Making an Effective Use of Speech Data for Acoustic Modeling

Rong Zhang

CMU-LTI-07-016

Language Technologies Institute
School of Computer Science
Carnegie Mellon University
5000 Forbes Ave., Pittsburgh, PA 15213
www.lti.cs.cmu.edu

Thesis Committee:

Alexander I. Rudnicky, Chair
Richard Stern
Tanja Schultz
Karthik Visweswariah

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy
In Language and Information Technologies

Abstract

Automatic recognition of continuous speech has been acknowledged as one of the most challenging problems today. The performance of a continuous speech recognition system highly depends on the availability of sufficient speech data and transcripts of good quality. In most cases, however, carefully prepared in-domain data is not easy to obtain because collecting a large amount of transcribed speech data is normally a time-consuming and expensive process. The acoustic model trained without the support of sufficient training data is less capable in handling the complexity and variability of human speech, and thus performs poorly in real world application. This raises us the questions such as how to effectively exploit the given training data to improve the performance of recognition systems, and how to explore error-prone but informative data sources and incorporate them into acoustic model training.

This thesis summarizes our efforts in investigating solutions to address the above issues. The work can be divided into two parts. We first investigate Boosting algorithm, an ensemble based supervised training approach, which iteratively creates multiple acoustic models with complementary error patterns by manipulating the distribution of training data. While a great deal of research has been conducted on Boosting style acoustic model training, the techniques developed so far have obvious weaknesses that limit their applications in continuous speech recognition. Specifically, conventional Boosting training approach mainly targets at optimizing an utterance level objective function related to sentence error, with relatively less attention paid to reduce word errors. Moreover, the distribution of training data is updated on a sentence basis such that each word is given equal weight in subsequent model training, regardless if the questioned word is a correct or incorrect decoding result. We approach these problems by presenting a novel frame level Boosting algorithm which enables acoustic model training to minimize word or sub-word error rate. We also present an improved sentence hypothesis combination algorithm that uses Neural Nets to incorporate a number of features for generating more desirable combination results. Moreover, we describe the contribution of N-best list re-ranking in Boosting training and hypothesis combination, which is shown to be an effective approach to improve recognition performance.

The second part of this thesis focuses on unsupervised and lightly supervised training techniques that attempt to extend the training set from transcribed speech to closed captioned or even un-transcribed raw speech. Data selection, the approach to identify a subset of data that can best improve recognition performance, appears to be the key issue in this research area. Most

conventional approaches prefer to select data predicted with high confidence for model re-training in order to prevent misrecognized examples from being added to training set. However, this strategy often results in only the examples that match well to the current model being selected and re-training with such examples can become a process that reinforces, rather than eliminates, the estimation bias inherited from the initial transcribed set. To address this problem, we present a novel clustering based data selection strategy that aims to increase the diversity of selected data and makes the selection comply with underlying distribution. Experiments show that the new proposed strategy consistently outperforms conventional approaches in a variety of speech recognition tasks. In addition, we investigate the generalization of Boosting algorithm from supervised training to unsupervised training by using Minimum Bayes Risk decoding, as well as its integration with clustering based data selection for better handling both transcribed and un-transcribed speech data. Experimental results show that the cooperation of data selection and unsupervised Boosting can significantly improve recognition performance.

Acknowledgements

Simply stated, there is no way I could have ever finished this thesis on my own.

First and foremost, I would like to thank my advisor, Professor Alexander I. Rudnicky, for his support, guidance, encouragement and patience during the course of this study. He has been a role model of professionalism and integrity, as well as a good friend. His flexible working style inspires me to think and work independently. He also decisively guided me through the difficult moments when I was frustrated and confused.

I want to thank my thesis committee: Professor Richard Stern, Professor Tanja Schultz, and Dr Karthik Visweswariah. Thank you for your willingness to serve on my committee and your valuable comments and suggestions for my work.

Professor Ravi Mosur, I thank you for your advices and clarifications about CMU Sphinx system. Dr Evandro Gouvea, I thank you for you assistance on matters related to the systems and clusters. Thanks to Rita Singh, Xiang Li, Lingyun Gu, Yi Wu, Dan Bohus, Ziad Al Bawab, Ananlada Chotimongkol, Satanjeev Banerjee, David Huggins-Daines, Wei Xu, Arthur Chan, for your kindly helps.

Special thanks to the colleagues of CMU speech group, past and present. I thank you for the countless fruitful discussions about my research and all of your thoughtful advice as I put this thesis together.

My parents have also been an important support despite the distance. I am grateful for their boundless love and the great sense of family they have provided to me.

Finally and most importantly, I want to thank my wife for her immense love and thoughtful encouragement. Thank you for believing in me and always doing your best to lend me a hand when I need it. You have sacrificed so much for my sake, and I am forever grateful.

Table of Contents

List of Figures

List of Tables

1 Introduction

Continuous speech recognition has been acknowledged as one of the most challenging problems today. There are many issues that contribute to the difficulty of automatically recognizing human speech, such as corruption of noise, variability of speaker and speaking mode, change of environment conditions, transmission of channel, inaccuracy of model assumption, complexity of language, etc.. In addition, as a statistical model based system, a speech recognizer demands sufficient well transcribed speech data for the model training in order to achieve satisfactory performance. However, it is often intractable to fulfill this requirement since the time and expense spent on speech data collection and transcription are not always affordable for many real word applications.

A great deal of work has been conducted to address this problem. One effort is the investigation of *ensemble* based recognition method which is shown to be less sensitive to the amount of training data. Simply speaking, an ensemble approach is a learning algorithm that constructs a set of individual speech engines and then combines their predictions together to produce a more accurate hypothesis. The underlying idea is that the combination of diversified recognizers that have uncorrelated, and ideally complementary, error patterns can offer a more robust generalization capability. The research of ensemble based methods mainly focuses on two issues: ensemble construction and information combination. Among a variety of ensemble construction methods, Boosting appears to be the most successful one in eliminating the correlation between ensemble members and the vulnerability to data sparseness [Freund and Schapire, 1997; Schapire et al., 1997, Breiman, 1998; Cook and Robinson, 1996; Cook et al, 1997; Schwenk, 1999; Zweig and Padmanabhan, 2000; Meyer, 2002; Dimitrakakis and Bengio, 2005]. For combination, word alignment and voting based ROVER demonstrates more advantages than other methods in efficiently combining recognition hypotheses which are in the format of word sequences [Fiscus, 1997; Schwenk and Gauvain, 2000; Goel et al, 2004].

Another effort to overcome the curse of deficiency of training data is to explore and exploit low cost data sources such as closed captioned speech and un-transcribed raw speech, which are much easier to obtain [Jang and Hauptmann, 1999a; Kemp and Waibel, 1999; Lamel et al, 2000; Kamm and Meyer, 2001; Wessel and Ney, 2001; Lamel et al, 2002; Cozman et al, 2003; Chen et al, 2004; Nguyen and Xiang, 2004; Visweswariah et al, 2004; Ma and Matsoukas, 2007]. It is known that these types of data can not be used directly because their transcripts are either unavailable or contain a high percentage of transcription errors. Normally, a data selection approach based on

confidence annotation is employed to filter the error-prone data. The examples evaluated with high confidence are added to acoustic model training while the remains are rejected. This is based on the assumption that if the adopted confidence annotator performs well, the examples marked with high confidence scores are more likely to have correct transcripts, and adding them to training set doesn't deteriorate the quality of training data. Therefore, for some researchers, the question of how to utilize closed captioned or un-transcribed speech is equivalent to how to improve the performance of confidence annotation.

Despite their successes, these previous attempts all have some drawbacks and further investigations are highly desirable. This thesis summarizes our research on these issues. Specifically, we will discuss Boosting training, hypothesis combination, unsupervised training as well as lightly supervised training in order to utilize the characteristics of different type of training data to improve the performance of speech recognition.

1.1 What this Thesis is about

The first part of the thesis considers the problem of building speech recognition systems with transcribed speech dataset of limit size. We address this problem from two perspectives: acoustic model training and hypotheses combination. The thesis first investigates ensemble based acoustic model training approaches, especially Boosting algorithms, which iteratively exploit the training data to create multiple acoustic models with complementary error patterns. We propose a frame level Boosting algorithm. Compared to conventional approach based on a sentence level objective function, our approach aims at the minimization of word or sub-word recognition errors, and is more effective to improve recognition performance. Moreover, our approach enables acoustic model training to focus on the misrecognized part of an utterance rather than, as adopted by conventional approach, giving the whole utterance an equal weight without discriminating recognition error from correct result. For hypotheses combination, we propose a new combination approach that improves hypothesis combination from two aspects. First, N-best list re-ranking technique is employed to identify more reliable hypotheses as the input of combination. Secondly, the desired word is determined by a two-stage scheme in which insertion detection and word scoring are performed separately by incorporating a number of informative features into the decision process. In contrast, conventional ROVER only uses a simple voting strategy that linearly combines two features: frequency of occurrence and confidence score.

The second part of this thesis investigates the use of low-cost error-prone data such as closed captioned speech and un-transcribed speech to improve acoustic model training. Conventional unsupervised or lightly-supervised training approaches normally utilize confidence annotation techniques to select data predicted with high confidence for acoustic model training. In this thesis, we first present a thorough analysis of conventional approaches, showing that only relying on confidence score for data selection can lead to an erroneous estimate to the true distribution when the confidence annotator is highly correlated with the recognizer in the information they use. We then propose a clustering based data section strategy which attempts to make the selection procedure comply with the underlying data distribution and increase the diversity of selected data to better cover acoustic space and phonetic classes. The strategy described is further applied to develop proper approaches for unsupervised and lightly supervised acoustic model training.

Finally, as an effort to unifying both supervised and unsupervised approaches, we investigate the generalization of Boosting algorithm to utilize un-transcribed speech data, as well as the combination of ensemble approach and clustering based data selection approach in unsupervised acoustic model training.

1.2 Thesis Outline

The thesis is organized as follows.

Chapter 2 provides the background information relevant to this thesis. We begin with a discussion of the important technologies used in current statistical speech recognition systems, including feature extraction, Hidden Markov Model, language model, and Viterbi search. We also describe several extensively used training criteria for building acoustic models, such as Maximum Likelihood Estimation, Maximum Mutual Information, Minimum Classification Error and Maximum *a posteriori* Estimation. We then discuss confidence annotation technique, as well as unsupervised and lightly-supervised acoustic model training techniques.

Chapter 3 reviews the experimental environment of this thesis. We first describe the Sphinx III system, especially its acoustic model training toolkit which is uaed as the experiment platform throughout this thesis. We also introduce the five speech datasets on which our algorithms are evaluated: CMU Communicator dataset, ICSI meeting dataset, 1997 Mandarin broadcast news, TDT-4 Mandarin broadcast news, and GALE BN-03 Mandarin dataset.

Chapter 4 presents an overview of ensemble based recognition and combination approaches. We first review ensemble learning algorithms, including Bagging, Boosting, Random Space, Random Forests, etc. We then describe speech specific approaches such as Multi-band and Multi-stream model. Lastly, we discuss combination approaches and focus on hypotheses combination techniques such as ROVER, consensus network and lattice combination.

Chapter 5 presents our efforts to improve Boosting based acoustic model training techniques. The conventional utterance level Boosting algorithm is described at the beginning. We then present a novel frame level Boosting training algorithm to overcome the weakness of conventional approach. Next, we present an improved hypothesis combination scheme that uses Neural Nets to incorporate a number of features for generating more desirable recognition results. In addition, investigations of post-processing techniques e.g. N-best list re-ranking which help to improve recognition performance are also reported.

Chapter 6 reviews unsupervised and lightly supervised acoustic model training technologies using low cost data such as un-transcribed speech or closed captioned speech. We describe in detail the confidence scoring based selective training method and its applications in identifying correctly transcribed speech data. Lastly, we present a discussion that shows only relying on confidence metric for data selection can lead to an erroneous estimate to the underlying distribution and cause degradation of recognition accuracy.

Chapter 7 presents our solution to the problem discussed in Chapter 6. We begin with the description of a novel clustering based data selection strategy which works on two aspects together: maintaining the correctness of transcripts and increasing the diversity of selected data. We also describe a normalization scheme that converts an arbitrarily long utterance into a fixed size vector in order to perform clustering in utterance space. We then report the experimental results of the proposed approach in unsupervised and lightly supervised acoustic model training.

Chapter 8 investigates the extending of Boosting algorithm to unsupervised acoustic modeling. We first describe frame level unsupervised Boosting algorithm based on Minimum Bayes Risk decoding. We then discuss the combination of the unsupervised Boosting algorithm and clustering based data selection algorithm in utilizing both transcribed and un-transcribed speech data. Encouraging experiment results on ICSI meeting recognition are also reported

Finally we conclude in Chapter 9 by summarizing the major contributions of this thesis and highlighting some directions for future research.

2 A Review of Automatic Speech Recognition Technologies

This chapter presents the background information relevant to the thesis. We begin with a brief introduction of continuous speech recognition process, including discussions from feature extraction to hypothesis generation. Specially, we describe important modules and techniques employed by contemporary speech recognition systems, such as of Hidden Markov Model (HMM), language model and Viterbi search. Next is an overview of training criteria for building acoustic models. We will cover Maximum Likelihood Estimation (MLE), Maximum *a posteriori* Estimation (MAP), as well as approaches of Discriminative training. We then discuss confidence annotation technique which is extensively used in speech recognition to evaluate the correctness of decoding results. We also describe unsupervised and semi-supervised learning approaches which are attempts to address the persistent problem of lacking well-transcribed speech data for acoustic model training. This chapter is concluded by a discussion of the characteristics of speech that distinguishes continuous speech recognition from other classification problems.

2.1 Statistical Speech Recognition

The goal of automatic speech recognition is to translate acoustic spoken utterances into other representations, such as sequence of words, which are easier for further computer or human processing. Figure 2.1 illustrates the general architecture of an automatic speech recognition system [Ney, 1990].

As Figure 2.1 shows, a speech recognition system mainly consists of five major building blocks: Feature Extraction, Hypothesis Search, Lexical Model, Acoustic Model and Language Model.

- *Feature Extraction*. The task of this module is to detect the analog speech signal, and parameterize it into a sequence of acoustic feature vectors $\mathbf{x} = (x_1, x_2, ..., x_T)$.

- *Hypothesis Search*. The task of this module is to find the best word string $h^* = (w_1, w_2, ..., w_K)$ with the highest *a posteriori* probability, that

$$h^* = \arg\max_h P(h \mid \mathbf{x}) = \arg\max_h P(h, \mathbf{x}) = \arg\max_h P(h)P(\mathbf{x} \mid h) \tag{2-1}$$

for a given speech input $\mathbf{x} = (x_1, x_2, ..., x_T)$.

- *Lexical Model*. A dictionary that defines the pronunciations and their variants for the words to be recognized, in terms of phonemes or other sub-word units, e.g. *triphones*.

- *Acoustic Model*. A statistical model that estimates $P(\mathbf{x} \mid h)$, the probability of observing a sequence of acoustic feature vectors $\mathbf{x}$ conditioned on a hypothesized word string h. Gaussian Mixtures based *Hidden Markov Model* (HMM) is the most widely adopted acoustic modeling technology in contemporary speech recognition systems.

- *Language Model*. A statistical model that provides measurement of $P(h)$, the prior probability of a particular word string h being spoken. N-gram model has been established as the *de facto* standard for large vocabulary continuous speech recognition.

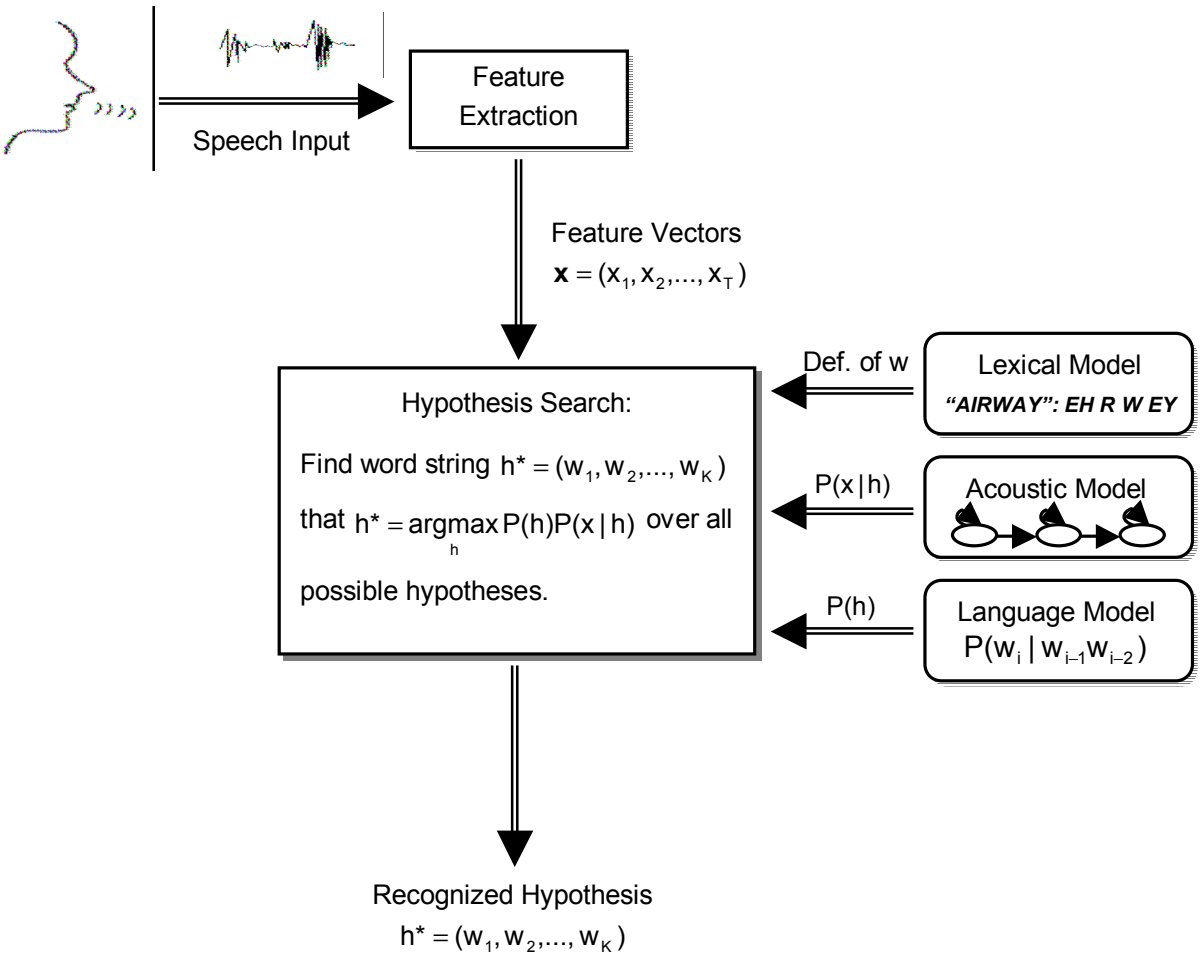

Figure 2.1 Block diagram of a continuous speech recognition system

2.1.1 Feature Extraction

A speech recognition system can not recognize the speech waveform directly. The raw speech signal contains too much redundant and unwanted information related to speakers and recording channels, as well as a variety of background and random noise. Normally, speech signal is converted into a parameterized sequence of feature vectors by front-end processing to emphasize the characteristics of spoken words and suppress other irrelevant information.

The first step of feature extraction is to divide the continuous speech signal into short overlapped time slices, which are called *frames*. The duration of a frame is typically about 20~30 ms based on the observation that speech signal is quasi-stationary within such a short interval in the sense that its statistical properties do not change too much. After that, each frame is then windowed and transformed into frequency domain via spectral analysis to obtain an acoustic feature vector as the representation of that time piece.

Typical features used in continuous speech recognition include *Linear Predictive Coefficients* (LPC), *Cepstral Coefficients*, *Mel-frequency Cepstral Coefficients* (MFCC) and *Perceptual Linear Predictive Coefficients* (PLP) [Davis and Mermelstein, 1980; Hermansky, 1990]. Some of them are motivated by the nature of human hearing. For the experiments described in this thesis, the inputs to speech recognizers are sequences of vectors composed of MFCC features and their first and second order temporal derivatives, named as delta-MFCC and delta-delta-MFCC, respectively. The features are calculated using CMU Sphinx III toolkit [Ravishankar, 1996] (http://www.cs.cmu.edu/~rsingh/sphinxman/s3manual.html).

2.1.2 Acoustic Model

Model Unit

The first question that acoustic modeling need to answer is to decide the best model unit for speech recognition. The objective of speech recognition is to transcribe speech signal into text, i.e. word string. However, this doesn't mean it's a good choice to build acoustic model on word level. In the case of large vocabulary speech recognition, there are so many words to be trained that make it intractable to collect sufficient training data. Moreover, the training process has to be restarted in order to add new word to the vocabulary.

A better option is to use *Sub-word* unit, such as syllable or phoneme which can be modeled and shared across words. Phoneme is widely accepted as model unit for English speech recognition. There are about 50 phonemes in spoken English. Such a small number means it's unnecessary to worry about the deficiency of training data. Furthermore, new word can be easily added to vocabulary by defining the pronunciation in terms of phonemes.

The weakness of phoneme model is that it doesn't consider the influence of left and right context, so unable to provide accurate description for speech signal. *Triphone*, a context dependent model unit that consider not only the current phoneme but also its left and right neighbors, is proposed to address this weakness, and is potentially capable to capture the coarticulation effect between adjacent speech units [Bahl et al, 1980; Hwang, 1993].

Unfortunately, the number of triphones can be very large. With 50 phonemes, there can be up to 50^3 triphones. Therefore, these triphone models have to be tied together to form a smaller model set, in order to maintain a reliable estimation for the parameters. For example, in CMU Sphinx system, triphones are finally clustered into equivalence classes called *senones* [Lee, 1990].

Hidden Markov Model

In order to find the most likely word string $h^* = (w_1, w_2, ..., w_K)$ given the acoustic observation $\mathbf{x} = (x_1, x_2, ..., x_T)$ in Equation (2-1), current continuous speech recognition systems evaluate the term $P(\mathbf{x} \mid h)$ through the means of Hidden Markov Models. A HMM is a network consisting of a set of connected states, each of which characterizes certain part of model target. For example, when using a 3-state HMM to model a phoneme, the three states correspond to the beginning, the middle and the ending segment of the phoneme respectively. Mathematically, a HMM, λ, can be expressed as a triplet that $\lambda = \{S, \mathbf{A}, \mathbf{B}\}$, where S is the state set that $S = \{s_1, s_2, ..., s_N\}$ which means this is a N-state HMM, $\mathbf{A}$ is the matrix of transition probabilities that $\mathbf{A} = [a_{i,j}]$ in which $a_{i,j}$ denotes the probability moving from state s_i to state s_j, and $\mathbf{B}$ is the set of observation probability functions that depicts the likelihood of emitting an acoustic feature at a particular state. In the case of continuous HMM, the output probability usually takes the form of Gaussian Mixtures that

$$b(s_i, \mathbf{x}) = P(\mathbf{x} \mid s_i) = \sum_k c_{s_i,k} N(\mathbf{x}; \mu_{s_i,k}, \Sigma_{s_i,k}) \qquad (2\text{-}2)$$

Where $N(\mathbf{x};\mu_{s_i,k},\Sigma_{s_i,k})$ is a Gaussian distribution with mean $\mu_{s_i,k}$ and variance matrix $\Sigma_{s_i,k}$, and $c_{s_i,k}$ is the mixture weight associated to $N(\mathbf{x};\mu_{s_i,k},\Sigma_{s_i,k})$. Figure 2.2 illustrates a simple example of a 3-state HMM which employs a left-to-right topology. The only allowable transitions in this HMM are self-transition that is back to the current state or transition to the state immediately on the right.

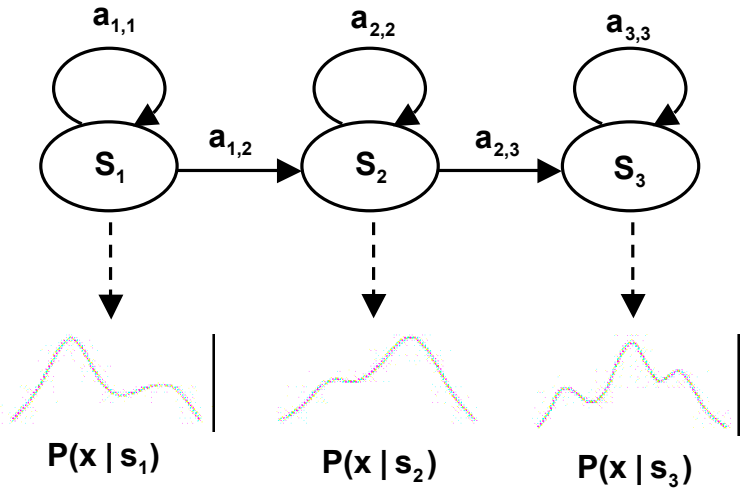

Figure 2.2 An example of a 3-State left-to-right hidden markov model

There are three problems of interest involved in the training of HMM and recognition using HMM [Rabiner, 1989; Rabiner and Juang, 1993].

