mohit Agarwal, Pinar Akyazi, Kai Feng, Arnab Ghoshal, Ondrej Glembek, Nagendra Goel, Martin Karafiat, Daniel Povey, Ariya Rastrow, Richard C. Rose, and Samuel Thomas, "Multilingual acoustic modeling for speech recognition based on subspace gaussian mixture models," in *ICASSP*, 2010.

[21] Yanmin Qian, Daniel Povey, and Jia Lu, "State-level data borrowing for low-resource speech recognition based on subspace gmms," in *Interspeech*, 2011.

[22] Daniel Povey, *Discriminative training for large vocabulary speech recognition*, Ph.D. thesis, Cambridge University Engineering Dept, 2003.

[23] Ran Xu, Qingqing Zhang, Jielin Pan, and Yonghong Yan, "Investigations to minimum phone error training in bilingual speech recognition," in *FSKD*, 2009.

[24] Ching-Feng Yeh, Yiu-Chang Lin, and Lin-Shan Lee, "Minimum phone error model training on merged acoustic units for transcribing bilingual code-switched speech," in *ISCSLP*, 2012.

[25] Li Deng, Jinyu Li, Jui-Ting Huang, Kaisheng Yao, Dong Yu, Frank Seide, Michael L. Seltzer, Geoff Zweig, Xiaodong He, Jason Williams, Yifan Gong, , and Alex Acero, "Recent advances in deep learning for speech research at microsoft," in *ICASSP*, 2013.

[26] Ngoc Thang Vu, David Imseng, Daniel Povey, Petr Motlicek, Tanja Schultz, and Herve Bourlard, "Multilingual deep neural network based acoustic modeling for rapid language adaptation," in *ICASSP*, 2014.

[27] Livescu K., Fosler-Lussier E., and F. Metze, "Subword modeling for automatic speech recognition: past, present, and emerging approaches," *IEEE Signal Processing Magazine*, 2012.

[28] Chung-Hsien Wu, Han-Ping Shen, , and Yan-Ting Yang, "Phone set construction based on context-sensitive articulatory attributes for code-switching speech recognition," in *ICASSP*, 2012.

[29] Raul Fernandez, Jia Cui, Andrew Rosenberg, Bhuvana Ramabhadran, and Xiaodong Cui, "Exploiting vocal-source features to improve asr accuracy for low-resource languages," in *Interspeech*, 2014.

[30] Jui-Ting Huang, Jinyu Li, Dong Yu, Li Deng, and Yifan Gong, "Cross-language knowledge transfer using multilingual deep neural network with shared hidden layers," in *ICASSP*, 2013.

[31] Ching-Feng Yeh, Chao-Yu Huang, Liang-Che Sun, , and LinShan Lee, "An integrated framework for transcribing mandarin-english code-mixed lectures with improved acoustic and language modeling," in *ISCSLP*, 2010.

[32] Ngoc Thang Vu, Dau-Cheng Lyu, Jochen Weiner, Dominic Telaar, Tim Schlippe, Fabian Blaicher, Eng-Siong Chng, Tanja Schultz, and Haizhou Li, "A first speech recognition system for mandarin-english code-switch conversational speech," in *ICASSP*, 2012.

[33] Ching-Feng Yeh, Liang-Che Sun, Chao-Yu Huang, and Lin-Shan Lee, "Bilingual acoustic modeling with state mapping and three-stage adaptation for transcribing unbalanced code-mixed lectures," in *ICASSP*, 2011.

[34] Ching-Feng Yeh and Lin-Shan Lee, "Transcribing code-switched bilingual lectures using deep neural networks with unit merging in acoustic modeling," in *ICASSP*, 2014.

[35] Ching-Feng Yeh, Aaron Heidel, Hong-Yi Lee, and Lin-Shan Lee, "Recognition of highly imbalanced code-mixed bilingual speech with frame-level language detection based on blurred posteriorgram," in *ICASSP*, 2012.

[36] David Imseng, Herve Bourlard, Mathew Magimai.-Doss, and John Dines, "Language dependent universal phoneme posterior estimation for mixed language speech recognition," in *ICASSP*, 2011.