The Evaluation Problem: Given a HMM and a sequence of acoustic observations, what is the probability of the observations being generated by HMM? This is addressed by forward-backward algorithm [Baum, 1972].

The Decoding Problem: Given a HMM and a sequence of acoustic observations, what is the most likely sequence of state transition in the HMM that produces the observation? This corresponds to the recognition process that seeks the most likely hypothesis. The is addressed by using the Viterbi algorithm [Viterbi, 1967].

The Training Problem: Given the topology of HMM and sequences of acoustic observations, How to learn the model's parameters so that it has the maximum probability of generating the observations? This is addressed by the Baum-Welch algorithm [Baum, 1972].

2.1.3 Language Model

For speech recognition, especially for large vocabulary speech recognition, *language model* (LM) contributes significantly to the performance of system [Rosenfeld, 2000]. Generally speaking, language model aims to provide a statistical framework to estimate the prior probability of a word sequence $h = (w_1, w_2, ..., w_K)$ being spoken. Using the chain rule, the probability of the sequence can be represented by the products of the conditional probabilities of the questioned words given their histories:

$$P(h) = P(w_1, w_2, ..., w_K) = \prod_{k=1}^{K} P(w_k \mid w_1, ..., w_{k-1}) \tag{2-3}$$

In practice, one can not obtain a reliable estimation for $P(w_k \mid w_1, ..., w_{k-1})$ since an arbitrarily long history would demand enormous amount of training data. *N-gram* model is proposed to address this problem, in which the history is truncated into the most recent *n*-1 words with the assumption that a *n*-1 order Markov source is sufficient to generate the language. For example, for a *trigram* model, the most widely used language model for speech recognition, only the previous two words are concerned. That is,

$$P(w_k \mid w_1, ..., w_{k-1}) \approx P(w_k \mid w_{k-2}, w_{k-1}) \tag{2-4}$$

Such simplification benefits language modeling by reducing the number of free parameters into an affordable range, and making it possible, in combination with smoothing and back-off techniques, to train language model on a limited size corpus. However, this assumption has obvious weakness that it doesn't take into account the contexts wider than *n*-1 words, and thus loses a great deal of useful language information, such as long distance word correlations, syntactic constraints and semantic consistence.

2.1.4 Search

The goal of the search module in a speech recognition system is to find the most likely hypothesis given a sequence of acoustic feature vectors. Unlike other common multiple-class classification problems in which the number of possible classes is limited to small size, continuous speech recognition system has to work in a huge hypothesis space for which the regular classification

strategy loses efficacy. For example, suppose the system has a vocabulary of M words and the maximum length of a sentence hypothesis in terms of number of word is set to K. The number of possible hypothesis is at least:

$$1 + M + M^2 + M^3 + ... + M^K = \frac{M^{K+1} - 1}{M - 1} \tag{2-5}$$

Please note the number listed above doesn't consider the temporal segment information of each word. Namely, we don't know exactly when a word begins and ends. If we take account of such boundary information into calculation, the number of possible hypotheses would be much larger. Apparently, such a huge number makes it impossible for the recognizer to traverse every hypothesis and compute probability for them.

The complexity of this search problem could be reduced by using dynamic programming, which breaks the global search down to successive local searches. Viterbi decoding is a dynamic programming algorithm that searches the state space for the most likely state sequence that accounts for the input speech. Viterbi decoding is performed in a time-synchronous fashion that the input speech is processed one frame at a time, and all the states are updated for that frame before moving on to the next frame. In addition, a beam pruning technique is usually applied to limit the search by pruning out the less likely partial hypothesis.

2.1.5 Measuring Performance

The performance of a speech recognition system is measured in two aspects: speed and accuracy. Speed is very important for real time systems e.g. dialog system, for which a quick response to user's inquiry is one of the basic requirements. In this thesis, however, we focus only on the recognition accuracy since our goal is to investigate the effectiveness of proposed algorithms rather than to improve their efficiency.

In recognition, a recognizer can generate extra errors by inserting words by mistake. So the accuracy of a recognizer can not be measured by simply counting the number of correctly recognized words. For most applications, instead, the recognition performance is measured by *Word Error Rate* (WER), a metric relevant to how many mistakes the recognizer made. The definition Word Error Rate is given as follows,

$$Word\ Error\ Rate = \frac{Substitutions + Deletions + Insertions}{Number\ of\ Spoken\ Words} \tag{2-6}$$

Word Error Rate is calculated by performing alignment between hypothesis and reference. So it is difficult to express this metric in a form of differentiable function and integrate it into acoustic and language model training. Please note that the definition in Eq. (2-6) doesn't consider word segmentation information. This means, two hypotheses would have the same score even if one is better than another in providing more accurate word boundary.

2.2 Training Criteria for Acoustic Modeling

This section presents a brief review of the training criteria used in acoustic modeling, including *Maximum Likelihood Estimation* (MLE), *Discriminative Training* and *Maximum a posteriori Estimation* (MAP). These criteria have been investigated by speech community for decades and successfully applied to many real systems. However, they unavoidably have certain weaknesses which become the motivation of new approaches, e.g. ensemble based methods investigated in this thesis, which aim to achieve better performance than these classic criteria.

2.2.1 Maximum Likelihood Estimation

MLE is the dominant principle that most speech recognition systems are trained with [Rabiner and Juang, 1993; Jelinek, 1998]. The performance of MLE training usually works as the baseline for evaluating new training methods. Given the training instances and their class labels, MLE criterion optimizes model parameters by maximizing the class conditional probability of the training data. For speech recognition, this criterion could be expressed as follows. Supposing we have a training set $\Psi = \{(\mathbf{x}_i, y_i) \mid 1 \le i \le N\}$ where $\mathbf{x}_i$ is the sequence of acoustic features extracted from the *i*-th utterance in training corpus, and y_i is its transcript, MLE aims to find a model λ^* that

$$\lambda^* = \arg\max_{\lambda} P(\Psi \mid \lambda) = \arg\max_{\lambda} \prod_{i=1}^{N} P(\mathbf{x}_i, y_i \mid \lambda) = \arg\max_{\lambda} \sum_{i=1}^{N} \log P(\mathbf{x}_i, y_i \mid \lambda) \tag{2-7}$$

Using chain rule, the probability $P(\mathbf{x}_i, y_i \mid \lambda)$ can be further expressed as

$$P(\mathbf{x}_i, y_i \mid \lambda) = P(y_i \mid \lambda) * P(\mathbf{x}_i \mid y_i, \lambda) \tag{2-8}$$

$P(y_i \mid \lambda)$ denotes the prior probability of the word string y_i being spoken. Its value is provided by language model. $P(\mathbf{x}_i \mid y_i, \lambda)$ denotes the conditional probability of observing acoustic features $\mathbf{x}_i$ given y_i. Its value is provided by acoustic model.

For most speech recognition systems, the training of language models is a separate process independent of the training of acoustic models. We thus exclude it from the following discussion. Eliminating the influence of language model, the MLE criterion for acoustic modeling can be described as to find an acoustic model λ^* that

$$\lambda^* = \arg\max_{\lambda} \sum_{i=1}^{N} \log P(\mathbf{x}_i \mid y_i, \lambda) \tag{2-9}$$

It has been shown that, if the assumption of the adopted statistical model is correct and sufficient training data is available, MLE training can result in a perfect estimation of class conditional probability $P(\mathbf{x} \mid y, \lambda)$. However, it's difficult to fulfill these requirements in practice where the amount of training data is limited and the model assumption is usually empirically determined. Moreover, MLE training only considers the correct classes of training instances without taking into account the competing classes. In speech recognition, this may cause unfavorable result. As we know, the transcripts y of an utterance $\mathbf{x}$ and other competing hypotheses h that $h \neq y$ often share same words or phonemes. MLE training aiming at maximizing $P(\mathbf{x} \mid y, \lambda)$ could also boost the value of $P(\mathbf{x} \mid h, \lambda)$. As a consequence, the decoding with acoustic model obtained from MLE may not be able to output the hypothesis with lowest word error rate.

2.2.2 Discriminative Training

As discussed above, the result of MLE is a set of model parameters which maximize the likelihood of observing the training data given their labeled class. The estimation of the model parameters of a class only depends on the training data belonging to this class. In contrast to MLE, Discriminative Training attempts to incorporate additional knowledge by also using the data associated with competing classes. Namely, discriminative criteria not only try to maximize $P(\mathbf{x} \mid y, \lambda)$, the conditional probability of the training data given the correct class, but also try to

minimize $P(\mathbf{x} \mid h, \lambda)$, the conditional probability given alternative classes, and therefore to increase the separability among classes.

The research of discriminative training by speech community started from 1980s, the revolutionary era of speech recognition when HMM and MLE training obtained widespread applications. Nevertheless, speech community was soon distressed by the observation that the high likelihood doesn't always lead to the low recognition error. Discriminative criteria were then proposed to tackle the shortcomings of MLE training, and evolved to be a family of training approaches with a number of variants. Two of them, *Maximum Mutual Information* (MMI) and *Minimum Classification Error* (MCE), are discussed here [Juang and Katagiri, 1992; Schluter, 2000; Schluter et al, 2001].

Maximum Mutual Information (MMI)

The MMI criterion was proposed as an alternative to MLE in order to overcome the weakness of MLE, in particular the problem of using MLE with an inaccurate model assumption. The idea of the MMI criterion is to minimize the conditional model-based entropy $E_\lambda(h \mid \mathbf{x})$ over random hypothesis variable h given the input acoustic feature variable $\mathbf{x}$. This corresponds to find parameters for model λ that provide as much information as possible about the desired hypothesis given the input pattern $\mathbf{x}$.

The model based entropy for hypothesis variable h is defined as

$$E_\lambda(h) = -\sum_h P_{true}(h) \log P(h \mid \lambda) \tag{2-10}$$

where $P_{true}(.)$ denotes the true distribution for data generation while $P(. \mid \lambda)$ denotes the empirical model based on estimated distribution. This entropy of hypothesis measures the uncertainty of what hypothesis is spoken without knowing any acoustic information.

The conditional entropy of hypothesis given input acoustic feature is defined as

$$E_\lambda(h \mid \mathbf{x}) = -\sum_{h,\mathbf{x}} P_{true}(h, \mathbf{x}) \log P(h \mid \mathbf{x}, \lambda) \tag{2-11}$$

This entropy measures the uncertainty to predict what hypothesis is spoken given the input feature sequence $\mathbf{x}$. The amount of information provided by $\mathbf{x}$ about h is then represented as the mutual information, the difference between these two entropies.

$$I_\lambda(h;\mathbf{x}) = E_\lambda(h) - E_\lambda(h \mid \mathbf{x}) = \sum_{h,\mathbf{x}} P_{true}(h,\mathbf{x}) \log \frac{P(h,\mathbf{x} \mid \lambda)}{P(h \mid \lambda)P(\mathbf{x} \mid \lambda)} \tag{2-12}$$

The goal of MMI estimation is to maximize the mutual information $I_\lambda(h;\mathbf{x})$ by optimizing model λ. As the training data is assumed to be representative, this is equivalent to choose value for model λ that maximizes

$$f_{MMI}(\lambda) = \frac{1}{N}\sum_{i=1}^{N} \log \frac{P(h = y_i, \mathbf{x} = \mathbf{x}_i \mid \lambda)}{P(h = y_i \mid \lambda)P(\mathbf{x} = \mathbf{x}_i \mid \lambda)} = \frac{1}{N}\sum_{i=1}^{N} \log \frac{P(\mathbf{x}_i \mid y_i, \lambda)}{\sum_h P(h, \mathbf{x}_i \mid \lambda)} \tag{2-13}$$

Comparison between MMI criterion and MLE criterion shows that the latter is only concerned to maximize $P(\mathbf{x} \mid y, \lambda)$, the class dependent conditional probability for the correct label, while the former maximizes the difference between $P(\mathbf{x} \mid y, \lambda)$ and background probability $P(\mathbf{x} \mid \lambda)$. Thus the MMI criterion is more discriminative than MLE criterion. The advantage of MMI training is that, maximizing $f_{MMI}(\lambda)$ is more reasonable than maximizing the likelihood of training data when the model assumption is not accurate. However, the computation cost of MMI training is more expensive than that of MLE training due to the need to consider all of the possible classes instead of the correct one only.

Minimum Classification Error (MCE)

As the name suggests, the MCE criterion is designed to minimize the classification error on the training set. A simple zero-one cost function would measure the error rate perfectly, but it violates the constraint that the function should be continuously differentiable in the model parameter space. In order to allow for parameter optimization using gradient descent based methods, MCE criterion uses a smoothed version of zero-one cost function by sigmoid transformation.

To measure the difference between the observation probabilities given the correct class y and other competing hypotheses $h \neq y$, a misclassification function is defined as

$$d(\mathbf{x}, y; \lambda) = -g(\mathbf{x}, y; \lambda) + [\frac{1}{|h| - 1}\sum_{h \neq y} g^\eta(\mathbf{x}, h; \lambda)]^{1/\eta} \tag{2-14}$$

where $g(\mathbf{x}, h; \lambda) = \log P(\mathbf{x}, h \mid \lambda)$, $|h|$ denotes the number of the possible hypotheses including y, and η is a positive parameter that controls how the competing hypotheses are weighted. In

the limit as $\eta \to \infty$, only the most likely competing class is taken into account while the others are ignored. In this case, the misclassification function is degraded into

$$d(\mathbf{x}, y; \lambda) \approx -g(\mathbf{x}, y; \lambda) + g(\mathbf{x}, h_1; \lambda) \tag{2-15}$$

where h_1 represents the best competitor, i.e. the top-1 hypothesis in the N-best list. The simplified misclassification leads to a variant of MCE criterion allowing for fast computation.

The interpretation of the misclassification function is that a positive value implies an utterance level recognition error, while a negative value implies the decoding result is likely to be correct (when choose a large value for η).

The misclassification measure is then embedded into s*igmoid* function

$$sigmoid(d) = \frac{1}{1 + \exp(-\rho * d + \theta)} \tag{2-16}$$

where ρ is the parameter which defines how sharply the sigmoid function changes at the transition point, and θ defines the location of the transition point. Apparently, the sigmoid function is a continuously differentiable function with the capability to approximate the zero-one cost function as showed in (2-17).

$$sigmoid(d) \begin{cases} \approx 0 & d << 0 \\ < 0.5 & d < 0 \\ > 0.5 & d > 0 \\ \approx 1 & d >> 0 \end{cases} \tag{2-17}$$

The complete objective function of MCE for the entire training set is then formulated as

$$\begin{aligned} & f_{MCE}(\lambda) \\ &= \frac{1}{N} \sum_{i=1}^{N} sigmoid(d(\mathbf{x}_i, y_i; \lambda)) \\ &= \frac{1}{N} \sum_{i=1}^{N} \frac{1}{1 + \exp\{-\rho * [-g(\mathbf{x}_i, y_i; \lambda) + (\frac{1}{|h| - 1} (\sum_{h \neq y^{(i)}} g^{\eta}(\mathbf{x}_i, h; \lambda))^{1/\eta})]\}} \end{aligned} \tag{2-18}$$

Similar to MMI criterion, MCE is also a discriminative criterion which aims to increase the separability between desired hypothesis and alternative hypotheses. MCE provides a framework that directly links the classification error with a continuous and differentiable loss function. Minimizing the MCE loss function is a way of minimizing the actual number of

misclassifications in training set. It has been demonstrated that given enough training data MCE could yield a classifier that is close to the Bayes decision rule, and thus minimizes the expected classification error rate.

2.2.3 Maximum A Posteriori Estimation (MAP)

In the discussion of MLE and Discriminative Training, the model λ itself is not treated as a random variable. This is somehow against the Bayesian learning theory in which everything could be random. MAP provides a framework that considers the uncertainty of the model parameter λ and incorporates it into the model optimization [Therrien 1992; Chien et al, 1997].

Suppose we have a training set $\Psi = \{(\mathbf{x}_i, y_i) \mid 1 \le i \le N\}$ where $\mathbf{x}_i$ is the sequence of acoustic features of the *i*-th utterance, and y_i is its transcript. MAP criterion is to find a value for the model parameters λ that maximizes the *posterior* probability of λ given training data.

$$\begin{aligned}\lambda^* &= \arg\max_{\lambda} P(\lambda \mid \Psi) \\ &= \arg\max_{\lambda} \frac{P(\lambda, \Psi)}{P(\Psi)} = \arg\max_{\lambda} P(\lambda, \Psi) = \arg\max_{\lambda} P(\lambda) P(\Psi \mid \lambda) \\ &= \arg\max_{\lambda} P(\lambda) \prod_{i=1}^{N} P(\mathbf{x}_i, y_i \mid \lambda)\end{aligned} \tag{2-19}$$

The normalizing factor $P(\Psi)$ is discarded from the MAP estimation, since $P(\Psi) = \int_{\lambda} P(\lambda, \Psi) d\lambda = \int_{\lambda} P(\lambda) P(\Psi \mid \lambda) d\lambda$, which has no effect on looking for the optimum of λ. Please note that in (2-19) the training instances are assumed i.i.d., so $P(\Psi \mid \lambda) = \prod_{i=1}^{N} P(\mathbf{x}_i, y_i \mid \lambda)$.

In (2-19), $P(\lambda)$ represents the prior probability of model parameter λ, which statistically describes our knowledge about λ before we observe the training data. Contrasting to MLE methods, MAP maximizes $P(\lambda)P(\Psi \mid \lambda)$ instead of $P(\Psi \mid \lambda)$ only. In this context, MLE could be viewed as a special case of MAP, which assumes that all values of λ are equally likely.

The Use of the prior information in parameter estimation is particularly important for dealing with the difficulties arising from sparse training data, for which the classical MLE approach

usually gives poor estimates of model parameters. MAP has been shown to be reasonably effective for speaker adaptation, in which it is difficult to collect sufficient training data for a new speaker. On the other hand, the performance of MAP estimation highly depends on the prior probability $P(\lambda)$. If our prior knowledge about λ is not accurate, MAP may lead to an erroneous estimate of λ.

2.3 Confidence Annotation

Confidence annotation, which task is to evaluate the correctness or reliability of recognition result, is a highly desirable capability for continuous speech recognition systems, since acting upon a misrecognized hypothesis can incur a high cost to the user either though an undesired side-effect or through time wasted on correction.

Chase [Chase, 1997] proposed the following framework for incorporating a confidence metric into a recognition system: (1) At what level should the confidence annotation be made; (2) What is the right way to define what is an error and what isn't; (3) What features are useful and how useful; (4) How to build a model combining the various features to create a confidence annotation; (5) How to measure the goodness of the feature and model. The answers to these questions depend on the particular application that incorporates the confidence annotator. For example, in the CMU Communicator system [Rudnicky et al, 1999], a telephone based dialog system that supports planning in travel domain, both the utterance level and word level confidence annotation are used. The former measures the confidence of the whole utterance, and the latter supplies the reliability description of each single word.

The key problem in confidence annotation is the selection of effective features [Wessel et al, 1998; Zhang and Rudnicky, 2001] and a variety of features have been proposed. These features are extracted from a variety of information source, e.g. acoustic, language model, N-best list or word lattice. Most are based on information from the decoder and moreover have the disadvantage that they unavoidably overlap in the information that they use, as is apparent in the common observation that the performance achieved by all the features together isn't much better than that with only the best feature. These features are for the most part redundant with the information used to generate hypotheses in the first place and so contribute little new information, particularly at the acoustic level.

The construction of a confidence annotator can be treated as a problem in pattern classification and a large variety of classification approaches can be considered for use, such as Neural Network, Decision Tree, Bayes Network, Support Vector Machine, AdaBoost, etc [Carpenter et al, 2001].

2.4 Unsupervised and Lightly Supervised Acoustic Model Training

For acoustic model training, collecting a large number of well-transcribed speech data is a time-consuming and expensive process. One the other hand, massive amount of un-transcribed raw data or closed captioned data are relatively easy to obtain. Thus there exists the need for an automatic learning procedure that can use both transcribed and un-transcribed data for model training. This family of approaches is usually referred to as unsupervised training, which uses un-transcribed data, or lightly supervised training, which uses closed captioned data [Jang and Hauptmann, 1999a; Kemp and Waibel, 1999; Lamel et al, 2000].

The following scenario is usually considered in unsupervised and lightly supervised acoustic model training. An initial seed model λ_0 is learned from the transcribed set and then applied to recognize the un-transcribed utterance. A confidence metric $f_c(h;\mathbf{x})$, which has been learned from available data, is used to provide each sentence hypothesis with a score measuring the likelihood of correctness for the recognition result given by λ_0. A certain number of recognized utterances, which hypotheses are marker with high confidence by $f_c(h;\mathbf{x})$, are then added to the transcribed set for training a new model λ_1. This process repeats until all of the un-transcribed data are exhausted or some early halting criterion is met.

Semi-supervised learning has elicited growing interests in various research fields and many novel approaches have been proposed with promising improvement of performance. However, researchers are divergent on the effectiveness of using data with high confidence score. For example, [Kemp and Waibel, 1999] suggests that this kind of data can't add substantial new information to the existing recognizer, while [Wessel and Ney, 2001] shows that the good performance is mainly due to the contribution of this portion.

2.5 Challenges of Continuous Speech Recognition

The performance of an ASR system is negatively impacted by a number of issues, such as corruption of noise, variability of speaker and speaking mode, change of environment conditions, transmission of channel, inaccuracy of model assumption, complexity of language, etc.. Similar problems also exist in other classification tasks, but are particularly serious in continuous speech recognition. In addition, some distinct properties of continuous speech recognition add further challenges to the task. Some of them are listed as follows.

- Both the input instance $\mathbf{x} = (x_1, x_2, ..., x_T)$ and output hypothesis $h = (w_1, w_2, ..., w_K)$ of a continuous ASR system are sequences that can be arbitrarily long. In contrast, for many common classification problems, the length of the input feature vector is fixed, and the output is only a simple class label.

- As we have discussed, the space of sentence hypothesis for a consecutive utterance can be very large. As a practical matter, it is impossible for a speech recognizer to enumerate all the hypotheses to estimate their probabilities. Some special techniques have to be used to reduce the search space, e.g. dynamic programming and pruning. However, these can cause the correct hypothesis to be early deleted from the search process.

- In LVCSR systems, sub-words, e.g.. *triphones* or *senones*, are chosen as the speech unit for acoustic model training. On the other hand, the speech corpora are organized as a set of utterances that are transcribed into string of words. The segment information necessary for sub-word model training is not available in the speech dataset due to the expense of transcription. One has to use *Viterbi* based forced-alignment or other methods to guess a likely boundary for each sub-word appearing in an utterance.

- There is also a considerable mismatch between the training criteria of acoustic modeling and the measurement of recognition performance. The speech community often uses *Word Error Rate* (WER), a word level metric to evaluate system performance by computing recognition errors in terms of *substitution*, *deletion* and *insertion*. However, as we have shown, current acoustic modeling criteria, e.g. MLE, MCE and MMI, focus on how to reduce sentence level training errors rather than word level errors. This mismatch leads to an often observed phenomenon where the hypothesis with the highest likelihood is not the one with the lowest word error rate.

Continuous Speech Recognition has been acknowledged as one of the most challenging classification tasks. The primary goal of this thesis is to investigate suitable acoustic model training strategies that can fulfill the special requirements of continuous speech recognition. Next several chapters will cover our research results on Boosting training, ROVER combination, unsupervised training as well as lightly supervised training.

3 Experimental Environments

This chapter presents a brief overview of the speech recognition system, CMU Sphinx III system, and speech corpora used in our experiments discussed by this thesis. Even though we report results only using Sphinx III system and the corpora described later in this chapter, the algorithms we will propose are independent of specific speech recognizer and corpora, and are applicable to any other continuous speech recognition systems and different domains. The reason of selecting Sphinx III system as the platform of our experiments is that, as a well known and open-source system (http://cmusphinx.sourceforge.net/html/cmusphinx.php), it can provide reader a trustable and easily obtainable basis to understand and repeat our results.

3.1 CMU Sphinx III System

The Sphinx III system is used as the baseline speech recognition system to evaluate the algorithms to be proposed in this thesis. Sphinx III is a large vocabulary, speaker independent, HMM based continuous speech recognition system developed at Carnegie Mellon University [Lee, 1988; Huang et al, 1993; Ravishankar, 1996]. It differs from earlier systems, i.e. Sphinx I and Sphinx II systems, in the form and topology of probability density functions adopted to describe acoustic observations. Sphinx III is the first version of CMU-developed fully continuous HMM system in which the Gaussian distributions are state-specific and thus support a more detailed modeling of acoustic variability of speech. In contrast, Sphinx I is a discrete HMM recognizer based on Vector Quantization, and Sphinx II is a semi-continuous HMM recognizer that all the phonetic units share a small set of global Gaussian distributions,

Sphinx III is based on contextual phoneme or triphone models which are capable of handling coarticulation effects of spoken language by considering the influence of neighboring context. However, because the number of parameters to be estimated for triphone models is large, and because the amount of training data required for a reliable estimation is enormous, Sphinx III uses a sub-phonetic clustering approach to share parameters among similar triphones. The output of clustering is a set of shared distributions, called as tied states or senones. Compared to triphones, the total number of free parameters to be estimated for senone models is reduced significantly.

Next we introduce how acoustic model training is performed using Sphinx III system.

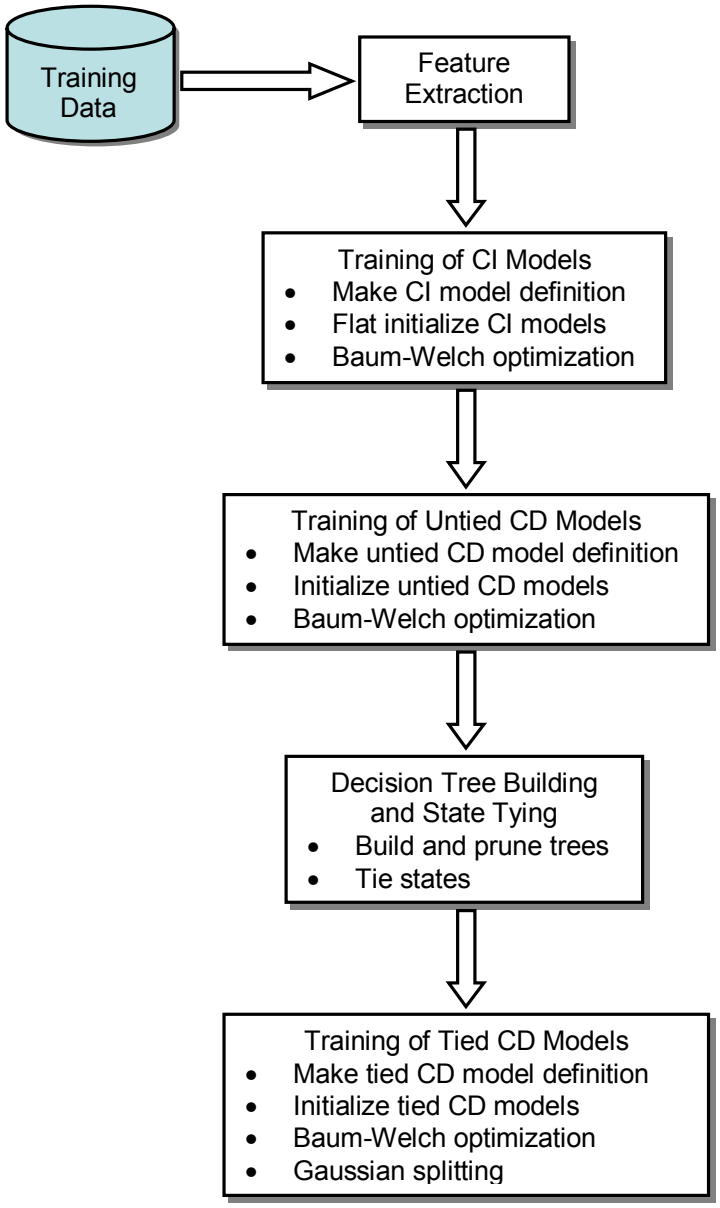

Figure 3.1 The chart of Sphinx III acoustic model training

Sphinx III provides user a complete toolkit including all the necessary functions for acoustic model training. As illustrated in Figure 3.1, the acoustic model training procedure using Sphinx III can be roughly divided into five steps from the point of feature extraction to the point of senone model training (http://www.cs.cmu.edu/~rsingh/sphinxman/s3manual.html).

- *Feature Extraction*. The first step is to convert input speech into feature vectors. In this thesis, 12 dimension MFCC plus 1 dimension power are extracted as the base feature for each frame. This feature is then extended to 39 dimensions by augmenting first order and second order time derivatives, in order to capture the transition trend of spectrum. In addition, Cepstral mean normalization (CMN) is optional to be applied at the utterance level to remove statistical biases of the mean which might have been introduced by linear channel distortion.

- *Training of context independent (CI) models*. This step is to train HMM model for each context independent phonemes. The model parameters include state transition probability, component weight of Gaussian Mixture, mean and variance of Gaussian distribution. To initialize the means and variances, global values of these parameters are first calculated and then copied into appropriate positions in the parameter files. An iterative EM learning procedure, based on Baum-Welch algorithm [Rabiner and Juang, 1993] is employed to optimize model parameters with respect to maximizing the probability of generating input acoustic observations. Each iteration results in slightly better HMMs for the CI phones. However, too much iteration can lead to models which overfit to the training data.

- *Training of untied context dependent (CD) models*. This step is to train HMM model for each context dependent phoneme or triphone. Model parameters, including transition probability, component weight, mean and variance, are initialized by copying values from corresponding CI models. As in CI training, the Baum-Welch algorithm is iteratively used and the iteration is followed by a normalization operation to compute final model parameters.

- *Decision Tree building*. This step is to build phonetic decision trees and use them to decide which of the triphone HMM states are similar to each other, so that they can be tied together to train one jointly shared state, called senone, using the data from all these similar states. The optimal number of senones varies from application to application. It depends on the amount of available training data and the number of triphones present in the task.

- *Training of tied context dependent (CD) models*. This step is to train tied context dependent phoneme or senone models. Rather than use single Gaussian distribution per state to model output probability as in CI and CD (untied triphone) training, we use a mixture of Gaussian distributions for each senone in order to increase its modeling capability. The number of Gaussians in the mixture is pre-determined with respect to the amount of available training data. Moreover, the number should be set to the power of 2 (i.e. 2^N) for the convenience of Gaussian splitting. The training starts from the training of 1 Gaussian per state model, which is initialized by copying

parameters from corresponding triphone models. Baum-Welch algorithm is employed to perform an EM style iterative optimization. Once the training of single Gaussian model is done, the Gaussian distribution is split into two by perturbing its mean slightly, thus starting the iteration of 2-Gaussian model training. The procedure of iteration and splitting repeats until the training of pre-determined number of Gaussians is completed.

3.2 Speech Datasets

In this section, we briefly describe the speech databases used in this thesis. All of these datasets, even though some of them were collected by certain research group, are either open to the public for research use or can be obtained from Linguistic Data Consortium (LDC) at http://www.ldc.upenn.edu.

- *CMU Communicator dataset*. The corpus was collected using the CMU Communicator system, a telephone based dialog system that supports planning in a travel domain [Rudnicky et al, 1999]. The input speech signal was automatically detected and segmented by the speech recognition module of the dialog system. The segmented speech is stored to disk and then transcribed by human experts. The sampling rate is 8 KHz due to the constraint of telephone channel. The training set has 31,248 utterances, accounting for 30 hours speech, which were collected from the year of 1998 to 2000. The test set consists of 1,689 utterances, accounting for 0.5 hour speech, which were collected during a NIST evaluation conducted in July 2000. The average duration of these utterances is about 3 seconds. There are 9,769 words in the vocabulary. 39 dimension feature vector (12 MFCC, 12 delta-MFCC, 12 delta-delta-MFCC, 3 power) is extracted on frame basis for recognition.

- *ICSI meeting dataset*. The ICSI meeting dataset is a corpus of 75 meetings collected at ICSI, Berkeley during the years from 2000 to 2002 [Janin et al, 2003]. All the meetings are generally regular weekly meetings of different ICSI groups with different topics. The normal duration of each meeting is less than an hour, and in a total there is approximately 70 hour speech. The meetings were simultaneously recorded using close-talking microphones for each speaker, as well as a few of table microphones. All meetings were recorded at the same instrumented conference room. In the experiments of this thesis, the speech data is down-sampled to 11025 Hz and encoded as 16 bits per example. Frame rate is set to 105 per second. 39 dimension (12+12+12+3) MFCC feature vector is calculated for each frame. In our unsupervised acoustic model training

experiments, we use 10 meetings as the transcribed set for initial acoustic training, 61 meetings as the un-transcribed set for unsupervised training, 3 meetings as the hold-out set for recognizer tuning, and 1 meeting as the test set.

- *1997 Mandarin Broadcast News*. This set consists of 30 hours of recorded Mandarin broadcast news data. The data were collected from three sources: Voice of America (VOA), China Central Television (CCTV), and a commercial radio based in Los Angeles (KAZN-AM). The transcripts were created by native speakers of Mandarin and word segmentation (white-space between words) is included. The speech data is sampled as 16 KHz. 39 dimension MFCC feature is calculated for each frame.

- *TDT-4 Mandarin Broadcast News*. This corpus contains Chinese news data used in 2002 and 2003 TDT (Topic Detection and Tracking) technology evaluations. Data sources include Xinhua News (XIN), Zaobao News (ZBN), China Broadcasting System (CBS), China Television System (CTS), Voice of America (VOA), China National Radio (CNR) and China Central Television (CTV). The speech data is sampled as 16 KHz. The transcripts are mostly closed captions which contain a large number of transcription errors and thus can not be used directly for acoustic model training. In this thesis, a novel lightly supervised learning algorithm is proposed to address this problem and evaluated on this corpus. As before, our experiments also use 39 dimension MFCC feature as the representation of speech.

- *GALE BN03 Mandarin Dataset*. This is an un-transcribed speech corpus used in GALE (Global Autonomous Language Exploitation) project. The corpus includes the raw broadcast speech data collected in the year 2001 and 2003 from sources of China Central Television (CCTV), Voice of America (VOA) and New Tang Dynasty Television (NTDTV). The sampling rate is 16 KHz. Our unsupervised acoustic model training algorithm is evaluated on this corpus. 39 dimension MFCC feature is used in our experiments.

3.3 Summary

In this chapter, we reviewed the structure of CMU Sphinx III system. In particular, we described the 5-step acoustic model training procedure that Sphinx III supports. It consists of feature extraction, training of CI models, training of untied CD models, decision tree build & state tying, and training of tied CD models. In addition, we described the five speech corpora that we employ to evaluate the performance of our algorithms in the following chapters. The corpora include

CMU Communicator dataset, ICSI meeting dataset, 1997 Mandarin Broadcast News, TDT-4 Mandarin Broadcast News, and GALE BN03 Mandarin Dataset.

4 Ensemble Based Speech Recognition

This chapter presents a brief overview of ensemble based pattern classification and speech recognition approaches. We begin with a discussion of why ensembles of classifiers can outperform single classifier in addressing complicated classification problem. We then describe several extensively used learning algorithms for constructing ensembles. Next, we discuss the applications of ensemble based approaches in continuous speech recognition. We will cover some well known approaches including multi-band and multi-stream model, Boosting. ROVER, consensus network, and lattice combination.

4.1 Introduction

Ensemble methods are learning algorithms that construct a set of classifiers then combine their individual decisions in some fashion, in order to classify new examples. It has been shown that such a combination of "weak" classifiers can result in a "strong" composite classifier whose classification performance is much better than that of any single classifier. As one of the most active topics in many research areas, the ensemble approach is also known as the *Fusion of Models*, *Mixture of Experts*, *Committee of Learners*, *Multiple Classifier System*, *Consensus Theory*, as well as by other names.

The observation that ensembles perform better than any of its individual members is not an accidental phenomenon. To see why, imagine we have an ensemble of $2n+1$ classifiers, which have uncorrelated error patterns. Supposing e is the error rate of the worst classifier, the performance of ensemble after combination with majority voting would be no worse than $\sum_{k=n+1}^{2n+1} C_{2n+1}^{k} e^{k} (1-e)^{2n+1-k}$. For example, in the case of combination of three independent classifiers which error rates are equal to 20%, the overall error rate of the ensemble will be dramatically reduced to 10.4%.

Beyond this intuitive explanation, [Dietterich, 1998] gives a deeper analysis for why ensembles can improve performance, or why it is not possible to find a single classifier that works as well as an ensemble. [Dietterich, 1998] shows that the strength of ensemble lies on its competence and flexibility in dealing with the following three situations: the training data may not provide

sufficient information to choose a single best classifier; the learning algorithm we adopted may not be able to solve the difficult search problem we pose; and the hypothesis space may not contain the true function.

The example above also indicates the key issues to be considered in constructing a successful ensemble: the individual classifiers need be *accurate* and *diverse* [Kuncheva et al., 2002; Shipp & Kuncheva, 2002; Kuncheva & Whitaker, 2003]. A basic requirement for an individual classifier of ensemble is that its error rate should be at least better than random guessing. For binary classification, this means the error rate of any component of ensemble should be lower than 50%. However, this condition alone can not guarantee that the performance of ensemble is superior to its components. The overall performance of ensemble depends not only on the accuracy of each component, but also on how well different classifiers complement to each other. For example, a combination of identical classifiers will not provide any help to the classification even though they have high accuracy each alone. Research has shown that the error pattern of an individual classifier plays a critical role in ensemble based classification. For an ideal ensemble, we expect that individual classifiers are diverse so that the errors made by them can be uncorrelated and complementary. Obviously, this is not an easily-fulfilled requirement in the case of real-world applications such as continuous speech recognition. Investigations on measures of diversity and on the realization of diverse classifiers have been the focus of ensemble research.

4.2 Ensemble Learning

Many approaches for constructing ensemble have been developed. These approaches can be roughly categorized into three classes in accordance with the different objects being exploited in training procedure. In addition, there are some ad hoc methods specific to particular algorithms or applications. This section presents a brief review of some representative ensemble learning methods.

4.2.1 Manipulating the Training Data

This class of methods employs a strategy that generates multiple hypotheses by running the learning algorithm several times, each time with a different subset or distribution of training data.

The training data can be either randomly selected or selected according to an objective function. Well-known examples include *Bagging* and *Boosting*.

Bagging

Bagging is a straightforward way of manipulating training set for ensemble construction [Breiman 1996]. In each round of Bagging learning, a new training set is created using the technique called *Bootstrap Sampling*, which randomly draws a sample of *N* examples with replacement from the original training set. Note that some examples in the original set may not appear in a Bootstrap sample while others may appear more than once. A new single classifier is then learned from the sampled training set. Ensemble is constructed by running this procedure repeatedly, and the final hypothesis of the ensemble is determined by selecting the one best agreed on by individual classifiers. The algorithm is illustrated in Table 4.1, in which function $f(\mathbf{x}, y)$ can be interpreted as a classifier or recognition model that maps a feature/label pair to some confidence or probabilistic metrics such that $0 \le f(\mathbf{x}, y) \le 1$.

Input:

- Training set of *N* labeled examples $\Psi = \{(\mathbf{x}_i, y_i) \mid 1 \le i \le N\}$, where feature vector $\mathbf{x}_i = (x_{i1}, x_{i2}, ..., x_{iD}) \in \mathbf{R}^D$, a *D*-dimension space, and class label $y_i \in Y = \{c_1, c_2, ..., c_M\}$.
- A learning algorithm.
- An integer *K* specifying the number of individual classifiers in ensemble.

Training: For *k*=1 to *K*,

- Create a Bootstrap replicate Ψ_k with replacement from the original training set Ψ.
- Learn a new classifier $f_k(\mathbf{x}, y)$ on Ψ_k.

Generalization:

- The class label for a new example $\mathbf{x}$ is determined by majority voting:

$$y^* = \arg\max_{y \in Y} \sum_{k=1}^{K} f_k(\mathbf{x}, y)$$

Table 4.1 Bagging algorithm

Bagging is shown to be effective when used with "unstable" learning algorithms such as Decision Tree and Support Vector Machine, for which a small change of training data may lead to a large

change of learned concept. However, it may cause degradation when applied to "stable" learning algorithm such as K-Nearest Neighbor.

Input:

- Training set of N labeled examples $\Psi = \{(\mathbf{x}_i, y_i) \mid 1 \le i \le N\}$, where feature vector $\mathbf{x}_i = (x_{i1}, x_{i2}, ..., x_{iD}) \in \mathbf{R}^D$, a D-dimension space, and class label $y_i \in Y = \{c_1, c_2, ..., c_M\}$.
- A learning algorithm.
- An integer K specifying the number of individual classifiers in ensemble.

Initialization:

- Let $B = \{(i, y) \mid 1 \le i \le N, y \in Y \text{ and } y \ne y_i\}$.
- Initialize distribution of training data: $D_1(i, y) = 1/|B|$ for all $(i, y) \in B$.

Training: For *k*=1 to *K*,

- Train a new classifier $f_k(\mathbf{x}, y)$ with respect to distribution $D_k(i, y)$.
- Compute pseudo loss for classifier $f_k(\mathbf{x}, y)$:

$$\varepsilon_k = \tfrac{1}{2} \sum_{(i,y) \in B} D_k(i, y)(1 - f_k(\mathbf{x}_i, y_i) + f_k(\mathbf{x}_i, y)).$$

- Set $\beta_k = \varepsilon_k / (1 - \varepsilon_k)$.
- Compute importance factor α_k for $f_k(\mathbf{x}, y)$: $\alpha_k = -\log \beta_k$.
- Update distribution $D_k(i, y)$ by:

$$D_{k+1}(i, y) = \frac{D_k(i, y)}{Z_k} \beta_k^{\frac{1}{2}(1 + f_k(\mathbf{x}_i, y_i) - f_k(\mathbf{x}_i, y))}$$

where Z_k is a normalization factor chosen to make $D_{k+1}(i, y)$ a distribution function.

Generalization:

- The class label for a new example $\mathbf{x}$ is determined by weighted voting:

$$y^* = \arg\max_{y \in Y} \sum_{k=1}^{K} \alpha_k f_k(\mathbf{x}, y)$$

Table 4.2 AdaBoost algorithm for multi-class classification.

Boosting

Boosting has been, so far, the most successful approach developed for constructing ensembles [Freund and Schapire, 1996; Freund and Schapire, 1997; Schapire et al., 1997, Breiman, 1998; Mason et al., 1999; Schapire, 1999; Schapire and Singer, 1999; Collins et al., 2000]. In Boosting, single classifiers are iteratively trained in a fashion such that *hard-to-classify* examples are given increasing emphasis. In particular, the algorithm maintains a probability distribution for the training data, and initially every example is assigned equal weight. In each round, a new single classifier is learned from the current distribution. Meantime, a parameter that measures the classifier's importance is determined in respect of its classification accuracy. The single classifier is then used to classify every training example. The probability distribution is updated in such a way that the weight of an example will be enhanced if it is misclassified, or reduced otherwise. As a result those examples which are difficult to classify will be given more weights in the training of subsequent classifiers. The AdaBoost algorithm for multi-class classification is illustrated in Table 4.2 [Freund and Schapire, 1996; Schapire and Singer, 1999].

Boosting has shown advantages, both theoretically and practically, in handing complicated classification problems. It has been demonstrated that the training error of the Boosting algorithm drops exponentially fast to zero as the number of combined classifiers increases. More important, bounds of generalization error of Boosting algorithm have been formulated in terms of VC-dimension and *margin*, which suggests that Boosting is not sensitive to the problem of overfitting.

4.2.2 Manipulating the Input Features

In this class of methods, the ensemble is constructed by manipulating the input features. A representative example is Random Space in which the individual classifiers are constructed from subspaces formed by randomly selecting a number of dimensions from the original feature space [Ho, 1995; Ho, 1998; Skurichina and Duin, 2002]. For a given feature space of D dimensions, there are possible 2^D such subspaces available for selection. In each round of training, a selection is made to learn a new classifier e.g. Decision Tree. In the generalization stage, the final decision on a questioned instance is reached through majority voting of individual classifiers. Table 4.3 shows the algorithm.

When the number of training instances is relatively small, Random Space can have the capability of solving the data scarcity problem by constructing classifiers through random selection among

2^D subspaces. On the other hand, in a high dimensional feature space, the possibility that all dimensions being selected is very low. Moreover, the vast number of subspaces provides more choices than is needed in practice, which to some extent makes the use of all dimensionality unnecessary. Therefore, while other classification methods suffer from the *curse of dimensionality*, Random Space may be able to take advantage of high dimensionality.

Input:

- Training set of N labeled examples $\Psi = \{(\mathbf{x}_i, y_i) \mid 1 \le i \le N\}$, where feature vector $\mathbf{x}_i = (x_{i1}, x_{i2}, ..., x_{iD}) \in \mathbf{R}^D$, a D-dimension space, and class label $y_i \in Y = \{c_1, c_2, ..., c_M\}$.
- A learning algorithm.
- An integer K specifying the number of individual classifiers in ensemble.

Training: For *k*=1 to *K*,

- Randomly select a d_k -dimension subspace from original D-dimension feature space ($d_k \le D$), and construct a new training set $\Psi_k^{d_k} = \{(\tilde{\mathbf{x}}_i^{d_k}, y_i) \mid 1 \le i \le N\}$ where $\tilde{\mathbf{x}}_i^{d_k}$ is the projected vector of $\mathbf{x}_i$ in the d_k -dimension subspace.
- Learn a new classifier $f_k(\mathbf{x}^{d_k}, y)$ on $\Psi_k^{d_k}$.

Generalization:

- The class label for a new example $\mathbf{x}$ is determined by majority voting:

$$y^* = \arg\max_{y \in Y} \sum_{k=1}^{K} f_k(\tilde{\mathbf{x}}^{d_k}, y)$$

where $\tilde{\mathbf{x}}^{d_k}$ is the projected vector of $\mathbf{x}$ in subspaces.

Table 4.3 Algorithm of random space

4.2.3 Manipulating the Output Targets

The third class of methods for constructing ensemble is to manipulate the output target values of the classifier. Error-Correcting Output Coding is an example of this class of methods [Dietterich and Bakiri, 1995]. Suppose that the number of classes, M, is large. The new learning problem can be constructed by randomly partitioning the M classes into two subsets, A and B. The training data is then re-labeled so that the examples with the original classes in set A are given new label 0

while the other examples are given new label 1. Thus, a multi-class problem is converted into a binary class problem. The re-labeled data is feed to the learning algorithm for generating a new binary classifier, and consequently the ensemble is obtained by repeating the process many times. In generation stage, the outputs of each binary classifier give votes to the original classes. The class with the most number of votes is selected as the final prediction for the entire ensemble.

4.2.4 Other Approaches

Methods for generating ensembles of classifiers can be viewed as means of injecting randomness into different level of learning. For example, in Bagging, the randomization is performed in Bootstrap sampling, and in Random Space, the randomization is performed in the selection of subspace. Random Forests, investigated by [Dietterich, 2000; Breiman, 2001], shows that the randomness can also be injected into the learning procedure itself. Random Forests is a combination of tree-structured classifiers e.g. Decision Tree, each of which is grown by randomly selecting a feature-value test from the top-*n* best feature-value tests for node splitting. In contrast, classic tree-growing method always uses the best feature, measured by information gain, to split node. It has been shown that Random Forests can yield performance comparable to Boosting algorithm, and demonstrated, to some extent, robustness to noise.

In addition to the ensemble methods listed above, there are also algorithm-specific and application-specific methods. In the next section, we will discuss methods developed for speech recognition.

4.3 Ensemble Approaches in Speech Recognition

Research in ensemble and combination methods for speech recognition has a long history that can be traced back to 1980s [Stolfo et al, 1989]. A large number of innovative and effective approaches that utilize the characteristic of continuous speech have been developed since then. Recently, the ensemble methods advocated by machine learning research, such as Boosting and Bagging, have also aroused extensive research interests in speech recognition community. This section presents a brief review of some widely used methods for constructing and combining multiple recognition engines.

4.3.1 Construction of Ensemble based Speech Engines

Multi-Band and Multi-Stream Models

Narrow band noise, noise that occurs in a certain frequency range, is a common reason for performance degradation of ASR systems. This is because, in conventional "full-band" recognition, feature extraction is carried out over the whole frequency domain. Thus corruption of the speech signal caused by noise in any sub-band is spread to all the components of acoustic feature vector. In contrast, experiments on articulation index [Fletcher, 1953] have shown that human auditory perception is based on decisions within narrow frequency bands that are processed independently of each other. Humans are able to extract sufficient residual information from the clean frequency sub-bands, even with a considerable part of frequency domain being corrupted by noise.