[37] Jochen Weiner, Ngoc Thang Vu, Dominic Telaar, Florian Metze, Tanja Schultz, Dau-Cheng Lyu, Eng-Siong Chng, and Haizhou Li, "Integration of language identifica-

tion into a recognition system for spoken conversations containing code-switches," in *SLTU*, 2012.

[38] N. Barroso, Karmele López de Ipiña, Aitzol Ezeiza, O. Barroso, and U. Susperregi, "Hybrid approach for language identification oriented to multilingual speech recognition in the basque context," in *Hybrid Artificial Intelligence Systems Lecture Notes in Computer Science*, 2010.

[39] Dau-Cheng Lyu and Ren-Yuan Lyu, "Language identification on code-switching utterances using multiple cues," in *Interspeech*, 2008.

[40] Yu Zhang, Ekapol Chuangsuwanich, and James R. Glass, "Language id-based training of multilingual stacked bottleneck features," in *Interspeech*, 2014.

[41] David A. van Leeuwen and Rosemary Orr, "Speech recognition of non-native speech using native and non-native acoustic models," in *Interspeech*, 1999.

[42] Ngoc Thang Vu, Yuanfan Wang, Marten Klose, Zlatka Mihaylova, and Tanja Schultz, "Improving asr performance on non-native speech using multilingual and crosslingual information," in *Interspeech*, 2014.

[43] Li Ying and Pascale Fung, "Code switch language modeling with functional head constraint," in *ICASSP*, 2014.

[44] Heike Adel, Dominic Telaar, Ngoc Thang Vu, Katrin Kirchhoff, and Tanja Schultz, "Combining recurrent neural networks and factored language models during decoding of code-switching speech," in *Interspeech*, 2014.

[45] Alan W Black and Tanja Schultz, "Speaker clustering for multilingual synthesis," in *MultiLing*, 2006.

[46] George Dahl, Dong Yu, Li Deng, and Alex Acero, "Context-dependent pre-trained deep neural networks for large vocabulary speech recognition," *IEEE Transactions on Audio, Speech, and Language Processing, Special Issue on Deep Learning for Speech and Langauge Processing*, 2012.

[47] Geoffrey Hinton, Li Deng, Dong Yu, George Dahl, Abdel rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara Sainath, and Brian Kingsbury, "Deep neural networks for acoustic modeling in speech recognition," *IEEE Signal Processing Magazine*, 2012.

[48] Zhi-Jie Yan, Qiang Huo, and Jian Xu, "A scalable approach to using dnn-derived features in gmm-hmm based acoustic modeling for lvcsr," in *Interspeech*, 2013.

[49] The Association for Computational Linguistics and Chinese Language Processing, *http://www.aclclp.org.tw/corp.php*.

[50] S. J. Young, J. J. Odell, and P. C. Woodland, "Tree-based state tying for high accuracy acoustic modeling," in *HLT*, 1994.

[51] C. J. Leggetter and P. C. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density hidden markov models," in *Computer Speech and Language*, 1995.

[52] C.H. Lee and J.L. Gauvain, "Speaker adaptation based on map estimation of hmm parameters," in *ICASSP*, 1993.

[53] LDC94S13A, *Wall Street Journal-based Continuous Speech Recognition (CSR) Corpus Phase II (WSJ1)*, 1994.

[54] Chiu yu Tseng, *Taiwan Asian English Speech Corpus Project (TWNAESOP)*, Academia Sinica, 2009-2012.

[55] Cambridge University Press, *Handbook of the international phonetic association*, 1999.

[56] UCL Division of Psychology & Language Sciences, *SAMPA - computer readable phonetic alphabet*, 1999.

[57] Yi-Jian Wu, Simon King, and Keiichi Tokuda, "Cross-lingual speaker adaptation for hmm-based speech synthesis," in *ISCSLP*, 2008.

[58] Thomas Niesler, "Language-dependent state clustering for multilingual acoustic modeling," *Speech Communication*, 2007.

[59] John R. Hershey and Peder A. Olsen, "Approximating the kullback leibler divergence between gaussian mixture models," in *ICASSP*, 2007.

[60] J. Silva and S. Narayanan, "Upper bound kullback-leibler divergence for transient hidden markov models," *IEEE Transactions on Signal Processing*, 2008.

[61] Yu Dong and Michael L. Seltzer, "Improved bottleneck features using pretrained deep neural networks," in *Interspeech*, 2011.