Motivated by these observations, researchers have proposed and investigated Multi-Band models to enhance the robustness of ASR system in noise environment [Bourlard et al., 1996; Bourlard and Dupont, 1997; Tibrewala and Hermansky, 1997; Hagen et al, 1998; Dupont and Ris, 2001; Hagen et al., 2001; Hagen and Bourlard, 2001]. In Multi-Band recognition, the speech spectral domain is split into several frequency sub-bands, each of which is then processed separately to extract acoustic features. This is followed by estimation of frame level phoneme probabilities for each sub-band using corresponding sub-band features. These probabilities are then combined by some suitable rules, such as weighted sum or product, to yield a new vector representing the merged probability estimates which is further used in decoding. Figure 4.1 illustrates the architecture of Multi-Band systems [Sharma, 1999]. The advantage of Multi-Band recognition is that the noise from one sub-band can be isolated from other bands in feature extraction and phoneme probability estimation. In addition, the impact of narrow band noise on overall recognition performance can be further offset by deemphasizing its weight in combination.

Multi-Stream models are ensemble approaches which slightly differ from Multi-Band model in the features used in recognition. As compared to Multi-Band models in which each stream comprises features extracted from certain sub-band of speech spectrum, the streams of Multi-Stream model capture features from the entire frequency domain using different processing algorithms [Janin et al, 1999; Sharma, 1999; Christensen et al, 2000; Hagen et al, 2000; Neto and Meinedo, 2000; Shire, 2000]. More important, the Multi-Stream model provides a generalized

framework that allows for the use of various kinds of combination strategies for continuous speech recognition, such as the combination of different type of features, different information sources, and different probability estimator or even acoustic models [Dimitrakakis and Bengio, 2004].

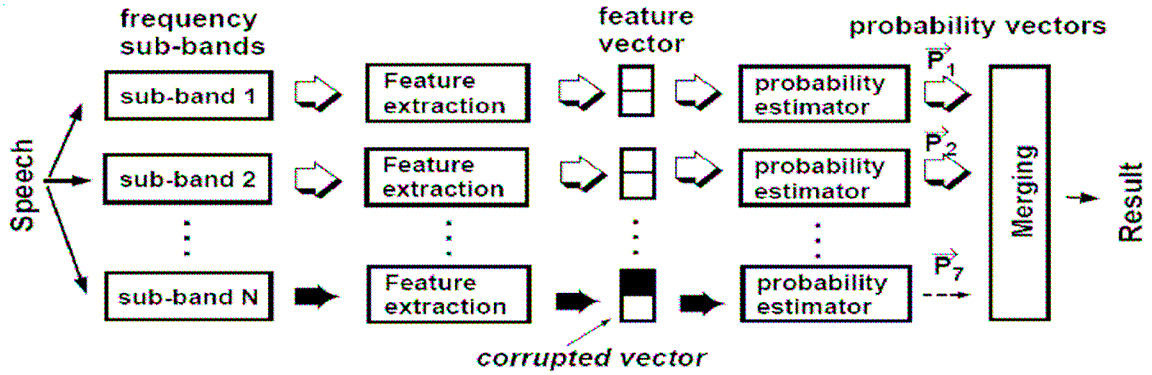

Figure 4.1 Multi-band model [Sharma, 1999]

Boosting for Acoustic Modeling

Boosting algorithm has been used to address a number of speech recognition problems and has been shown to be successful in improving system performance. Applications include speaker verification [Foo and Lim, 2002; Li et al, 2003; Asami etal, 2005], confidence annotation [Moreno et al, 2001], call routing [Rochery et al, 2002; Zitouni et al, 2002; Tur et al, 2004], speech detection [Xiong and Huang, 2002; Kwon and Lee, 2003], speech segmentation [Wang et al, 2003], emotion detection [Liscombe et al, 2005], intent classification [Tur, 2005], spoke language generation [Walker et al, 2003; Mairesse and Walker, 2005], etc.. However, the complexity of these applications does not exceed the level of standard multi-class classification.

The Boosting algorithm was also applied to LVCSR with respect to the characteristics of continuous speech. [Cook and Robinson, 1996; Cook et al, 1997; Schwenk, 1999] used Boosting algorithm to improve the performance of Hybrid HMM/Neural Network based speech recognizers. [Zweig and Padmanabhan, 2000; Meyer, 2002] developed practical utterance level Boosting training schemes that enable the technique to be used in large scale speech recognition task. [Zhang and Rudnicky, 2004a] extended Boosting training to the frame level, as an attempt to reduce word error rate instead of sentence error rate. [Dimitrakakis and Bengio, 2004; Dimitrakakis and Bengio, 2005] investigated Multi-Stream model as the platform to combine

acoustic models trained using Boosting algorithm. Substantial reductions of recognition error were achieved in these experiments.

It is instructive to compare Boosting with discriminative training methods, in order to better understand the effectiveness of Boosting in speech recognition. [Macherey et al, 2005] provides a unified framework for existing discriminative methods such as MMI, MCE, MPE and MWE. Given a training set of N labeled examples $\Psi = \{(\mathbf{x}_i, y_i) \mid 1 \le i \le N\}$, where feature vector $\mathbf{x}_i \in \mathbf{R}^D$ and class label $y_i \in Y = \{c_1, c_2, ..., c_M\}$ (in continuous speech recognition, Y denotes the hypothesis set), the unified objective function is defined as follows.

$$F = \frac{1}{N}\sum_{i=1}^{N} g\left(\left[\frac{\sum_{y \in Y} f^{\alpha}(\mathbf{x}_i; y) \cdot \theta(y, y_i)}{\sum_{y \in \tilde{Y}_i} f^{\alpha}(\mathbf{x}_i; y)}\right]^{1/\alpha}\right) \tag{4-1}$$

The choices of different form of smoothing function $g(z)$, concept function $f(\mathbf{x}; y)$, gain function $\theta(y, y_i)$ and hypothesis set $\tilde{Y}_i$ formulate particular discriminative training criteria [Macherey et al, 2005]. Table 4.4 lists some of the criteria commonly used in acoustic model training.

Criterion	$g(z)$	$f(\mathbf{x}; y)$	$\theta(y, y_i)$	$\tilde{Y}_i$
MMI	$\log(z)$	$P(\mathbf{x}, y)$	$\delta(y, y_i)$	Y
MCE	$-\frac{1}{1+z^{\beta}}$	$P(\mathbf{x}, y)$	$\delta(y, y_i)$	$Y - \{y_i\}$
MPE/MWE	z	$P(\mathbf{x}, y)$	$a(y, y_i)$	Y

Table 4.4 Discriminative training criteria formulated within a unified framework

Gain function $\theta(y, y_i)$ defines the similarity between two hypotheses. In the case of MMI and MCE training, $\theta(y, y_i)$ is expressed as the 0/1 function $\delta(y, y_i)$ that

$$\delta(y, y_i) = \begin{cases} 1 & y = y_i \\ 0 & otherwise \end{cases} \tag{4-2}$$

For MPE/MWE training, $\theta(y, y_i)$ is expressed as the accuracy function $a(y, y_i)$ which measures the phoneme or word accuracy of hypothesis y given transcripts y_i.

The objective function that Boosting algorithm aims to maximize is defined as follows [Schapire and Singer, 1999; Collins et al, 2000] (other variants exist).

$$F_{Boosting} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{y\in Y \text{ and } y\neq y_i}\exp[P(y \mid \mathbf{x}_i) - P(y_i \mid \mathbf{x}_i)\}] \tag{4-3}$$

(4-3) can also be formulated into the discriminative training framework given by (4-1) by setting $g(z) = -\frac{1}{z}$, $f(\mathbf{x}; y) = \exp(P(y \mid \mathbf{x}))$, $\theta(y, y_i) = \delta(y, y_i)$, and $\widetilde{Y}_i = Y - \{y_i\}$. Namely, Boosting is also a discriminative training method that the maximization of $F_{Boosting}$ via certain optimization procedures can increase the probability of the desired class y_i being predicted, and meantime decrease the probability of alternative classes $y \neq y_i$ being predicted. Therefore, Boosting owns the advantages of discriminative methods, such as having the capability to increase the *separability* between competing classes, and outperforming MLE (Maximum Likelihood Estimation) in the situation that model assumption isn't accurate.

On the other hand, Boosting differs from conventional discriminative training methods in the way that $f(\mathbf{x}, y)$ is generated. In discriminative training, $f(\mathbf{x}, y)$ is realized by learning and optimizing a single classification model. In contrast, Boosting training generates a set of models and combine their hypotheses together to make a better prediction of $f(\mathbf{x}, y)$. By utilizing appropriate combination techniques e.g. ROVER and consensus network, Boosting is shown to be more robust and flexible in handling complicated classification problems such as continuous speech recognition.

A preliminary experiment was conducted to compare Boosting with discriminative training. The speech corpus used in the experiment is the CMU Communicator corpus, an 8K-16bits telephone based speech dataset collected by CMU speech group. Please refer to Chapter 3 for the detail. Cambridge HTK-3.4 toolkit is used for discriminative training (http://htk.eng.cam.ac.uk). First, a baseline cross-word acoustic model is trained using MLE, and then optimized using MMI, MWE and MPE criterion respectively. The recommended "*approximate-error*" option is adopted for MPE. For Boosting, we trained a total of 5 acoustic models using the algorithm presented by [Zweig and Padmanabhan, 2000; Meyer, 2002]. The top-1 hypothesis output by each model was combined by using ROVER [Fiscus, 1997] to generate final hypothesis. For all the training criteria, the architecture of each acoustic model is set the same which is 2000/32 tied-state/Gaussians. Table 4.5 shows the recognition performance of MLE and discriminative training.

Training Criterion	Word Error Rate
MLE	15.84%
MMI	15.11%
MPE	14.88%
MWE	15.40%

Table 4.5 Performances of MLE training and discriminative training.

The word error rates of Boosting are illustrated in Table 4.6. The result for $N=n$ means that the error rate is calculated from the hypotheses generated by ROVER combination of n models.

# of Models	N=1	N=2	N=3	N=4	N=5
W. E. R.	15.84%	15.01%	14.48%	14.08%	13.87%

Table 4.6 Performances of Boosting training.

The experimental results demonstrate the effectiveness of Boosting algorithm in acoustic model training and continuous speech recognition. Boosting can be viewed as a discriminative training method but realized in the form of ensemble, so that it has the advantages of both. More detailed discussions and experiments of Boosting training are presented in Chapter 5.

Miscellaneous Methods

Other ensemble methods were also developed for improving recognition performance. [Cook and Robinson, 1995] built multiple speaker dependent acoustic models via utterance clustering, and combined them for speaker independent recognition. [Vergyri et al, 2000] investigated the feasibility of combining multilingual acoustic models to address the problem that sufficient training data is not available for a target language. [Wu et al, 1998; Hagen and Bourlard, 2000; Weber, 2000] incorporated multiple time scale information, e.g. phone-scale and syllable-scale information, into recognition by combining decoders with different time windows. More recently, [Siohan et al, 2005] used the technique of Random Forest to build multiple systems by injecting randomness into Decision Tree based state-tying procedure.

In practice, researchers have also proposed some simple but effective solutions. One well-known approach is to build acoustic models based on the separation of gender (male or female), age (children or adult), channel (telephone or cell phone), or dialect. For example, the CMU Communicator system adopted a parallel decoding architecture such that the speech recognition

module incorporates two independent decoders, one for male speakers and another for female speakers. Each decoder has its own acoustic model, but shares the same lexicon and language model. The two acoustic models are trained separately using speech data collected from male speakers or female speakers. In recognition, the "better" hypothesis that is predicted with the higher log likelihood score is selected as the final hypothesis.

4.3.2 Combination Approaches

The combination methods used in continuous speech recognition can be placed into three classes: feature combination, likelihood (or posterior) combination and hypothesis combination, in accordance with the recognition stage at which these methods are performed.

Feature Combination

Feature combination is a pre-processing method performed before the start of recognition. In feature combination, various acoustic features extracted from different information sources or calculated using different front-end algorithms are concatenated into a larger single vector as the feature for training and decoding. The mechanism of this method is illustrated in Figure 4.2. Strictly speaking, feature combination is not an ensemble based method since it essentially works on the platform of single recognition model.

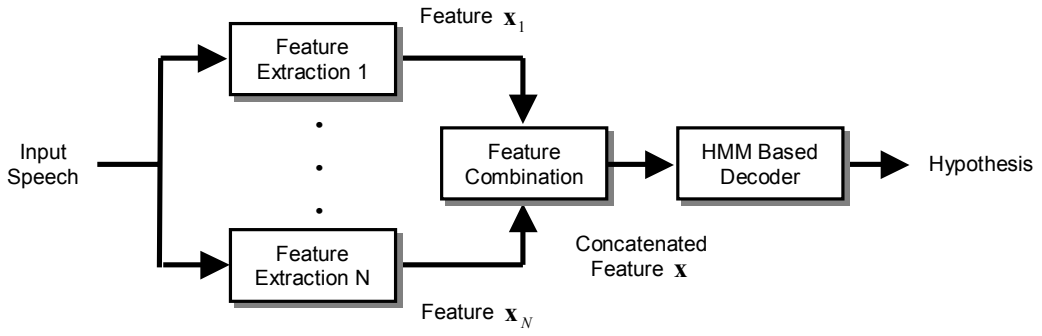

Figure 4.2 Feature combination

Likelihood / Posterior Combination

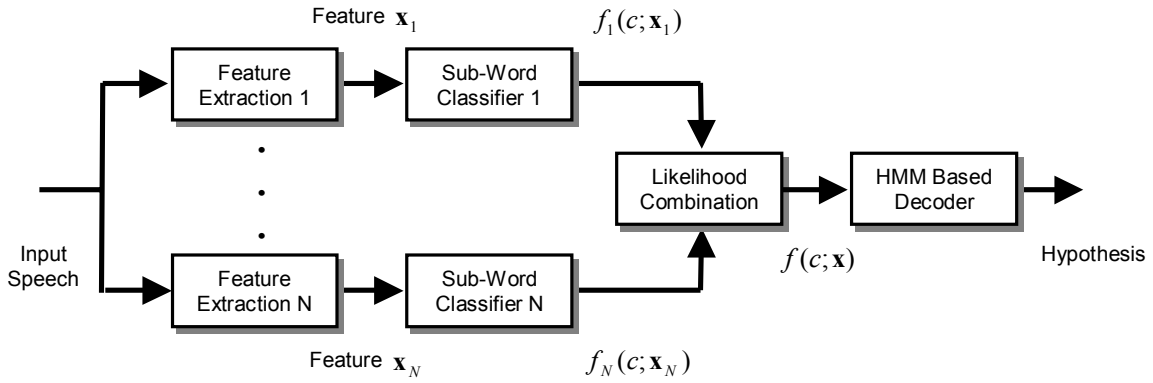

Figure 4.3 Likelihood/posterior combination

As shown in Figure 4.3, likelihood / posterior combination is performed within the recognition stage. In this class of methods, the probability observing a particular acoustic event, denoted by $f(c;\mathbf{x})$, is estimated by combining the output of individual classifiers. In implementation, $f(c;\mathbf{x})$ can be either a likelihood estimation $P(\mathbf{x}\,|\,c)$ or a posterior probability estimation $P(c\,|\,\mathbf{x})$. There are two issues to be considered in order to apply this kind of methods to continuous speech recognition. The first one is at what level the combination should be performed. Common choices include state and phoneme. For example, in Multi-Band acoustic modeling, spectral features extracted from frequency sub-bands are processed by multi-layer perceptron (MLP) based classifiers to generate phonetic probabilities. These phonetic probabilities are then combined frame by frame for subsequent classification [Sharma, 1999; Hagen et al, 2000]. The choice of a combination rule is another important issue which shows how to calculate $f(c;\mathbf{x})$ from $f_i(c;\mathbf{x}_i)$, the output of individual classifiers. Some commonly used rules are listed as follows [Kirchhoff and Bilmes, 2000; Kirchhoff et al, 2000; Zolnay et al, 2005]. In addition, more complicated combination techniques such as Neural Network can also be used for yielding better probabilistic estimate.

- *Product*

$$f(c;\mathbf{x}) = \frac{\prod_{i=1}^{N} f_i^{w_i}(c;\mathbf{x}_i)}{Z} \tag{4-4}$$

Where w_i denotes the weight associated with $f_i(c;\mathbf{x}_i)$, and Z is a normalization factor making $f(c;\mathbf{x})$ a probability density if necessary.

- *Sum*

$$f(c;\mathbf{x}) = \sum_{i=1}^{N} w_i f_i(c;\mathbf{x}_i) \tag{4-5}$$

- *Max*

$$f(c;\mathbf{x}) = \max_{i=1}^{N} f_i(c;\mathbf{x}_i) \tag{4-6}$$

Hypothesis Combination

The mechanism of hypothesis combination is illustrated in Figure 4.4. There are conflicting opinions with regard to the hypothesis combination performed when recognition has completed. On one hand, hypotheses at later stage of recognition are believed to be more robust because they carry wider temporal context. Thus, hypothesis combination can be more effective than early stage combination i.e. likelihood/posterior combination. However, this advantage is weakened from the observation that the correct hypothesis may have been pruned out before they reach the final state. Despite the debates, hypothesis combination is extensively used in speech recognition and demonstrates solid performance in many applications. Representative methods, ROVER, consensus network and word lattice combination, are discussed as follows.

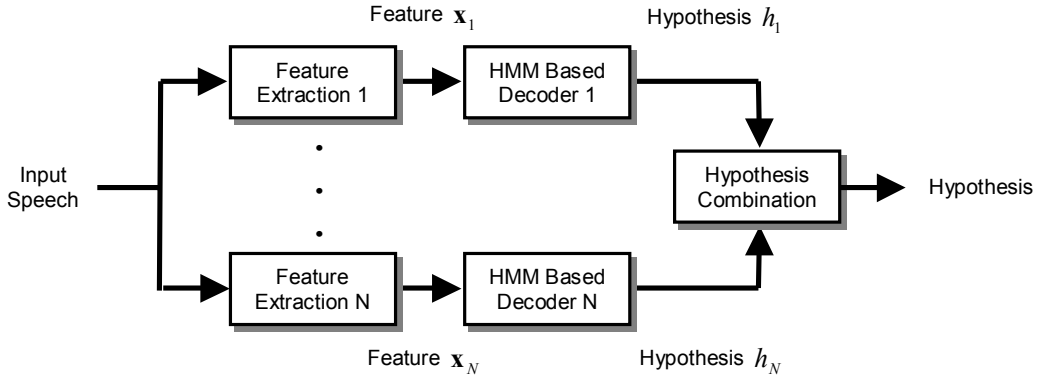

Figure 4.4 Hypothesis combination

- *ROVER*.

ROVER (Recognizer Output Voting Error Reduction) is a word-level combination approach developed at NIST. The goal of ROVER is to yield reduced word error rate by exploiting differences in the nature of the errors made by multiple speech recognition system [Fiscus, 1997;

Schwenk and Gauvain, 2000]. Rover proceeds in two stages. First, the hypotheses from different recognizers are progressively aligned together to build a single composite word transition network (WTN) by using dynamic programming. Once the network is generated, a voting scheme respecting frequency, word confidence and time information is adopted to select the words with highest number of votes as the new hypothesis.

- *Consensus Network*

The consensus network aligns links in a word lattice and transforms them into linear graph in which all paths pass through all nodes. The transformation is performed by a clustering procedure that groups time overlapped links into clusters based on their phonetic similarity, word probabilities and the precedence order of the links encoded in the original lattice. Given the alignment and the link posterior probabilities of a consensus network, the word sequence hypothesis with the lowest word error is obtained by selecting the word with the highest posterior probability at each node in the alignment [Mangu et al, 2000].

A theoretic analysis of the effectiveness of ROVER and consensus network was presented in [Goel and Byrne, 2000; Goel et al, 2000; Goel et al, 2004] with a unified framework of MBR (Minimum Bayes Risk) classification. ROVER and consensus network are viewed as efforts to generate hypothesis that minimizes expected word error rate rather than sentence error rate, where the word error criterion has a better match with the Levenshtein-distance based performance metric in speech recognition.

- *Word Lattice Based Combination.*

[Singh et al, 2001] investigated a different methodology for hypothesis combination. The idea is to merge the word hypotheses obtained from various recognition systems into a new word graph, and then search the best path from it as the final decoding output. Initially, each word in each of the hypotheses is represented by a node in the graph. The acoustic score of the node is set to that associated with the original word. In the next step, all nodes representing identical words hypothesized between the same time instants are collapsed into a single node. Finally, links are formed between all node pairs where the word-end time of one node and the word-begin time of the next node are within 30 ms of each other. After the word graph is constructed in this manner, a standard language model is used to score the paths through the graph and the best path is obtained as the final hypothesis.

Both ROVER and the Singh's technique are restricted to the use of a single best hypothesis of different recognizer. In many situations, the correct words do not appear in that single hypothesis.

Consequently, they are unable to be selected for combination. Motivated by the fact that word lattice can preserve more information than single hypothesis, [Li et al, 2002] extended the idea of [Singh et al, 2001] to the direct combination of word lattices. This scheme is also carried out in two stages. The word lattices generated from different ASR systems are first merged into a larger mixed lattice through operations of *merging edges*, *creating new edges* and *renormalizing scores*. Once this is done, searching algorithm, e.g. Viterbi or A*, is performed to seek the path with maximum cumulative score in the new mixed lattice as the combination result.

4.4 Summary

In this chapter, we briefly reviewed some well known ensemble based classification and combination approaches and their applications in continuous speech recognition. The approaches we have discussed include Boosting, Bagging, Random Space, Random Forests, Multi-Band and Multi-Stream Model, ROVER, consensus network, and Lattice combination. In next chapter, we will describe our efforts towards the improvement of Boosting training algorithm for acoustic modeling. In particular, we will present a novel frame level Boosting training algorithm and an improved hypotheses combination scheme, both of which demonstrate substantially better performance than conventional utterance level algorithms.

5 Frame Level Boosting Algorithms in Acoustic Model Training

As noted in the previous chapter, ensemble based classification and combination approaches have demonstrated promising performance on continuous speech recognition. This chapter presents our research results of Boosting based acoustic model training technologies. Specifically, we propose a novel frame level Boosting algorithm that enables acoustic model training more focus on misrecognized words within an utterance rather than, as adopted by conventional approaches, give the whole utterance an equal weight without discriminating recognition error from correct prediction.

Hypothesis combination technique is also investigated in this chapter for generating final system output with lower word error rate than the hypothesis of any individual speech engine. As mentioned in previous chapter, the object to be combined can be sentence hypothesis, consensus network, or even word lattice. On one hand, consensus network and word lattice appear to contain more information than single hypothesis. However, it is also shown the combination of consensus network or word lattice involves a vast amount of computationally expensive operations, e.g. compressing lattice to linear network and merging edges. The high cost lessens their attractiveness in handling ensembles of large sizes. Therefore, sentence hypothesis is selected in our experiment as the input of combination in order to achieve a computationally efficient combination. We propose an improved hypothesis combination scheme that uses Neural Nets to incorporate a number of features for generating more desirable combination results. In addition, the use of N-best list re-ranking as a means to identify better hypothesis is also investigated in this chapter.

This chapter is organized as follows. We begin with the discussion of utterance level Boosting training in Section 5.1 and show that the strength of Boosting algorithm is not simply due to the increase of the number of Gaussian distribution in acoustic model. The frame level Boosting training scheme aiming to reduce word/sub-word error rate rather than sentence error rate is described in Section 5.2. Section 5.3 discusses the experiments using N-best list re-ranking and Neural Nets to improve the performance of hypothesis Combination.

All the experiments make use of the CMU Sphinx III system for acoustic model training and testing. The speech corpus used in these experiments is the CMU Communicator corpus, an 8K-16bits telephone based speech dataset collected by CMU speech group. Please refer to Chapter 3

for the detail. Word error rate is adopted as the primary metric to evaluate recognition performance. In addition, we also report sentence error rate as the secondary metric in order to better understand experimental results.

5.1 Utterance Level Boosting Training of Acoustic Models

This section describes our investigation on the effectiveness of Boosting algorithm in acoustic model training.

Initialize:

- Let $\Psi_0 = \Psi$ where Ψ is the original training set.

For k = 1 to K:

- Train a new acoustic model Λ_k from data set Ψ_{k-1}.
- Generate hypothesis set $H_{\mathbf{x}} = \{h\}$ for each utterance $\mathbf{x} \in \Psi_{k-1}$ using Λ_k, and compute posterior probability $P_{\Lambda_k}(h \mid \mathbf{x})$ for each hypothesis $h \in H_{\mathbf{x}}$.
- Compute pseudo loss ε_k that

$$\varepsilon_k = \frac{1}{2 \mid \Psi_{k-1} \mid} \sum_{\mathbf{x} \in \Psi_{k-1}} \frac{1}{\mid H_{\mathbf{x}} \mid} \sum_{h \in H_{\mathbf{x}} \text{ and } h \neq y} [1 - P_{\Lambda_k}(y \mid \mathbf{x}) + P_{\Lambda_k}(h \mid \mathbf{x})]$$

  Where y denotes the correct transcripts for utterance $\mathbf{x}$.

- Set $c_k = \varepsilon_k / (1 - \varepsilon_k)$.
- Calculate new weight for each utterance $\mathbf{x} \in \Psi_{k-1}$ that

$$w(\mathbf{x}) = \sum_{h \in H_{\mathbf{x}} \text{ and } h \neq y} c_k^{\frac{1}{2}[1 + P_{\Lambda_k}(y \mid \mathbf{x}) - P_{\Lambda_k}(h \mid \mathbf{x})]}.$$

- Resample training data according to normalized $w(\mathbf{x})$, forming a new training set Ψ_k.

In generalization:

- The hypothesis to a new utterance $\mathbf{x}$ is determined by

$$h^* = \arg\max_h \sum_{k=1}^{K} \log \frac{1}{c_k} P_{\Lambda_k}(h \mid \mathbf{x}).$$

Table 5.1 Utterance level Boosting algorithm for acoustic modeling

5.1.1 Utterance Level Boosting Algorithm

The standard Boosting algorithm, as illustrated in Table 4.2, was initially designed for binary or multi-class classification. It does not consider the special requirements of continuous speech recognition. An utterance level training approach has been investigated by [Zweig and Padmanabhan, 2000; Meyer, 2002; Zhang and Rudnicky, 2003b] to address this problem.

Suppose we have a training set $\Psi = \{(\mathbf{x}_i, y_i) \mid 1 \le i \le N\}$ for continuous speech recognition, where, $\mathbf{x}_i$ is the sequence of feature vectors for the *i*-th training utterance in speech corpus, while y_i is the corresponding transcript. The essence of Boosting style acoustic model training is to minimize the value of the following loss function which is strongly related to sentence level recognition error.

$$L = \sum_{i=1}^{N} \sum_{h \neq y_i} \exp[P_\Lambda(h \mid \mathbf{x}_i) - P_\Lambda(y_i \mid \mathbf{x}_i)] \tag{5-1}$$

where h denotes possible sentence hypothesis for input speech, Λ denotes the recognition model or ensembles to be learned, and $P_\Lambda(h \mid \mathbf{x})$ denotes the posterior probability for hypothesis h. Table 5.1 illustrates the utterance level Boosting algorithm, which can be interpreted as a process to minimize the value of (5-1) by iteratively constructing an ensemble of *K* acoustic models.

A couple of obstacles need be tackled before we can apply this algorithm to acoustic modeling. First, for continuous speech recognition, the number of possible hypotheses for an utterance could be infinite. Assume that the recognizer has 5,000 words in its vocabulary, and that the maximum length for an utterance is restricted to 20 words. Theoretically, without taking into account the segmentation information associated with each word, the recognizer could output up to about 5000^{20} different hypotheses. Clearly, such a huge number makes it intractable for the recognizer to traverse all of the classes. To solve this problem, we compress the hypothesis space into a subset of limited size. In our experiments, the hypothesis set $H_{\mathbf{x}}$ contains only the hypotheses in the N-best list.

Second, the Boosting algorithm requires a probabilistic estimate of $P_\Lambda(h \mid \mathbf{x})$ for each class, while most speech recognizers only output the log-likelihood scores provided by the acoustic and

language models, whose value range is too large to be qualified as a good option for implementation. We therefore use the following scheme for converting the likelihood scores into posterior probability.

$$
\begin{aligned}
P_\Lambda(h \mid \mathbf{x}) &= \frac{P_\Lambda(h,\mathbf{x})}{P_\Lambda(\mathbf{x})} \approx \frac{P_\Lambda(h,\mathbf{x})^\beta}{\sum_{h' \in \text{N-best list of } \mathbf{x}} P_\Lambda(h',\mathbf{x})^\beta} \\
&= \frac{[P_\Lambda(h)P_\Lambda(\mathbf{x} \mid h)]^\beta}{\sum_{h' \in \text{N-best list of } \mathbf{x}} [P_\Lambda(h')P_\Lambda(\mathbf{x} \mid h')]^\beta} \\
&\approx \frac{\exp[\alpha \log P_\Lambda(h) + \log P_\Lambda(\mathbf{x} \mid h)]^\beta}{\sum_{h' \in \text{N-best list of } \mathbf{x}} \exp[\alpha \log P_\Lambda(h') + \log P_\Lambda(\mathbf{x} \mid h')]^\beta}
\end{aligned}
\tag{5-2}
$$

Where $\log P_\Lambda(\mathbf{x} \mid h)$ and $\log P_\Lambda(h')$ are acoustic model scores and language model scores, respectively. α is language model weight, and β is a smoothing parameter whose value is empirically set to control how the hypotheses in N-best list are weighted. In the case that the correct transcripts does not exist in the N-best list, one can run forced alignment to get the log-likelihood score, or simply choose a small default value for it.

5.1.2 Experiments of Utterance Level Boosting on CMU Communicator Dataset

The utterance level Boosting algorithm is evaluated using the CMU Communicator corpus. In our experiments, the language model was pre-trained and fixed during the iteration of Boosting training. Conventionally, Boosting based ensemble uses majority voting as the method to generate the final hypothesis for questioned utterances (please see Table 5.1 for the detail). Majority voting is essentially a sentence level selection scheme which only considers existing hypotheses for determining the most probable one. In comparison, ROVER is a word level combination scheme which can create a new hypothesis by using the information of word segmentation, frequency and confidence. We have shown that ROVER is much better than majority voting in handling sequence combination [Zhang and Rudnicky, 2004b]. Therefore, ROVER is adopted as the standard combination method in our experiments.

The goal of our first experiment is to determine the appropriate number of Gaussian distributions for acoustic modeling. The number of senones in our acoustic models is fixed at 2000. The number of Gaussian per senone being tested includes 8, 16, 32 and 64, which in turn correspond

to a total of 16K Gaussians, 32K Gaussians, 64K Gaussians and 128K Gaussians, respectively. Table 5.2 presents the recognition performance, measured by word error rate and sentence error rate, with different number of Gaussians. Experimental results show that the acoustic model with 64K Gaussians achieves the best performance. Therefore, we will use it as the standard decoding configuration, and its 14.90% word error rate will be viewed as the baseline for the following experiments.

# of Gaussian	16K	32K	64K	128K
W.E.R	17.31%	15.66%	14.90%	16.87%
S.E.R.	25.78%	22.41%	20.99%	22.18%

Table 5.2 Performance with different number of Gaussians.

The second experiment is to investigate the performance of utterance level Boosting algorithm in acoustic model training. The architecture of each acoustic model is set to 2000/32 senone/Gaussians as determined in the previous experiment. Table 5.3 presents the word error rate and sentence error rate as a function of N, the number of acoustic models in the ensemble, which value is up to 8. Please note that the result for $N=n$ means that the error rate is calculated from the hypotheses generated by ROVER combination of n models.

Size	N=1	N=2	N=3	N=4	N=5	N=6	N=7	N=8
W.E.R	14.90%	13.62%	12.88%	12.68%	12.39%	12.37%	12.43%	12.38%
S.E.R.	20.99%	19.99%	19.70%	19.51%	19.69%	19.69%	19.16%	19.10%

Table 5.3 Performance of utterance level Boosting algorithm on CMU Communicator dataset

Table 5.3 shows that Boosting algorithm demonstrates significant improvements over the baseline. Word error rate is down to 12.37% from 14.90% when 6 acoustic models are generated and combined, which represents a 17.0% relative reduction on the recognition errors.

The comparison between Table 5.2 and Table 5.3 suggests that the improvement of recognition performance of Boosting training isn't caused by simply increasing the number of Gaussians. In the single model recognition system, use of too many parameters will deteriorate the performance. For example, when the total number of Gaussians used in acoustic model increases to 128K from 64K, word error rate is up to 16.87% from 14.90%, which represents a 13.2% relative degradation. In contrast, such problem isn't observed in Boosting training. The lowest word error rate is

achieved by using 6 sets of acoustic model, which in a total account for 384K Gaussians. In addition, adding more models to the ensemble only results in a slight degradation of word error rate, e.g. 12.38% when *N*=8 vs. 12.37% when *N*=6.

Table 5.3 shows that the two metrics, word error rate and sentence error rate, can have considerably different results in measuring recognition performance. For example, when the number of acoustic models used in ensemble increases from 4 to 5, the word error rate is reduced by 0.29% while sentence error rate is up by 0.18%. Another example is that, when the number of acoustic models increases from 6 to 7, the sentence error rate is improved by 0.53% while the word error rate is deteriorated by 0.06%. This phenomenon is partly due to that ROVER is essentially a word level combination approach aiming to identify most probable word rather than select whole sentence. In addition, this also indicates that training methods based on loss function describing sentence errors may not necessarily lead to models with improved word error rate.

5.2 Frame Level Boosting Training

The experimental results reported in the previous section demonstrate the effectiveness of the Boosting algorithm in acoustic modeling. However, analysis shows that utterance level Boosting algorithm has two weaknesses that hurt its capability for handling continuous speech recognition. First, as illustrated in (5-1), the loss function of the utterance level Boosting algorithm is designed to describe sentence error, which is a sentence is judged as correct only when all the words within it are correct. It pays less attention to model word error, the most widely accepted metric to evaluate the performance of a speech recognizer. Specifically, the posterior probability $P_{\Lambda}(h \mid \mathbf{x})$ used in the loss function is defined to measure how likely that the sentence hypothesis h as a whole is correct given input speech $\mathbf{x}$. From $P_{\Lambda}(h \mid \mathbf{x})$ only, we are unable to tell how many words in the hypothesis are misrecognized. Moreover, due to the inaccuracy of model assumption, in many cases the hypothesis with higher posterior probability can contain more word level recognition errors than the one with lower posterior probability. Even though there is a strong correlation between sentence-level and word-level recognition errors, training with a criterion aiming to boost posterior probability for sentence may not necessarily result in optimal acoustic models with lower word error rate. Second, in the utterance level Boosting algorithm, re-sampling is performed on the whole utterance. This means all the words within the same utterance always have equal weights. However, intuitively, the misclassified words should be distinguished from

correctly recognized words and given more attention in the training of subsequent model. To address these two problems, we propose a frame level Boosting training scheme for acoustic modeling in which the loss function is constructed to measure frame based recognition errors, and re-sampling is applied to focus on the more confusable parts within an utterance, where the errors actually occur [Zhang and Rudnicky, 2004a].

5.2.1 Frame Level Posterior Probability

We define a metric, *frame level posterior probability*, to quantitatively measure how likely the hypothesis given to a particular frame by the existing acoustic model is correct. This metric enables us to focus on the *hard-to-learn* regions within the utterance and increase their weights in the model training.

Let $\mathbf{x}$ be the sequence of feature vectors for an utterance with T frames that $\mathbf{x} = (x_1, x_2, ..., x_T)$. For a particular frame x_t, we use $\theta_u(t)$ to denote the possible hypothesis on it, and $r_u(t)$ to denote the desired one, where the sub-index $\boldsymbol{u}$ denotes the unit chosen for the hypothesis. In continuous speech recognition, there are at least four choices for $\boldsymbol{u}$: ***s***entence, ***w***ord, ***p***honeme and st***a***te. For example, $\theta_w(t)$ and $r_w(t)$ represent the hypothesized and desired word at frame t, respectively. It bears noting that most speech corpora don't provide segmentation information at the word, phoneme and state levels. This means that, except for the $r_s(t)$ whose value could be obtained directly from transcripts, for all the other three units we need to perform an forced-alignment to determine the $r_u(t)$, and in these cases the $r_u(t)$ is only an approximation to the correct one.

The definition of frame level posterior probability of $\theta_u(t)$ being the hypothesis at frame t is given as follows.

$$P_\Lambda(\theta_u(t) \mid \mathbf{x}) = \frac{P_\Lambda(\theta_u(t), \mathbf{x})}{P_\Lambda(\mathbf{x})} \approx \frac{\sum_{h \in H_{\mathbf{x}},\ \theta_u(t) = \text{the hypothesized result of } h \text{ at frame } t} P_\Lambda(h, \mathbf{x})^\beta}{\sum_{h' \in H_{\mathbf{x}}} P_\Lambda(h', \mathbf{x})^\beta} \tag{5-3}$$

where β is a empirically determined smoothing factor, $H_{\mathbf{x}}$ denotes the search space of hypotheses for utterance $\mathbf{x}$, and h denotes a sentence hypothesis in $H_{\mathbf{x}}$. $H_{\mathbf{x}}$ is constrained to N-best lists in our experiments.

5.2.2 Loss Function of Frame Level Boosting

On the basis of frame level posterior probability, the loss function for the frame level Boosting training algorithm is defined as follows.

$$L = \sum_{i=1}^{N} \sum_{t=1}^{T_i} \sum_{\theta_u(i,t) \neq r_u(i,t)} \exp[P_\Lambda(\theta_u(i,t) \mid \mathbf{x}_i) - P_\Lambda(r_u(i,t) \mid \mathbf{x}_i)] \tag{5-4}$$

where T_i is the number of frames of utterance $\mathbf{x}_i$. $\theta_u(i,t)$ and $r_u(i,t)$ denote the hypothesized result and desired output for frame *t* of $\mathbf{x}_i$ respectively. In Eq. (5-4),

$$L_{i,t} = \sum_{\theta_u(i,t) \neq r_u(i,t)} \exp[P_\Lambda(\theta_u(i,t) \mid \mathbf{x}_i) - P_\Lambda(r_u(i,t) \mid \mathbf{x}_i)] \tag{5-5}$$

is called pseudo-loss which describes the degree of confusion at frame *t* for recognition. A high value of $L_{i,t}$ indicates that the questioned frame is possibly misrecognized and its weight needs be increased in the subsequent model training. Thus minimizing the value of this loss function will help to realize that $P_\Lambda(r_u(i,t) \mid \mathbf{x}_i) >> P_\Lambda(\theta_u(i,t) \mid \mathbf{x}_i)$ for $\theta_u(i,t) \neq r_u(i,t)$, and increase the accuracy of recognition.

Please note that the loss function (5-4) provides a generalized framework to measure different types of recognition errors by using different unit ***u***. For example, the loss function used by utterance level Boosting algorithm for modeling sentence error, as shown in Eq. (5-1), can be viewed as a special case of (5-4) by setting the unit ***u*** to sentence hypothesis. According to the definition, $\theta_s(i,t)$ is one of the sentence hypotheses in $H_{\mathbf{x}}$ of utterance $\mathbf{x}_i$, while the $r_s(i,t)$ is actually the correct transcripts y_i. Thus we have,

$$\begin{aligned} L &= \sum_{i=1}^{N} \sum_{t=1}^{T_i} \sum_{\theta_u(i,t) \neq r_u(i,t)} \exp[P_\Lambda(\theta_u(i,t) \mid \mathbf{x}_i) - P_\Lambda(r_u(i,t) \mid \mathbf{x}_i)] \\ &= \sum_{i=1}^{N} \sum_{t=1}^{T_i} \sum_{h \neq y_i} \exp[P_\Lambda(h \mid \mathbf{x}_i) - P_\Lambda(y_i \mid \mathbf{x}_i)] \\ &= \sum_{i=1}^{N} T_i \sum_{h \neq y_i} \exp[P_\Lambda(h \mid \mathbf{x}_i) - P_\Lambda(y_i \mid \mathbf{x}_i)] \end{aligned} \tag{5-6}$$

Comparing (5-6) to the loss function of utterance level Boosting algorithm as shown in (5-1), the only difference is that (5-6) considers the utterance length in terms of frames. Similarly, we can also set the unit ***u*** to word or phoneme, to construct loss functions as well as training schemes that aim to reduce word errors or phone errors for continuous speech recognition.

5.2.3 Frame level Boosting training scheme

Initialize:

- Let $\Psi_0 = \Psi$ where Ψ is the original training set.

For k = 1 to K:

- Train a new acoustic model Λ_k from data set Ψ_{k-1}.
- Determine the value of $r_u(i,t)$ for each frame of each utterance $\mathbf{x}_i \in \Psi_{k-1}$. Run forced-alignment if necessary.
- Generate hypothesis set $H_{\mathbf{x}_i} = \{h\}$ for each utterance $\mathbf{x}_i \in \Psi_{k-1}$ using Λ_k, and compute posterior probability $P_\Lambda(\theta_u(i,t) \mid \mathbf{x}_i)$ for every possible $\theta_u(i,t)$ at frame *t*.
- Compute pseudo loss

$$\varepsilon_k = \frac{1}{2 \mid \Psi_{k-1} \mid} \sum_{\mathbf{x}_i \in \Psi_{k-1}} \frac{1}{T_i} \sum_{t=1}^{T_i} \frac{l_{i,t}}{\mid \theta_u(i,t) \mid}$$

where

$$l_{i,t} = \sum_{\theta_u(i,t) \neq r_u(i,t)} [1 - P_{\Lambda_k}(r_u(i,t) \mid \mathbf{x}_i) + P_{\Lambda_k}(\theta_u(i,t) \mid \mathbf{x}_i)].$$

- Set $c_k = \varepsilon_k / (1 - \varepsilon_k)$.
- Calculate new weight for each frame *t* of each utterance $\mathbf{x}_i \in \Psi_{k-1}$ that

$$w_{i,t} = \frac{1}{\mid \theta_u(i,t) \mid} \sum_{\theta_u(i,t) \neq r_u(i,t)} c_k^{\frac{1}{2}[1 + P_{\Lambda_k}(r_u(i,t) \mid \mathbf{x}_i) - P_{\Lambda_k}(\theta_u(i,t) \mid \mathbf{x}_i)]}$$

- Resample training data according to normalized $w_{i,t}$, forming new training set Ψ_k.

In generalization:

- The hypothesis to a new utterance $\mathbf{x}$ is determined by using ROVER to combine the sentence hypotheses generated by the *K* acoustic models.

Table 5.4 Frame level Boosting algorithm for acoustic modeling

The frame level Boosting training scheme is presented in Table 5.4. Please note that in this algorithm the weight $w_{i,t}$ is associated with the frame t of utterance $\mathbf{x}_i$, rather than with the whole utterance. This leaves us with the question of how to resample the training data for acoustic model training. We use a simple but effective strategy to solve this problem, in the way that a new feature sequence is created for utterance $\mathbf{x}_i$ by duplicating frame $x_{i,t}$ for $\lfloor w_{i,t} \rfloor$ times. It should be pointed out that this method is rather ad hoc and that further investigation is necessary to improve this re-sampling method.

5.2.4 Experiment of Frame Level Boosting on CMU Communicator Dataset

To have fair comparison with conventional utterance level Boosting approach, the experiment of frame level Boosting algorithm is also conducted on CMU Communicator speech corpus. We use the same architecture for acoustic modeling. Each model has 2000 senones and the number of Gaussian per senone is set to 32. As in utterance level Boosting training, the number of acoustic models in ensemble is set to 8. We also use ROVER, the word level hypothesis combination method, to generate the final hypothesis for ensemble.

As discussed above, the loss function illustrated in Eq. (5-4) can be modified to measure different type of recognition errors by selecting different unit ***u***. For example, when ***u*** = sentence, the frame level Boosting algorithm is back-off to the conventional utterance level Boosting which has been investigated in previous section. Among the choices of ***u*** = word, ***u*** = phoneme and ***u*** = state, we use context independent phoneme (***u*** = phoneme) in our experiment. This is because the number of context independent phonemes is much less than that of words or states (senones), e.g. there are totally 55 phonemes in our English recognition system including 45 phonemes for modeling speech signal and 10 filler phonemes for modeling non-speech signal such as breath and noise. In comparison, the number of words and states are 9.5K and 2K, respectively. So the estimate of $P_\Lambda(\theta_u(t) \mid \mathbf{x})$ is more reliable when ***u*** = phoneme than use of other sub-word units. In the training, forced alignment is performed to obtain the most likely $r_u(t)$ (***u*** = phoneme) for each frame.

Table 5.5 presents the word error rate and sentence error rate of frame level Boosting algorithm varying with the ensemble size from 1 to 8. The baseline word error rate is 14.90%, realized by recognition with single acoustic model. Word error rate is reduced to 11.60% from the baseline,

when 6 acoustic models are generated and used. This represents a 22.1% relative reduction of the recognition errors and beats the 17.0% relative reduction achieved by conventional utterance level Boosting. Similar to utterance level Boosting, performance degradation is also observed for frame level Boosting. The word error rate increase slightly to 11.69% when using 8 acoustic models from 11.60% when using 6 acoustic models. However, compared to the result shown in Table 5.2, frame Boosting that utilizes multiple acoustic models is less sensitive to data sparseness problem than single model recognizer.

Size	*N*=1	*N*=2	*N*=3	*N*=4	*N*=5	*N*=6	*N*=7	*N*=8
W.E.R	14.90%	13.65%	12.68%	12.06%	11.62%	11.60%	11.65%	11.69%
S.E.R.	20.99%	20.11%	19.04%	18.51%	17.80%	17.80%	18.03%	17.80%

Table 5.5 Performance of frame level Boosting algorithm on CMU Communicator dataset

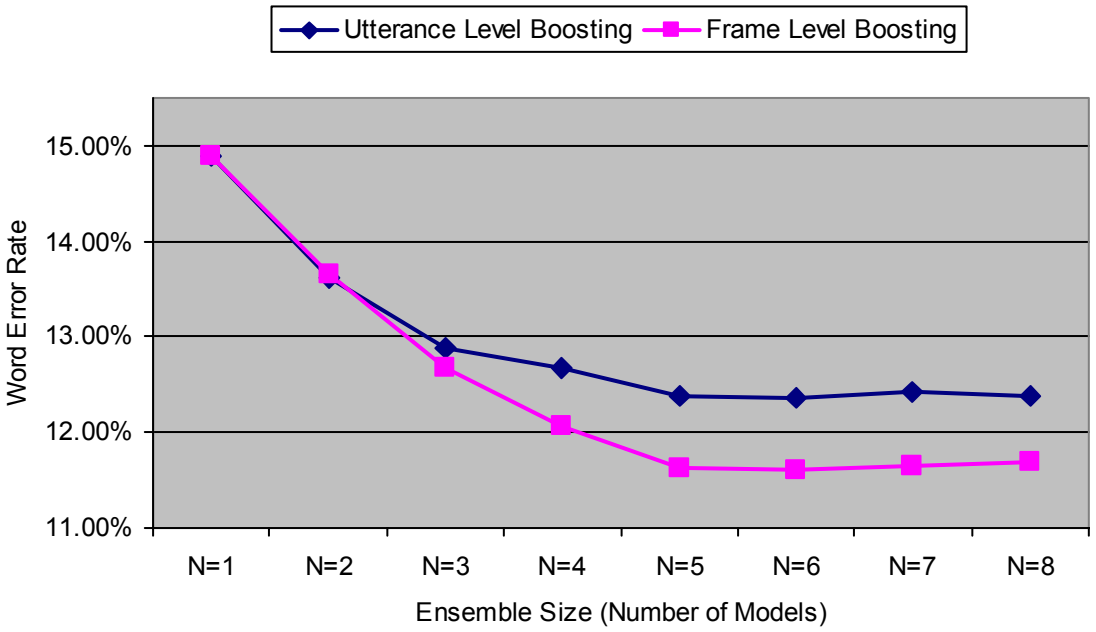

Figure 5.1 Comparisons of frame level Boosting algorithm and utterance level Boosting algorithm on CMU Communicator dataset

More detailed performance comparison between frame level and utterance level Boosting algorithms is given in Figure 5.1. The results show that except for *N*=1 and *N*=2, frame level Boosting consistently outperforms utterance level Boosting on the CMU Communicator dataset.

We also conducted a significance test, paired *t*-test, on performance differences between frame level and utterance level Boosting algorithms. The test is performed to the word error rates given in Table 5.5 and Table 5.3. The word error rate at N=1 isn't considered in calculation since it is the baseline result shared by both algorithms. We obtained that t= 4.42 and *P-value*=0.004. Therefore, we conclude that the proposed frame level Boosting algorithm improves the recognition performance significantly over utterance level Boosting algorithm.

5.3 An Improved Hypothesis Combination Scheme

The conventional utterance level Boosting training focuses on reducing sentence level error rate rather than reducing word level error rate, where the word level metric is most widely adopted in speech recognition. This weakness is partially addressed by the frame level Boosting algorithm. As described in the previous section, the loss function is modified to model word or sub-word e.g. phoneme level recognition errors. In this section, we approach this problem from a different perspective, by investigating post-processing techniques i.e. hypothesis combination to improve the performance of ensembles.

As discussed at the beginning of this chapter, we choose sentence hypothesis rather than consensus network or word lattice as the object of combination. In the research area of sentence combination, ROVER has been the *de facto* standard well studied for years. However, it still has ample room for improvement. We present a new combination scheme which improves standard hypothesis combination in two aspects: 1. we investigate N-best list re-ranking technique in order to select more likely hypothesis as the input of combination; 2. we propose a Neural Network based two-level scoring scheme for identifying correct hypothesized word. [Zhang and Rudnicky, 2004b; Zhang and Rudnicky, 2006b]

5.3.1 Applying N-Best List Re-Ranking to Hypothesis Combination

In our previous experiments, ROVER is performed to combine the top-1 hypothesis generated by each acoustic model. Examination of N-best list reveals that the best hypothesis, the one with the lowest word error rate, is not always in top-1 position. This phenomenon is caused by many reasons, such as inaccurate acoustic and language models, unavailability of sufficient training data, and lack of good features. N-best list re-ranking is a post-processing technique that attempts

to locate the hypothesis with lowest word errors rather than to accept the top-1 result blindly, and its effectiveness has been proved by many independent experiments. In the following experiment, the original top-1 hypothesis is replaced by the one generated by N-best list re-ranking as the object for combination.

The application of N-best re-ranking involves two aspects: the identification of useful features and the selection of an effective re-ranking technique. According to our experience in many classification tasks, good features usually play a more important role in creating a successful system. Five features, including information from different sources, are investigated in our experiments to measure the reliability of a questioned sentence hypothesis h:

- ***LM-Backoff-Mode***. This is a language model related feature. For each word, the value of backoff mode is determined according to whether the 1, 2, or 3-gram is used to compute language model score. For a sentence hypothesis, the feature value is set to the average value of every word.

- ***Utterance level posterior probability*** $P_\lambda(h \mid \mathbf{x})$. This is a feature measuring the correctness of the whole sentence hypothesis. One implementation scheme has been discussed in Section 5.1.1.

- ***Word level posterior probability*** $P_\lambda(w \mid \mathbf{x})$. w denotes a word in hypothesis. This feature measures how likely a particular hypothesized word is a correct recognition result. The value is computed from the word lattice or the N-best list by summing and normalizing the scores of paths passing through the word in question [Wessel et al, 1998]. The feature value for a hypothesis is set to the average value of words.

- ***Frame level posterior probability*** $P_\lambda(\theta_u(t) \mid \mathbf{x})$. This feature is to measure how likely the hypothesis given to a particular frame by the existing acoustic model is correct. Section 5.2.1 provides an implementation scheme for this feature. For a sentence hypothesis in the N-best list, its feature value is computed by summing and normalizing $P_\lambda(\theta_u(t) \mid \mathbf{x})$ over frames. The calculation is as follows.

$$f = \frac{1}{T}\sum_{t=1}^{T} P_\lambda(\theta_u(t) = r_u(t) \mid \mathbf{x}) \tag{5-7}$$

where T denotes the number of frames in the utterance, and $\theta_u(t)$ and $r_u(t)$ denote the predicted hypothesis and desired hypothesis at frame t, respectively. (5-7) can be

interpreted as an approximation to the frame based word or sub-word accuracy. In our experiment, the unit u is set to isolated phoneme, and forced alignment is used to estimate $r_u(t)$.

- ***WTN based word vote***. The entire N-best list is first converted into a Word Transition Network (WTN) through word-to-word alignment. For each word, we then calculate the number of occurrences (votes) in the WTN. The feature value for a sentence hypothesis is obtained by averaging the vote over every word appearing in that hypothesis.

In our experiment, we use Neural Network as the re-ranking method. The inputs are the five features, and the output is trained to approximate the word accuracy of each sentence hypothesis. The hidden layer of the Neural Network is set to contain 20 nodes. After re-ranking, the hypothesis with the highest estimated word accuracy will be chosen as the best hypothesis for ROVER combination.

Size	*N*=1	*N*=2	*N*=3	*N*=4	*N*=5	*N*=6	*N*=7	*N*=8
W.E.R	14.39%	13.45%	12.28%	11.99%	11.47%	11.34%	11.40%	11.51%
S.E.R.	20.46%	19.87%	18.81%	18.21%	17.56%	17.56%	17.50%	17.45%

Table 5.6 Performance of frame level Boosting training + N-best list re-ranking on CMU Communicator dataset

The experiments are conducted on CMU Communicator dataset using Sphinx III system. Frame level Boosting algorithm, described in previous section, is employed to train a total of 8 acoustic models. Each testing utterance is decoded using these 8 models separately to generate N-best lists. We then perform N-best list re-ranking to select more likely hypothesis for ROVER combination instead of accepting top-1 list directly. The experimental results of using top-1 hypotheses have been reported in Table 5.5 of Section 5.2.4. Table 5.6 presents the combination results on the basis of N-best list re-ranking. *N*=*n* means that the hypotheses of n acoustic models are used in ROVER combination.

Comparison between Table 5.6 and 5.5 shows that the Neural Nets based re-ranking achieves consistent improvement of recognition performance. For example, at the baseline of using single acoustic model (N = 1) for decoding, the word error rate realized by N-best list re-ranking is 14.39%, 0.5% better than that without re-ranking. Moreover, the word error rate further decreases to 11.34% when combining 6 models, compared to the 11.60% without re-ranking. We conducted significance test, paired *t*-test, on performance differences between using and not using N-best list

re-ranking. The test is performed to the word error rates given in Table 5.6 and Table 5.5. We obtained that t=5.05 and *P-value*=0.001, which indicates the probability that the better performance of N-best list re-ranking is obtained by chance is very small. The results manifest the effectiveness of the Neural Nets based re-ranking method, as well as that of the features we adopted in this experiment. A graphical performance comparison of ROVER combination using and not using N-best list re-ranking is presented in Figure 5.2.

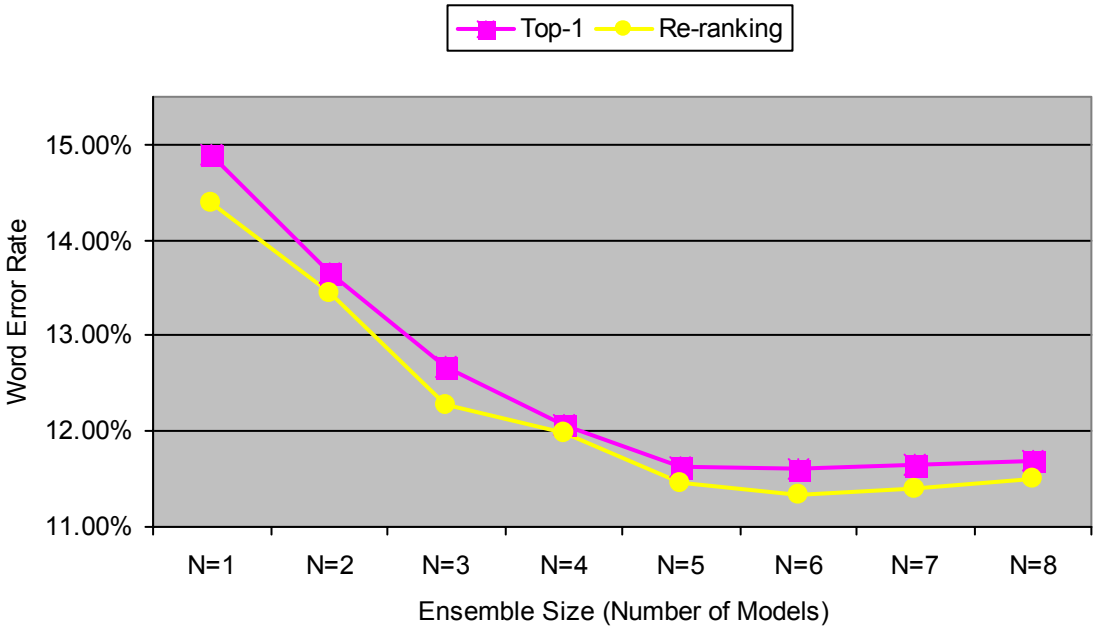

Figure 5.2 Performances of ROVER combination with and without N-best List re-ranking

5.3.2 Neural Network Based Scoring Scheme

ROVER is a word-level combination approach developed at NIST that aims to yield reduced word error rate by exploiting differences in the nature of the errors made by multiple speech recognition system [Fiscus, 1997]. ROVER proceeds in two stages. In the first stage, the best word hypotheses produced by different recognizers are progressively aligned together to build a single composite Word Transition Network (WTN) by using dynamic programming. Once the WTN is generated, each node in the network is then evaluated by a voting scheme to select the best word as the final recognition result.

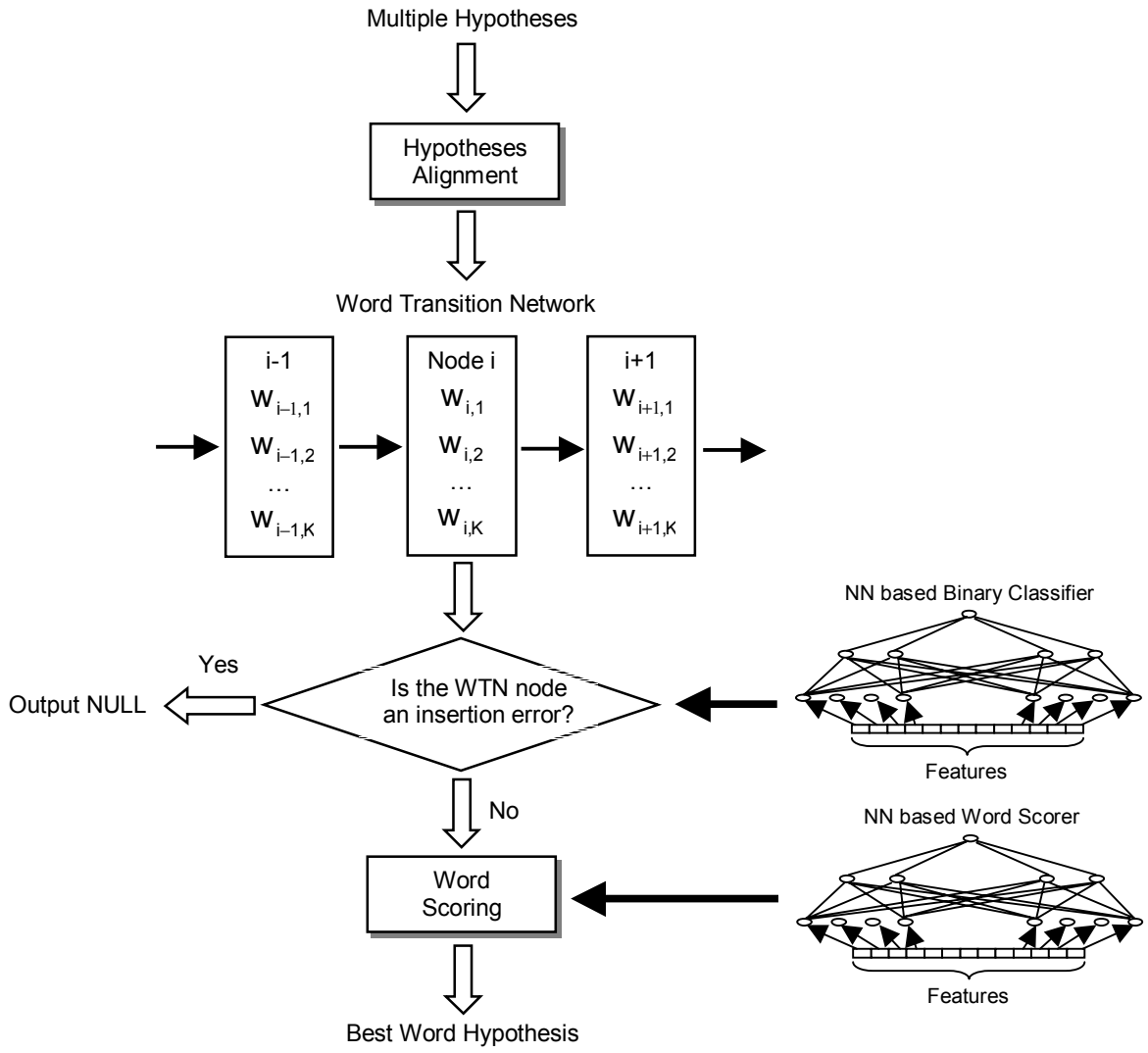

Figure 5.3 Neural Networks based two-stage scoring scheme

The original version of ROVER uses a simple voting strategy that linearly combines two types of information in word selection: frequency of occurrence and confidence score. The general scoring scheme is formulated as follows.

$$Score(word) = \beta * N(word) + (1-\beta) * C(word) \tag{5-8}$$

where $N(word)$ denotes the normalized frequency of occurrence, $C(word)$ denotes the average or maximum confidence score, and β is a parameter tuned to balance $N(.)$ and $C(.)$.

Our preliminary research showed that in many cases, the correct hypothesized words cannot be found due to the simplicity of this strategy. To address this issue, we propose a two-stage scoring scheme in which the detection of insertion error is separated from word scoring [Zhang and

Rudnicky, 2006b]. Specifically, once the WTN is constructed, we first use a binary classifier to determine if the WTN node in question is an insertion error. If not, each word in the node will be scored on the basis of a variety of features extracted from multiple information sources. The two-stage scheme is plotted in Figure 5.3. As shown in Figure 5.3, both the insertion detection and word scoring are implemented by using Neural Network.

The first task is to train a Neural Network based binary classifier to determine if a WTN node is an insertion error. For each node, there are five features designed to fulfill this task. Please refer to previous sections for the features which implementations are not given here .

- ***Average frequency of occurrence for real words.***
- ***Average frequency of occurrence for filler words and null arcs.***
- ***Average word level posterior probability for real words.***
- ***Average word level posterior probability for filler words and null arcs.*** A default value is set to null transition arcs since they do not have word probabilities.
- ***Entropy.*** This feature is designed to measure the degree of confusion within a WTN node. The feature value is computed from the normalized frequency of occurrence of each word in the node.

To train the Neural Network, the class label of each WTN node is set to either 1 or 0, representing whether it is an insertion error or not. The value can be manually transcribed or obtained by performing an alignment with references as in our experiments. The Back-propagation algorithm is used in our experiments as the training method. The hidden layer of the Neural Network contains 20 nodes.

The next task is to train another Neural Network to determine the most likely word for each WTN node not classified as insertion error. For each word to be evaluated, its input to this Neural Network consists of seven features.

- ***Frequency of occurrence.***
- ***LM Back-off Mode.***
- ***Contextual LM Back-off Mode.*** The average *LM Back-off Mode* over the left and right neighbors of the questioned word.
- ***Utterance level posterior probability.***

- ***Word level posterior probability.***

- ***Frame level posterior probability.*** This feature originally measures the probability of a word occurring at a given frame [Zhang and Rudnicky, 2004a]. For a word in the WTN node, the feature value is computed by averaging frame probability, across all the frames that the word spans.

- ***Recognizer's word accuracy.*** The word accuracy of the recognizer that generates the questioned word. The value is computed on the training set.

The neural network is trained in a discriminative way to minimize the following objective function.

$$L = \sum_{i=1}^{I} \sum_{w \neq d} \exp(s(\mathbf{x}_{i,w}) - s(\mathbf{x}_{i,d})) \tag{5-9}$$

where $s(.)$ denotes the scoring function defined by the Neural Network, $\mathbf{x}_{i,d}$ denotes the input feature vector of the desired word d in WTN node i, $\mathbf{x}_{i,w}$ denotes the feature vector of competing word w in the same node, and I is the number of WTN nodes participating the training. The desired word of each WTN node is determined by aligning WTN with references. Neural Network scorer has one hidden layer containing 30 nodes, and we use gradient descent as the learning approach to optimize its parameters.

The experiment evaluating the new scoring scheme is carried out on the basis of frame level Boosting and N-best list re-ranking. As before, a total of 8 acoustic models are trained using frame level Boosting algorithm described in Section 5.2. For a testing utterance, the 8 models are used respectively to decode the utterance and generate N-best lists as the recognition results. N-best list re-ranking, the post-processing technique described in Section 5.3.1, is performed to select more likely hypothesis for combination. The selected hypotheses are aligned together to build a Word Transition Network (WTN). The proposed scoring scheme is then applied to each WTN node to determine the most desirable hypothesized word. Table 5.7 presents the experimental results of using this new two-stage scoring scheme.

Size	N=1	N=2	N=3	N=4	N=5	N=6	N=7	N=8
W.E.R	14.39%	13.22%	12.17%	11.78%	11.31%	11.12%	11.23%	11.36%
S.E.R.	20.46%	19.57%	18.81%	18.21%	17.56%	17.45%	17.28%	17.41%

Table 5.7 Performance of Neural Network based scoring scheme on CMU Communicator dataset

Figure 5.4 compares the performance of new scoring scheme with that of standard ROVER. Please see Section 5.3.1 for more detailed result of standard ROVER combination based on N-best list re-ranking. As shown in Figure 5.4, the new scoring scheme achieves a consistent gain over standard ROVER. Paired *t*-test test is conducted to further measure the performance difference between the new scoring scheme and standard ROVER. The test is performed to the word error rates given in Table 5.7 and Table 5.6. The word error rate at N=1 isn't considered in calculation since it is shared by both methods. We obtained that t=10.9 and *P-value*=0.001. Therefore, we conclude that the new scheme significantly outperforms ROVER in improving recognition accuracy.

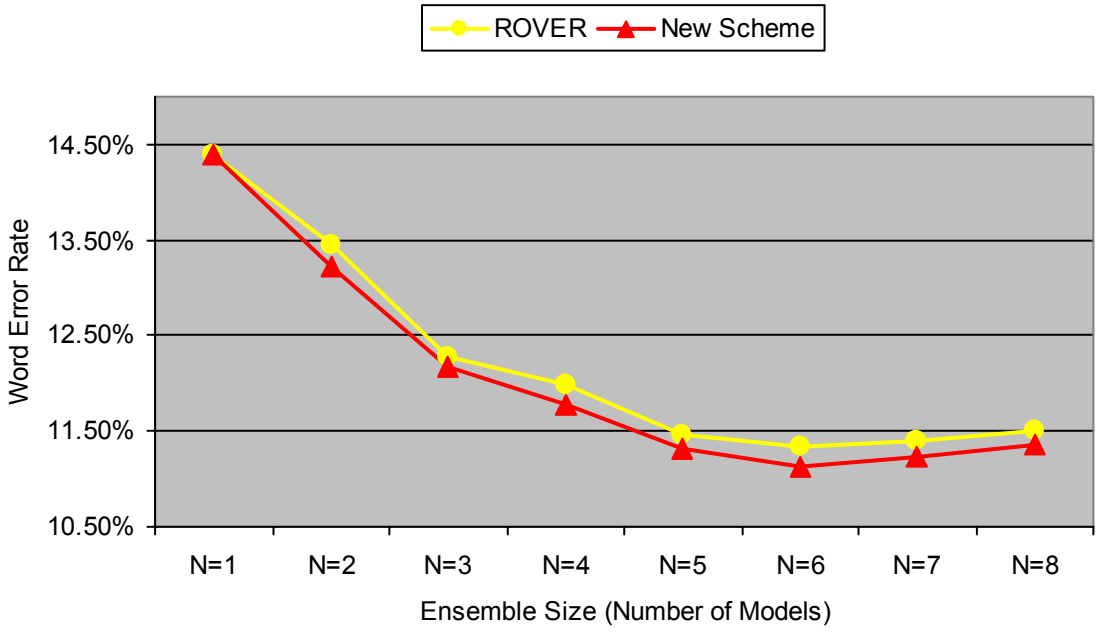

Figure 5.4 Performances of Neural Network based scoring scheme and standard ROVER combination

5.4 Improving Estimation of Word/Sub-word Boundary

In the frame level Boosting training algorithm presented in Table 5.4, the sub-word hypothesis for each frame, $r_u(t)$, is estimated by forced alignment on the basis of current acoustic model. We noticed that the performance of Boosting training is highly related to the result of forced

alignment. Therefore, investigation on a robust forced alignment method is desirable. In the following experiment, we use an ensemble based voting scheme to determine the boundary of word or sub-word. Suppose a total of K acoustic models, $\Lambda = \{\Lambda_1, \Lambda_2, ..., \Lambda_K\}$, have been trained. Each acoustic model is called to perform forced alignment separately for each training utterance. Specifically, for a word or sub-word e.g. phoneme q in the transcripts, each model $\Lambda_k (1 \le k \le K)$ generates its own estimate of the starting frame s_k and ending frame e_k. The boundary predicted by entire ensemble is set to the mean of s_k and e_k:

$$s = \frac{1}{K} \sum_{i=1}^{K} s_k \tag{5-10}$$

$$e = \frac{1}{K} \sum_{i=1}^{K} e_k \tag{5-11}$$

The ensemble based boundary estimation is implemented into frame level Boosting training algorithm and tested with CMU Communicator dataset. We use the same architecture for acoustic modeling. Each model has 2000 senones and each senone has 32 Gaussians. The maximum number of acoustic models in ensemble is also set to 8. Context independent phoneme is adopted as hypothesis unit ***u***. We also use ROVER to combine individual hypotheses generated by each acoustic model. Table 5.8 presents the recognition error rates. Comparison between Table 5.8 and Table 5.5 shows that ensemble based boundary estimation performs modestly better than single model based forced alignment (except N=6). The result is encouraging since the ensemble based estimation is only a simple voting strategy. We believe further improvement can be obtained by investigating and utilizing effective segmentation methods.

Size	N=1	N=2	N=3	N=4	N=5	N=6	N=7	N=8
W.E.R	14.90%	13.65%	12.41%	11.95%	11.60%	11.67%	11.43%	11.62%
S.E.R.	20.99%	20.11%	19.16%	18.57%	18.27%	18.09%	17.74%	17.92%

Table 5.8 Performance of frame level Boosting algorithm using ensemble based sub-word boundary estimation.

5.5 Summary

This chapter summaries our research results on ensemble based acoustic model training and speech recognition.

We presented a novel frame level Boosting training algorithm which demonstrates more advantages than conventional utterance level algorithm. We also presented a new combination scheme that improves hypothesis combination from two perspectives: N-best re-ranking technique is used to provide better candidates for combination, and Neural Network is used to incorporate a number of features for generating more desirable recognition results. Our proposed approaches were evaluated on CMU Communicator dataset. Figure 5.5 plots the word error rates of utterance level Boosting algorithm, frame level Boosting algorithm, and improved hypothesis combination scheme. Some important observations obtained in our experiments are given as follows.

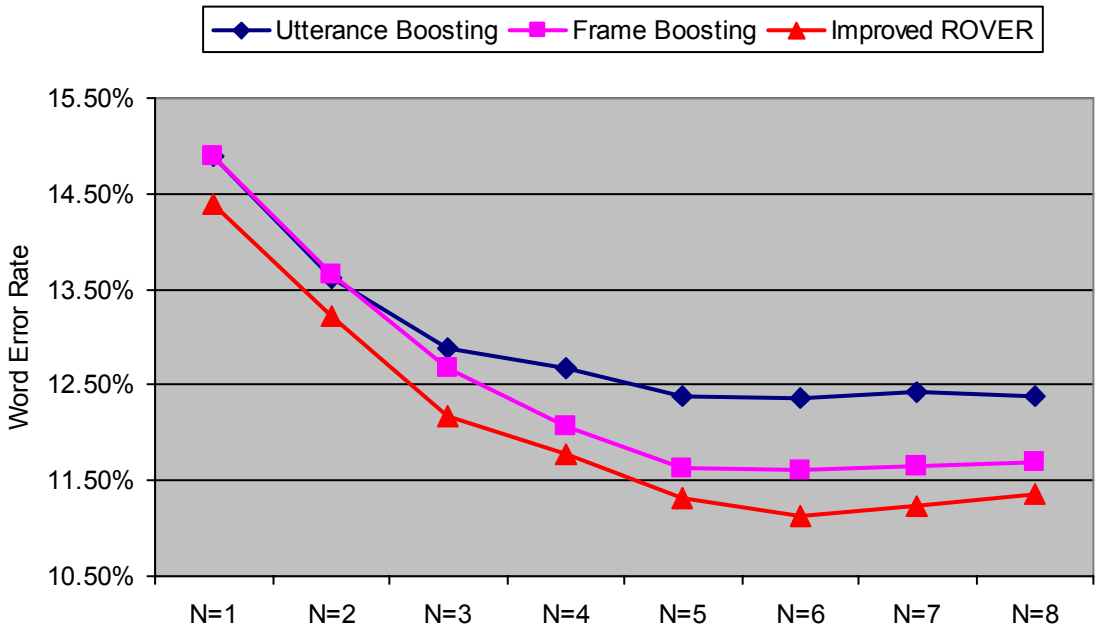

Figure 5.5 Experimental results of utterance level Boosting training, frame level Boosting training, and improved hypothesis combination using N-best list re-ranking and Neural Nets based scoring.

- Significant improvements of recognition performance are achieved by using Boosting based acoustic model training. For CMU Communicator dataset, the word error rate is reduced to 12.37%, by using utterance level Boosting training, from the baseline of 14.90%, the error rate of traditional recognition system using single acoustic model. Relatively, utterance level Boosting reduce recognition errors by 17.0%.
- Boosting based acoustic model training is shown to be more robust to data sparseness problem. When using single acoustic model, the increase of model size, i.e. number of

Gaussians, can deteriorate recognition performance drastically. In comparison, adding more models to Boosting based recognition only results in a slight performance degradation.

- Frame level Boosting algorithm has more advantages than utterance level Boosting algorithm in generating discriminative acoustic models, and consequently, outperforms the latter with a big margin. On the basis of frame level Boosting training, word error rate is reduced to 11.60% which beats the performance of utterance level Boosting training by relatively 6%.
- In addition, frame level Boosting algorithm provides a generalized framework which supports acoustic model training to minimize different types of recognition errors, e.g. word or sub-word error. Conventional utterance level Boosting can be viewed as a special case of frame level Boosting.
- Improving combination techniques is manifested to be an effective way to reduce recognition errors. In our experiments, the integration of N-best list re-ranking and Neural Network based word scoring further reduce recognition errors to 11.12%, from 11.60% which is the word error rate achieved by standard ROVER combination.
- Overall, by using all the techniques described in this chapter together, word error rate is reduced to 11.12% from 14.90%, representing a relative 25.4% improvement of recognition performance.

6 Acoustic Model Training Using Un-transcribed and Closed Captioned Speech Data

The lack of carefully prepared speech data has been a persistent problem for statistical acoustic model training for which the sufficiency of training data is a basic requirement for pursuing good performance. To address this problem, researchers started to explore and exploit easily-obtained data sources such as un-transcribed and closed-captioned speech, and investigate methods combining all the available data for acoustic model training [Blum et al, 1998; Jang and Hauptmann, 1999a; Kemp and Waibel, 1999; Lamel et al, 2000; Nigam et al, 2000; Kamm and Meyer, 2001; Wessel and Ney, 2001; Lamel et al, 2002; Cozman et al, 2003; Moreno et al, 2003; Chen et al, 2004; Nguyen and Xiang, 2004; Visweswariah et al, 2004; Zhang et al, 2005; Ma and Matsoukas, 2007]. This chapter presents a review to the related research carried out so far.

We begin this chapter with the discussion of the advantages to use un-transcribed and closed captioned speech data for acoustic model training, followed by the discussion of the difficulty to use them. We then describe confidence scoring based data selection approach and its applications in unsupervised and lightly supervised training. Finally, we discuss the drawbacks of the confidence scoring based data selection approach, showing that this approach can deteriorate classification performance rather than improve it.

6.1 Necessity to Use Un-Transcribed and Closed-Captioned Speech Data

HMM and Gaussian Mixture based statistical learning approaches are the mainstream acoustic modeling technologies for state-of-the-art speech recognition systems. To have an accurate estimate of model parameters, sufficient well transcribed training data is highly desired. (Here we only concentrate on the effect of the training data, while other issues that also have impacts on acoustic model training, e.g. the environment noise and speaker variation, are beyond the research scope of this thesis.) However, as often experienced, collecting high-quality transcribed data is a time-consuming task. For example, in our meeting recognition research, it needs more than one week for a skilled transcriber to produce the transcripts of a two-hour meeting. Obviously, only relying on manual transcription makes the development of a recognition system too long to be

acceptable for customers. In addition, the cost to cultivate qualified transcribers and maintain a routine transcription is also intolerable for projects with tight budget.

On the other hand, there are other types of speech data which can benefit acoustic model training, e.g. raw speech without any transcripts and closed captioned speech with error-prone transcripts. In most cases, un-transcribed speech data is significantly easier to collect than transcribed one. For example, we can build a program to electronically record the broadcast news in TV and radio programs. Large corpus with thousands hour raw speech can then be built up very quickly with almost no cost. The easiness and low cost of data collection have made un-transcribed speech very attractive. The most challenging and demanding research topic is the investigation of an automatic learning procedure that can exploit both transcribed and un-transcribed speech data for acoustic model training. This kind of approaches is usually referred to as unsupervised training. (Some machine learning literatures use a slightly different term *semi-supervised training* to name the approach since it still uses a small fraction of transcribed data to train an initial acoustic model. However, this thesis follows the tradition of speech community.) Active learning is another approach exploiting un-transcribed speech. It requires human interaction to determine the most informative data for model re-training.

Some TV broadcast programs provide speech recordings along with roughly transcribed closed-captions. Both the speech and captions can be automatically captured and thus to form a usable alternate of transcribed training data. Compared to un-transcribed raw speech, closed captioned speech is more welcomed for its availability of manual transcripts. Recently, research of using closed captioned speech data for acoustic model training has elicited growing interests. This research is referred to as lightly-supervised learning by some literatures since the transcription errors in closed captions need efforts to identify and correct.

6.2 Difficulty to Use Un-Transcribed and Closed-Captioned Speech Data

Raw speech and closed-captioned speech have the potential to substantially reduce the development cost of building up a speech recognition system. However, they can not be used directly in acoustic model training. In order to use un-transcribed raw speech, we have to first produce transcripts for it. The commonly adopted method is to use an initial “seed” acoustic model trained from available transcribed data or borrowed from similar application domain, to

decode the raw speech, and accept the decoding hypotheses as transcripts. Apparently, the quality of the automatically generated transcripts highly depends on the performance of initial model. In the case that the initial model is far from perfect or old domain mismatches to the new domain, the decoding hypotheses will unavoidably contain a large amount of recognition errors. If we add the erroneous transcripts to acoustic model training, as seen in many experiments, it is very likely that the recognition performance of new trained model gets degraded rather than improved.

Similar situation also exists in closed captions. Even though completed by human transcriber, closed captions are known to contain a high percentage of transcription errors compared with carefully prepared transcripts of same shows. For example, omissions of words and phrases are often observed in closed captions. As we know, acoustic model training relies on an alignment between speech signal and phoneme sequence which is derived from transcripts. If the erroneous closed caption with word omissions is used for training, it can lead to the corruption of phoneme model learning.

In addition to the defects of transcripts, raw speech and closed captioned speech also suffer the problem of being short of some important information necessary for acoustic model training, such as audio segmentation, speaker and gender identity, label of non-speech events e.g. noise and music, etc.. Some segmenting and clustering software e.g. CMUseg [Siegler et al, 1997] can be used to automatically generate some of the missing information. However, just like speech recognizer, segmenter can incur extra errors in data processing, and then further increase the difficulty of un-transcribed and closed-captioned speech being used for acoustic model training.

6.3 Data Selection

Given the characteristics of un-transcribed and closed-captioned speech, blindly accepting all the available data is not a good choice because of the existence of transcription errors. Instead, most researchers prefer to adopt a selective training strategy to handle these two types of data. This strategy, referred to as data selection or data filtering, is the key issue of unsupervised and lightly-supervised acoustic model training [Kemp and Waibel, 1999; Kamm and Meyer, 2001; Wessel and Ney, 2001; Lamel et al, 2000; 2004; Visweswariah et al, 2004; Zhang et al, 2005]. Simply speaking, given a large collection of un-transcribed or close captioned speech data, the goal of data selection strategy is to identify and exploit a subset of data that can best improve recognition performance.

The identification of usable un-transcribed raw data is realized by performing confidence scoring to the decoding hypotheses. Confidence scoring is a technique extensively adopted in speech recognition research. Its task is to quantitatively measure how likely a questioned hypothesis is correct. The hypothesis can be a sentence, word, phoneme, or even a state. In implementation, a variety of features extracted from multiple information sources are integrated together by using some machine learning techniques, such as Neural Network, Decision Tree, Bayes Network, Support Vector Machine, etc.., to output a continuous or binary score as the estimate of the correctness of hypotheses. In the case of unsupervised training with un-transcribed data, the hypotheses along with speech data is accepted if its confidence score is greater than an empirically set threshold, otherwise they will be rejected.

The identification of usable closed captioned speech is performed in a similar way [Jang and Hauptmann, 1999a; Jang and Hauptmann, 1999b; Lamel et al, 2000; Chen et al, 2004; Nguyen and Xiang, 2004]. The audio data is first decoded with a biased language model which is over-weighted on the n-grams occurring in closed captions. The recognition hypotheses are then aligned to the corresponding closed-captions with respect to the similarity of word and time-stamp. The fragments which content are agreed by both decoding hypotheses and closed-captions are collected as the new training data. The alignment and voting based selection approach is essentially a confidence scoring technique which implicitly assumes that closed-caption and hypothesis have different error pattern so that the chance that both of them make the same error is very small.

Generally, most data selection approaches aim to identify and exploit correctly transcribed data. In unsupervised training with raw speech, they select data which hypotheses are predicted with high confidence score. In lightly-supervised training with closed captioned speech, they select data endorsed by both closed caption and recognition result. A first glance gives us the impression that this kind of methods is reasonable, because a high confidence score or unanimous voting usually implies that the corresponding transcripts are correct. Expanding the training set with correctly transcribed data should therefore be able to improve recognition accuracy. However, this concept is challenged by our research on English and Mandarin recognition. A detailed analysis will be presented later in this chapter.

6.4 Confidence Scoring Based Data Selection for Unsupervised Acoustic Model Training

The procedure of using un-transcribed audio data for acoustic model training is illustrated in Figure 6.1.

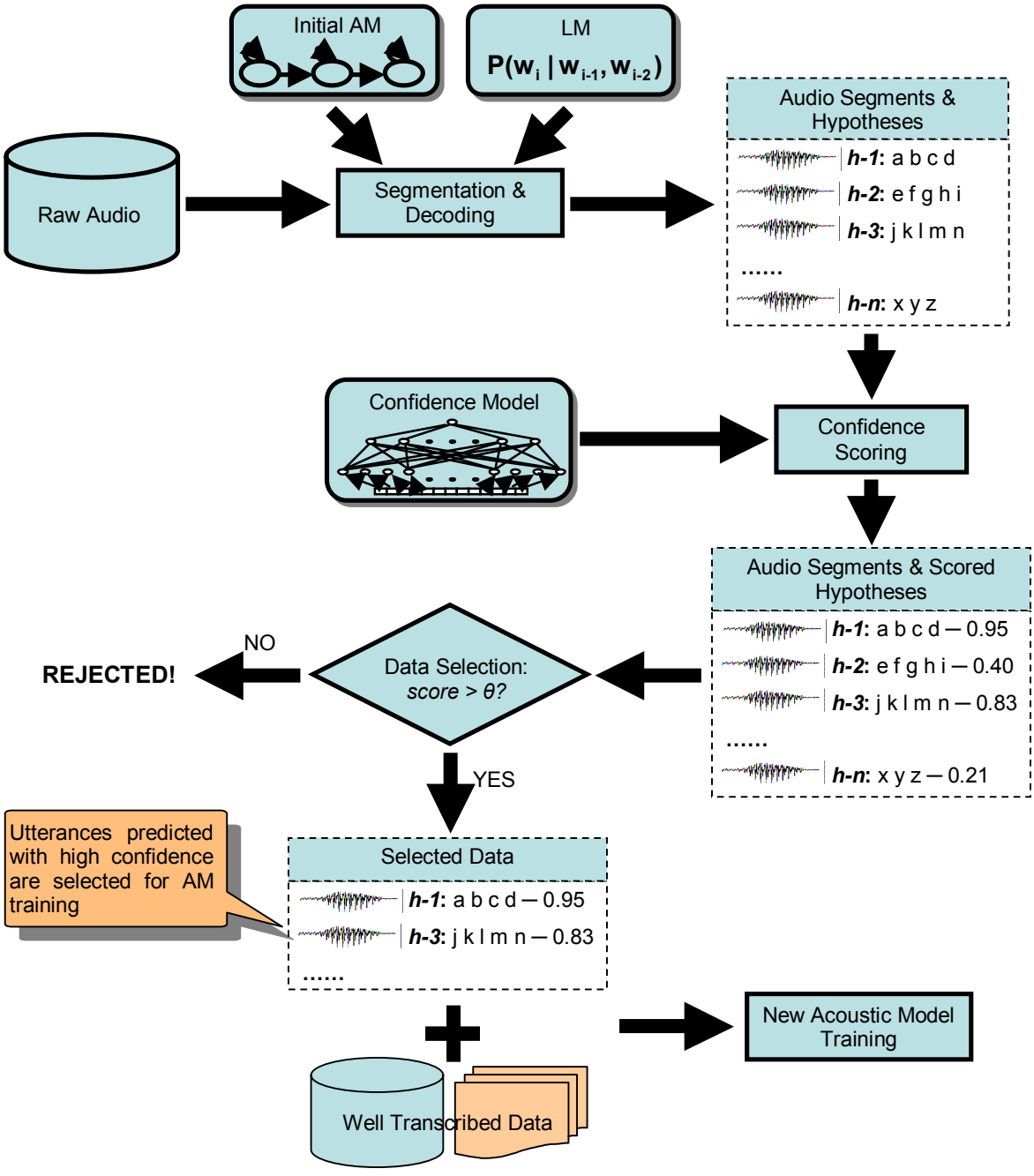

Figure 6.1 Unsupervised acoustic model training with un-transcribed audio. Confidence model is used to measure the correctness of decoding hypotheses. Only the data which confidence scores

are greater than an empirically set threshold have chance of being selected because its hypotheses are more likely to be correctly transcribed.

In Figure 6.1, a "seed" speech recognition system, including the initial acoustic model and language model as well as other necessary modules such as audio segmenter, is used to transcribe the raw data. The initial acoustic model is trained from the transcribed speech corpus or fetched from similar application domain. By performing segmentation and decoding, the continuous audio data is broken down into a sequence of segments, called as utterance in this thesis, and hypothesis is produced for each segment. A confidence model is then applied to measure the correctness of the hypotheses. We assume an utterance level confidence scoring which is to provide a metric indicating how likely the sentence hypothesis as a whole is a reliable recognition result. Correspondingly, the data selection is also carried out on utterance level so that the entire utterance is either accepted or discarded depending on its confidence score. A confidence threshold θ, which value is empirically set or determined by experiments on hold-out set, is used to filter the audio data: only the utterances which confidence scores are greater than θ are selected, otherwise rejected. A new larger speech corpus is then formed by combining the selected utterances, including both audio signal and hypothesis, with the existing transcribed data.

6.5 Alignment and Voting Based Data Selection for Lightly-Supervised Acoustic Model Training

Figure 6.2 illustrates the lightly-supervised acoustic model training process that uses alignment and voting to identify correctly transcribed closed-captioned audio data. A biased language model is trained by over-weighting on the n-grams occurring in closed captions. The continuous audio data is then segmented into isolated utterances and decoded using the biased language model and initial acoustic model. The hypotheses are aligned to closed captions. The fragments which contents are agreed by both hypotheses and captions are assumed to be correct and exacted to form a new training set. In many cases this can result in that only a portion of an utterance rather than the entire utterance is selected, so that some important information may lose. To address this, practically we often require that the length of a selected fragment, in terms of contiguous words or phonemes, must be grater than a minimum value in order to increase the robustness of training.

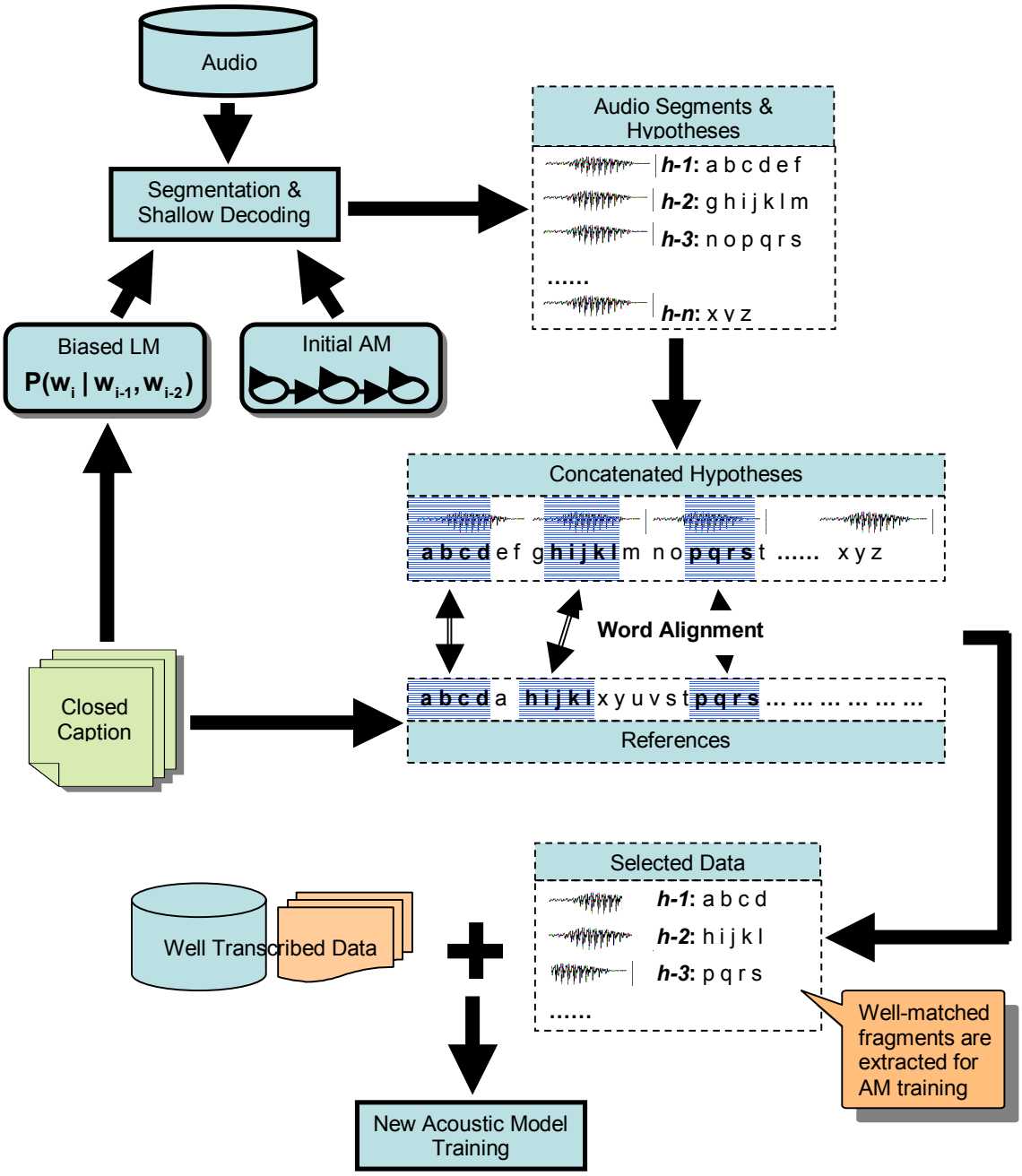

Figure 6.2 Lightly-supervised acoustic model training with closed captioned audio. Word alignment and voting is used to identify the correctly transcribed captions. The fragments agreed by both decoding hypotheses and closed captions are assumed correct and then collected for new acoustic model training.

6.6 Related Work

The potential of un-transcribed and closed captioned audio to provide almost unlimited training resource and substantially reduce manual effort have elicited extensive research interests in speech community. A number of approaches have been studied to improve the performance of unsupervised and lightly-supervised acoustic model training. In [Kemp and Waibel, 1999] un-transcribed data is exploited in a bootstrap fashion with the help of confidence measures. [Riccardi and Hakkani-Tur, 2003] investigated methods combing unsupervised and active learning in order to minimize human supervision as well as maximize system performance. [Jang and Hauptmann, 1999a; Jang and Hauptmann, 1999b] started the research of lightly supervision of using closed captioned broadcast news for their multi-media projects. [Nguyen and Xiang, 2004] proposed an approach that uses a biased language model, instead of general language model, to carry out a fast and constrained recognition. Noticing that mismatched regions between hypotheses and captions may be more desirable in training, [Chen et al, 2004] investigated approaches e.g. consensus network to recover data with recognition errors. Other representative works include [Lamel et al, 2000; Kamm and Meyer, 2001; Wessel and Ney, 2001; Lamel et al, 2002; Cozman et al, 2003; Moreno et al, 2003; Visweswariah et al, 2004; Ma and Matsoukas, 2007] etc..

As we have mentioned, there is a notable characteristic shared by unsupervised and lightly-supervised training approaches: they concentrate on how to find out correctly transcribed data. To achieve this, confidence scoring or word alignment techniques are investigated. We will show next that the confidence scoring based data selection is not always suitable for acoustic model training, and there exist other important issues which must be considered in order to make an effective use of un-transcribed and roughly transcribed audio data [Zhang and Rudnicky, 2006a].

6.7 Problems with Confidence Scoring Based Data Selection

This section discusses the problem of using conventional confidence scoring based data selection strategy in unsupervised acoustic model training. Similar problem also exists in lightly supervised training.

We consider the following scenario of selecting un-transcribed or roughly transcribed audio data for acoustic model training. An initial acoustic model λ_0 is learned from available transcribed set and then applied to recognize the audio data which has been segmented into a set of utterances. A confidence metric $f(h;\mathbf{x})$ is called to provide each utterance a score measuring the likelihood of correctness for the recognition hypothesis. Here, $\mathbf{x}$ denotes the feature sequence for an utterance, and h denotes the hypothesis given by model λ_0. We assume $f(h;\mathbf{x})$ is an utterance level metric which is to quantify the correctness of whole sentence hypothesis h. Utterances, segmented audio along with hypothesis, scored with high confidence are selected and then combined with well-transcribed data to train a new acoustic model λ_1 which is expected to have better performance than λ_0. Please see Figure 6-1 and 6-2 for a graphical illustration.

First, let's consider a question: is the confidence scoring model independent of the recognition model? There exists an extreme example often referred by machine learning research: co-training for webpage classification [Blum et al, 1998], in which the feature sets can be split into two unrelated and redundant subsets, each of which is sufficient for classification. We can use one of them to construct a classifier, and the other to construct a confidence annotator for measuring the correctness of classification result made by the former. To some extent, this can result us an unbiased confidence metric independent to classifier. However, most speech recognition applications cannot offer such a feature division. Instead, the confidence metric in speech recognition is primarily constructed on the basis of information supplied by the recognition model. For example, *Posterior word probability* is one of the most effective confidence measures for continuous speech recognition [Wessel et al, 1998]. Its implementation is shown in (6-1), in which the hypothesis space is restricted to word lattice:

$$P_\lambda(< w, t_s, t_e >| \mathbf{x}) = \frac{\sum_{h \atop s.t.<w,t_s,t_e>\in h} P_\lambda(h,\mathbf{x})}{\sum_{h \atop s.t.\, h\in word-lattice} P_\lambda(h,\mathbf{x})} \tag{6-1}$$

where $\mathbf{x}$ denotes the feature sequence for input speech, w is the questioned word with starting time t_s and end time t_e, and h denotes a path (sentence hypothesis) in the word lattice. Please note that *Posterior word probability* is calculated from the acoustic score $\log[P_\lambda(x \mid h)]$ and language model score $\log[P_\lambda(h)]$, both of which are output by the recognizer.

Because of the high correlation between confidence metric and recognizer in the features exploited, the selection of un-transcribed and roughly transcribed data with high confidence score often leads to only those examples that match well to the current recognition model being picked. The high scored examples, even though more likely to be correctly recognized, can't add substantial new information to the acoustic mode training. Training with such examples will consequently be a process that reinforces what the current model already encodes [Kemp and Waibel, 1999], yet it is unable to reduce the estimation bias in initial acoustic model caused by scarcity of well-transcribed data or an inaccurate model assumption.

The discussion above shows that imperfect confidence scoring model, especially those correlated with recognizer, is unable to identify the most usable data for unsupervised acoustic model training. To address this problem, researchers work on various methods to improve the performance of confidence scoring, e.g. investigating new features less correlated to recognizer [Zhang and Rudnicky, 2001]. Theoretically, the *true* posterior probability $P(h \mid \mathbf{x})$ is the best confidence metric if it can be learned from training data. In speech recognition, the construction of a confidence scoring model can be viewed as an effort to approach $P(h \mid \mathbf{x})$ with some empirical assumptions. A widely accepted concept is that confidence scoring model is capable to discover good data for acoustic model training, if it is a perfect approximation to $P(h \mid \mathbf{x})$. However, this isn't always true for continuous speech recognition.

The construction of a speech recognition system is essentially a process to learn a probabilistic representation $P(\mathbf{x}, y)$ for the joint distribution of $\mathbf{x}$ and y. Here, $\mathbf{x}$ is the acoustic feature vector, and y is the corresponding class e.g. word, phoneme, contextual phoneme or tied state. Using chain rule, $P(\mathbf{x}, y)$ is further broken down into the product of $P(y)$ and $P(\mathbf{x} \mid y)$. $P(y)$ and $P(\mathbf{x} \mid y)$ are learned by language model training and acoustic model training respectively. In order to have an unbiased estimate of $P(\mathbf{x}, y)$, the selection of un-transcribed examples should also follow $P(\mathbf{x}, y)$. However, this requirement can not be fulfilled by confidence scoring based data selection which follows $P(y \mid \mathbf{x})$ instead. No matter how good the confidence metric is, the utilization of un-transcribed examples chosen with $P(y \mid \mathbf{x})$ may result in an erroneous estimate to $P(\mathbf{x}, y)$, sometimes even worse than the initial model learned from the small transcribed set.

Please note the underlying distribution $P(\mathbf{x}, y)$ is unknown to us, so it is impossible for us to use it directly in data selection. A reasonable and practical back-off is to perform a diversified selection that the selected examples can cover occurring acoustic phenomena and phonetic classes.

However, confidence metric based selection can not realize this either. In our preliminary experiments with a variety of confidence metrics, we observed that the selected un-transcribed examples often locate in some special regions, e.g. the region far from the class boundary, rather than distribute globally across the entire input space. In addition, we also observed that in some applications there is no un-transcribed example being selected for certain classes, because most of the examples of the classes are located in a region classified with low confidence. Obviously, this will result in a biased estimate of the underlying distribution.

6.8 Summary

This chapter presents an overview of unsupervised and lightly-supervised training approaches that utilize roughly transcribed or un-transcribed data to improve the performance of speech recognition system. We described the confidence scoring based selective training method and its applications in identifying correctly transcribed speech data. In the last section, we present a discussion that shows only relying on confidence metric for data selection can lead to an erroneous estimate of underlying distribution and cause degradation of recognition accuracy. Next chapter will present our solution, a clustering based data selection strategy, to address this problem.

7 Clustering Based Data Selection Approach for Unsupervised and Lightly Supervised Training

In previous chapter, we discussed the importance of using un-transcribed and roughly transcribed speech data (e.g. closed captioned speech) in improving recognition performance, and described the conventional selective training approach that relies on confidence metric to select correctly transcribed data for acoustic model training. We have shown that, due to its correlation to speech recognizer, as well as its conflict to the training criterion of generative model (e.g. Gaussian Mixture), confidence metric alone isn't sufficient in identifying the best usable data. We observed in preliminary experiments that the conventional approaches can result in only the examples matching well to the existing model are selected. Even though they are more likely to be correctly transcribed, these examples can not contribute much new information to model training. As a consequence, this can lead to a suboptimal model which performance may be even worse than the one without using any new data.

To address this problem, we present a novel clustering based selection strategy that aims to increase the diversity of selected utterances while maintain the correctness of the transcripts [Zhang and Rudnicky, 2006a]. This chapter is organized as follows. Our new strategy is described in Section 1. We then evaluate it using four speech corpora: *ICSI meeting dataset*, *1997 Mandarin Broadcast News*, *TDT-4 Mandarin Broadcast News*, and *GALE BN03 Mandarin Dataset*. These datasets include three different types of training data: transcribed speech, un-transcribed speech and closed captioned speech. They also cover recognitions of two totally different languages: phoneme based English and syllable based Mandarin. This chapter is concluded in Section 4 with a brief summary.

7.1 Clustering Based Data Selection Strategy

A question that need be answered first is that at what level the data selection should be performed. Options include utterance, word, phoneme or even state. In our experiments, we use utterance as the basic unit and data selection is thus conducted in the fashion that an utterance is accepted or rejected as a whole. Compared to word or sub-word units, utterance is a more natural way to organize speech data. It's much easier for audio segmenters to detect the boundary of an utterance

than that of word or sub-word. In addition, the co-articulation effect, which is important for acoustic modeling, can be preserved when using utterance. Correspondingly, confidence scoring is also performed on utterance level that the object to be evaluated is the entire hypothesized sentence rather than each single word or sub-word.

7.1.1 Increase Diversity in Data Selection

As we discussed in previous chapter, conventional data selection approaches only focus on how to ensure if the selected examples have correct transcripts, while ignore other issues. In contrast, our strategy considers two aspects together in data selection: the *correctness* of transcripts and the *diversity* of selected data.

1. Identifying examples with correct transcripts is still the primary goal of our data selection strategy. The transcripts are either manually labeled as in closed captioned speech, or obtained by automatic decoding as in unsupervised training with raw speech. Even though it is commonly observed that acoustic modeling is robust, to some extent, to the transcription errors, too many such errors are still a serious problem for learning a good model. Therefore, our approach also utilizes confidence scoring technique as a means to measure the correctness of questioned transcripts.
2. In addition, we consider the distribution of selected data. Ideally, we want the examples being selected comply with the true distribution. Since the true distribution is unknown to us, in implementation, we seek a *diversified* selection that aims to increase the diversity of selected data i.e. its coverage to acoustic and phonetic phenomenon. Namely, we hope every occurring acoustic and phonetic class can have sufficient representatives being selected into new training set.

The key issue to achieve a diversified selection is how to model and capture variability of speech via an automatic learning procedure. In our research, this is realized by an unsupervised clustering scheme. In the case of training with un-transcribed speech, the input audio is first broken down into segments, referred as utterances in this thesis, with the help of an audio segmenter e.g. CMUseg [Siegler et al, 1997] or ISL-segmenter [Jin and Schultz, 2004]. Utterances are converted into feature sequences $\mathbf{x}$, and decoded by an existing acoustic model to generate hypotheses y. The duple $< \mathbf{x}, y >$ can be viewed as a point and all the $< \mathbf{x}, y >$ together form a high dimensional input space. Clustering algorithm e.g. K-Means is called in to

partition the space into a number of clusters. Each cluster is a collection of utterances which share some behaviors in common. Specifically, they have similar acoustic patterns or belong to certain phonetic classes. Thus the varieties of speech implicitly carried by utterance $< \mathbf{x}, y >$ are categorized into clusters. By sampling these clusters respectively, we are able to collect utterances representing different acoustic and phonetic phenomena. Confidence scoring is performed with sampling in order to give utterances with correct hypotheses higher chance of being selected. The sampling is carried out as follows: for each cluster, all the utterances are sorted according to their confidence score from high to low, and the top *n*% utterances with the highest scores are then identified as the desirable data.

Please note that if the number of clusters is set to 1 that the entire input space is treated as a single cluster, the above strategy is back to conventional confidence scoring based data selection which only considers the correctness of hypothesis y. On the other hand, if the number of clusters is set to the number of utterances, namely, each $< \mathbf{x}, y >$ is a cluster by itself, the above strategy is equivalent to perform a random selection that every utterance $< \mathbf{x}, y >$ has equal chance, *n*%, of being selected. In this case, the requirement of *diversity* is satisfied while the requirement of *correctness* is not. By changing the number of clusters, the proposed strategy allows us to find an appropriate point to balance the two requirements, *correctness* and *diversity*, in data selection.

Clustering in input space enables us to model and capture the variability of speech phenomena, and thus to perform a diversified selection. Please note an utterance can carry multiple information, e.g. acoustic info, phonetic info, semantic info, speaker info, etc.. Currently, we only exploit acoustic and phonetic information. The investigation of how to use other information is an important issue in our future research. Currently, we adopt K-Means as the clustering method, which is equal to assume the distribution of utterances $< \mathbf{x}, y >$ follows Gaussian Mixtures. Apparently, this is a strong assumption. However, as an extensively adopted method to depict acoustic observations, the effectiveness of Gaussian Mixtures has been manifested by many recognition experiments. So in this thesis we just keep using this assumption on a practical basis.

The clustering based data selection strategy for unsupervised acoustic model training is formulated in Table 7.1. The strategy can be extended to lightly supervised training in a similar way except the confidence scoring model $f(h;\mathbf{x})$ is replaced by word alignment and voting.

Input:

Ψ_T : Transcribed set

Ψ_U : Un-transcribed set.

M : Number of clusters.

$n\%$: Percentage of data to be selected.

Initialization:

1. Learn an initial recognition model $\boldsymbol{\lambda}$ from transcribed set Ψ_T.

2. Learn a confidence model $f(h;\mathbf{x})$ from transcribed set Ψ_T as well as other available information sources.

3. Decode all the un-transcribed utterances $<\mathbf{x},?>\in\Psi_U$ using initial model $\boldsymbol{\lambda}$, so that $<\mathbf{x},?>\Rightarrow<\mathbf{x},y>$ where y is the decoding hypothesis.

4. Compute confidence score for each utterance $<\mathbf{x},y>\in\Psi_U$ using confidence scoring model $f(h;\mathbf{x})$.

5. Convert the utterances $<\mathbf{x},y>\in\Psi_U$ to utterance super vectors $\mathbf{z}$.

6. Partition the utterance vectors into M clusters $\mathbf{D}_1,\mathbf{D}_2,...,\mathbf{D}_M$ using K-means.

Unlabeled Data Selection:

For each cluster $\mathbf{D}_m$ $(1\le m\le M)$, :

7. Sort unlabeled utterances $<\mathbf{x},y>\in\mathbf{D}_m$ according to the confidence score from high to low.

8. Add the top $n\%$ utterances $<\mathbf{x},y>\in\mathbf{D}_m$ to the transcribed set Ψ_T.

Re-training:

9. Train a new model $\boldsymbol{\lambda}'$ using the new Ψ_T which is the combination of original transcribed set and selected utterances.

Table 7.1 Clustering based date selection strategy used in unsupervised acoustic model training. The values for parameters M and $n\%$ are either determined empirically or determined by experimenting on hold-out set. Please see Section 7.1.2 for how to convert an utterance $<\mathbf{x},y>$ to a fixed length vector $\mathbf{z}$.

7.1.2 Converting Utterance into Fixed Size Vector

To perform clustering algorithm e.g. K-Means to input space consisting of utterance, an obstacle has to be addressed. The length of utterances varies considerably in terms of the number of words, phonemes and frames. Specifically, a training utterance is expressed as $< \mathbf{x}, y >$ where $\mathbf{x}$ is the sequence of feature vectors representing acoustic info, and y is the transcripts from which phonetic info can be derived. Both the number of frames of $\mathbf{x}$ and the number of phonemes of y change a lot from utterance to utterance. Obviously, K-Means algorithm is unable to handle the utterance directly.

To address this problem, we present normalization scheme that converts an arbitrarily long utterance into a fixed size vector [Zhang and Rudnicky, 2006a]. The acoustic and phonetic information carried by $\mathbf{x}$ and y are then conveyed into the vector. Simply speaking, $\mathbf{x}$ is converted into a high dimensional vector $\mathbf{u}$ which represents the distribution of the questioned utterance over possible acoustic classes. On the other hand, y is converted into another high dimensional vector $\mathbf{v}$ which represents the distribution of the questioned utterance over possible phonetic classes. The reasons that we choose phoneme instead of other units, such as word or triphone as the basis of normalization are due to following concerns: (1) phoneme is a natural speech unit related to acoustic model training, especially to the context-independent model training; (2) the number of phonemes is proper for vector operation. For example, an English speech recognition system commonly exploits 40~50 phonemes. In comparison, the number of words or triphones can be in thousands. Once the conversion is done, we concatenate $\mathbf{u}$ and $\mathbf{v}$ to form the utterance vector. The detail of normalization is given as follows.

Suppose a training set $\Psi = \{< \mathbf{x}_i, y_i > | 1 \le i \le N\}$ Where $\mathbf{x}_i$ denotes the acoustic feature sequence of utterance i with T_i frames that $\mathbf{x}_i = [x_{i,1}, x_{i,2}, ..., x_{i,T_i}]$, and y_i denotes the corresponding transcripts containing K_i words that $y_i = [y_{i,1}, y_{i,2}, ..., y_{i,K_i}]$. Please note for an un-transcribed dataset, y_i is the decoding hypothesis generated by an existing acoustic model. Here $x_{i,t}$ denotes the feature vector at frame t, and $y_{i,k}$ denotes a hypothesized word.

The conversion of $\mathbf{x}$ ***to*** $\mathbf{u}$ ***:***

1. *Build acoustic feature space* $\mathbf{X}$ *by collecting all the frame vectors* $x_{i,t}$ *together, that* $\mathbf{X} = \{x_{i,t} \mid 1 \le i \le N, 1 \le t \le T_i\}$. *Each frame feature vector* $x_{i,t}$ *can be viewed as a point in space* $\mathbf{X}$.

2. *Partition* $\mathbf{X}$*-space into* L *clusters using unsupervised clustering algorithm e.g. K-Means. Each cluster can be interpreted as a special "class" representing certain speech phenomenon. The value of* L *is empirically set. We thus obtain a "codebook"* $\mathbf{C} = \{c_1, c_2, ..., c_L\}$ *in which "codewords"* c_l *is the centroid of the* l*-th cluster.*

3. *For* $\mathbf{x}_i = [x_{i,1}, x_{i,2}, ..., x_{i,T_i}]$, *calculate the closest cluster for each frame vector* $x_{i,t}$ *that* $c_{i,t} = \arg\min_{1 \le l \le L} dist(x_{i,t}, c_l)$ *where* $dist(.)$ *is the distance function, and then convert* $\mathbf{x}_i$ *from a sequence of feature vectors to a sequence of "codeword", so that* $\mathbf{x}_i = [x_{i,1}, x_{i,2}, ..., x_{i,T_i}] \Rightarrow \mathbf{x}_i' = [c_{i,1}, c_{i,2}, ..., c_{i,T_i}]$

4. *Further convert* $\mathbf{x}_i'$ *to a* L *dimensionality vector* $\mathbf{u}_i = [u_{i,1}, u_{i,2}, ..., u_{i,L}]$ *where* $u_{i,l} = \frac{\#\ of\ c_l\ in\ \mathbf{x}_i'}{T_i}$. *The value of* $u_{i,l}$ *is the normalized number of times that frame vectors are mapped to cluster* c_l.

Therefore, an utterance with T_i frames is normalized into a L dimensional vector. Each dimension of the vector describes how often a particular acoustic pattern, represented by the corresponding cluster, occurs in the questioned utterance. The vector as a whole describes how the utterance is distributed over these clusters.

The conversion of y ***to*** $\mathbf{v}$ **:**

The conversion of transcripts in terms of word to a fixed length vector is conducted in a similar way except phonemes are adopted as naturally formed clusters.

1. *Convert transcripts from word string to phoneme string. This can be done by simply looking-up system dictionary, so that* $y_i = [y_{i,1}, y_{i,2}, ..., y_{i,K_i}] \Rightarrow y_i' = [q_{i,1}, q_{i,2}, ..., q_{i,M_i}]$ *where* $q_{i,m}$ *denotes a phoneme.*

2. *Suppose system phoneme set has R phonemes:* $\mathbf{Q} = \{q_1, q_2, ..., q_R\}$. *We then convert* $y_i' = [q_{i,1}, q_{i,2}, ..., q_{i,M_i}]$ *into a R dimensional vector* $\mathbf{v}_i$, *that* $\mathbf{v}_i = [v_{i,1}, v_{i,2}, ..., v_{i,R}]$ *where* $v_{i,r} = \frac{\#\, of\ q_r\ in\ y_i'}{M_i}$.

The r-th dimension of $\mathbf{v}_i$ corresponds to the r-th phoneme in system phoneme set. Its value is the normalized number of times that the *r*-th phoneme is observed in the utterance. So the vector $\mathbf{v}_i$ essentially depicts how the utterance transcripts distribute over phonemes.

Concatenation of $\mathbf{u}$ ***and*** $\mathbf{v}$***:***

The final utterance vector is the concatenation of acoustic vector $\mathbf{u}$ and phoneme vector $\mathbf{v}$: $\mathbf{z} = \mathbf{u} + \mathbf{v} = [u_{i,1}, u_{i,2}, ..., u_{i,L}, v_{i,1}, v_{i,2}, ..., v_{i,R}]$. Thus the similarity between utterances can be measured by the distance between their normalized vectors. The dimensionality of utterance vector $\mathbf{z}$ is $L + R$. In our English recognition system, L, the number of clusters in acoustic space, is set to 256 and R, the number of phonemes, is 49, which results in each utterance vector has as high as 305 dimensions. Such a high dimensionality can cause problem for utterance level clustering. Thus investigation on dimension reduction method is necessary and is scheduled in our future research plan.

Please note that in our approach, the clustering technique is applied twice at different level. The first one is the clustering performed in frame vector space in order to normalize acoustic observation into fixed length vector. The second one is used to partition utterance space into a number of *classes*, each of which reflects certain acoustic and phonetic patterns, so that the diversity of selected data can be increased by sampling these classes respectively.

7.2 Experiment I: Unsupervised Acoustic Model Training

This section presents the experimental results of the clustering based data selection approach in unsupervised acoustic model training. The approach is evaluated on two speech recognition tasks: English meeting domain recognition and Mandarin broadcast speech recognition.

7.2.1 Unsupervised Acoustic Model Training on ICSI Meeting Dataset

Dataset and Configurations

Our first experiment is conducted on ICSI meeting dataset [Janin et al, 2003; Banerjee et al, 2004]. The dataset has a total of 75 meetings, accounting for 70 hours of speech data. We use 10 meetings as the transcribed set for initial acoustic training, 61 meetings as the un-transcribed set for unsupervised training, 3 meetings as the hold-out set for recognizer tuning, and 1 meeting (containing 7500 words) as the test set. A 13-dimension MFCC feature vector is computed for each frame and then expanded to 39-dimension by adding delta and delta-delta coefficients. The phone set contains 49 basic phonemes. A 3-state left-to-right HMM is adopted to model each speech unit. In context dependent training stage, these phonemes are transformed to triphones and then tied together to make 2000 senones. Each senone is modeled using a mixture of 32 Gaussians, giving a total of 64K Gaussians for acoustic modeling.

The language model was trained from the transcripts of 10-meeting transcribed set, without any access to the un-transcribed set. The language model is fixed in our experiments, since our main goal is to investigate suitable approaches for acoustic model training. We believe that unsupervised learning technique could be applied to language model training as well, and this issue has been listed in our research plan. The dictionary adopted in the experiments was based on the CMU Dictionary (containing more than 125k words). The actual vocabulary used in decoding was the intersection between dictionary and language model lexicon (unigram), which consists of 4200 words.

Confidence Measures

Our experiments employ a Neural Network based confidence annotator to measure the correctness of hypothesis. Data selection is performed on utterance level so that an utterance is either kept or rejected as a whole depending on its confidence score. The inputs to the Neural Network consist of four features representing both language model and acoustic model information: *LM-backoff-mode*, *Utterance-level-posterior-probability*, *Word-level-posterior-probability* and *Frame-level-posterior-probability* [Zhang et al, 2005], while the output is trained to approximate the word accuracy of sentence hypothesis. The Neural Network based confidence annotator is trained and tested on the 10 meetings transcribed data. Figure 7.1 plots its performance and shows the relationship between confidence score and word accuracy of

hypotheses (and also shows standard deviations). Figure 7.1 shows that the confidence score is generally proportional to word accuracy; that is, high confidence score indicates high accuracy and vise versa.

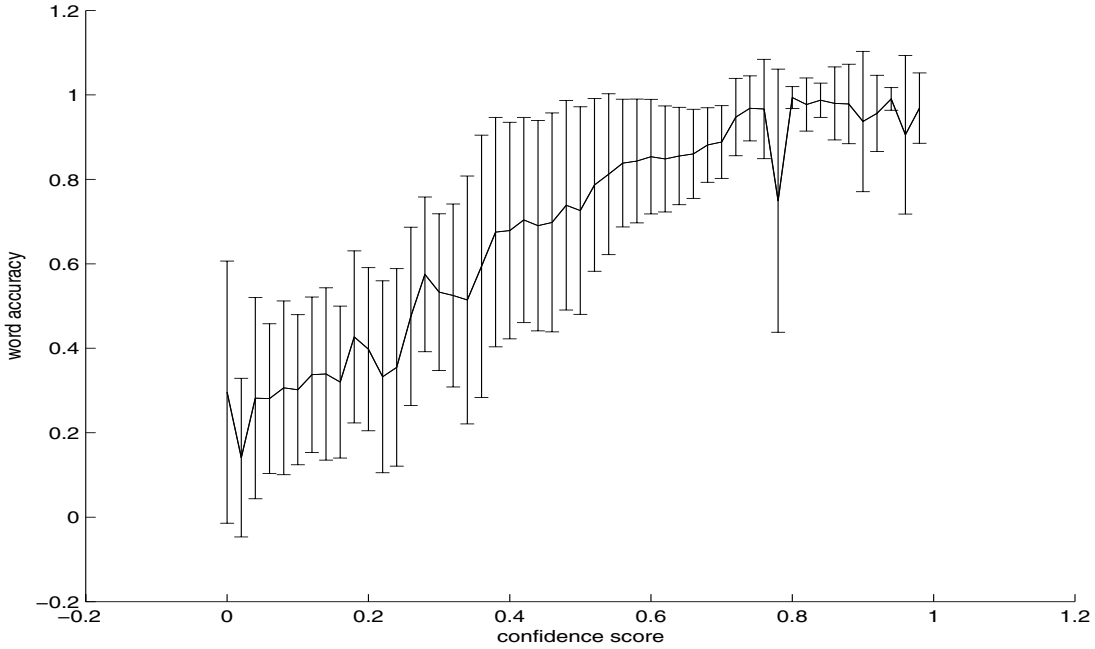

Figure 7.1 Performance of Neural Network based confidence annotator. X-axis denotes the confidence score that Neural Network predicts for a sentence hypothesis. Y-axis denotes the word accuracy of the hypothesis.

Acoustic Model Trained from	Word Error Rate
Transcribed Data	47.31%
Transcribed + Un-transcribed Data	44.41%

Table 7.2 Performances of two baseline systems. The first one is trained using transcribed data only. The second one is trained from the combination of transcribed and un-transcribed data, in which the hypothesis of un-transcribed data is decoded by the first model.

Baseline

The initial acoustic model was trained using the 10 transcribed meetings and realized a word error rate of 47.31% . All the un-transcribed speech data are decoded by using the initial model, and then appended to the transcribed set, along with their hypotheses, to train a new acoustic model.

Table 7.2 compares the word error rates of these two models. When using all the available un-transcribed data in acoustic model training without any data selection, the word error rate is down to 44.41%, which represents a 6.1% relative reduction from the word error rate of initial model.

Investigation of Utterance Vectorization in Data Selection

In the proposed clustering based data selection approach, each training utterance is first converted into a high dimensional vector $\mathbf{z}$ which is the concatenation of sub-vector $\mathbf{u}$ and sub-vector $\mathbf{v}$, representing acoustic and phonetic information respectively. The following experiment will show that the two types of information are both desirable, and complementary to each other, in achieving a diversified selection of un-transcribed data. To do this, three utterance vectorization methods are compared in the experiment: (1) only using $\mathbf{u}$ to represent an utterance; (2) only using $\mathbf{v}$ to represent an utterance; and (3) using $\mathbf{u}+\mathbf{v}$ to represent an utterance. 11 hours or 20% of un-transcribed speech data are selected by using the clustering based approach, and then added to the existing transcribed set to train new acoustic models. The cluster numbers in frame level and utterance level clustering are 256 and 64 respectively. Table 7.3 shows the performances of the acoustic models trained using the three different utterance vectorization methods. The experimental result demonstrates the importance of combining acoustic and phonetic information in utterance representation. It also suggests that recognition performance can be potentially further improved by incorporating more information e.g. speaker and gender info into vectorization. All the experiments reported in this chapter will adopt $\mathbf{u}+\mathbf{v}$ as the representation of utterance vector.

Utterance Vectorization	Word Error Rate
Using $\mathbf{u}$	46.15%
Using $\mathbf{v}$	45.33%
Using $\mathbf{u}+\mathbf{v}$	44.36%

Table 7.3 Comparison of three different utterance vectorization methods used in clustering based data selection.

Comparison of Clustering Based Data Selection with Confidence Based Data Selection

We further compare our clustering based data selection approach with traditional approaches that use only high confidence scores as the standard to select utterances. As in the previous experiment, the cluster number set to frame level and utterance level clustering are 256 and 64 respectively. The amount of un-transcribed speech data selected for model re-training is increased by 11 hours (or 20% of total un-transcribed speech) every time. Figure 7.2 plots the word error rates of the two approaches.

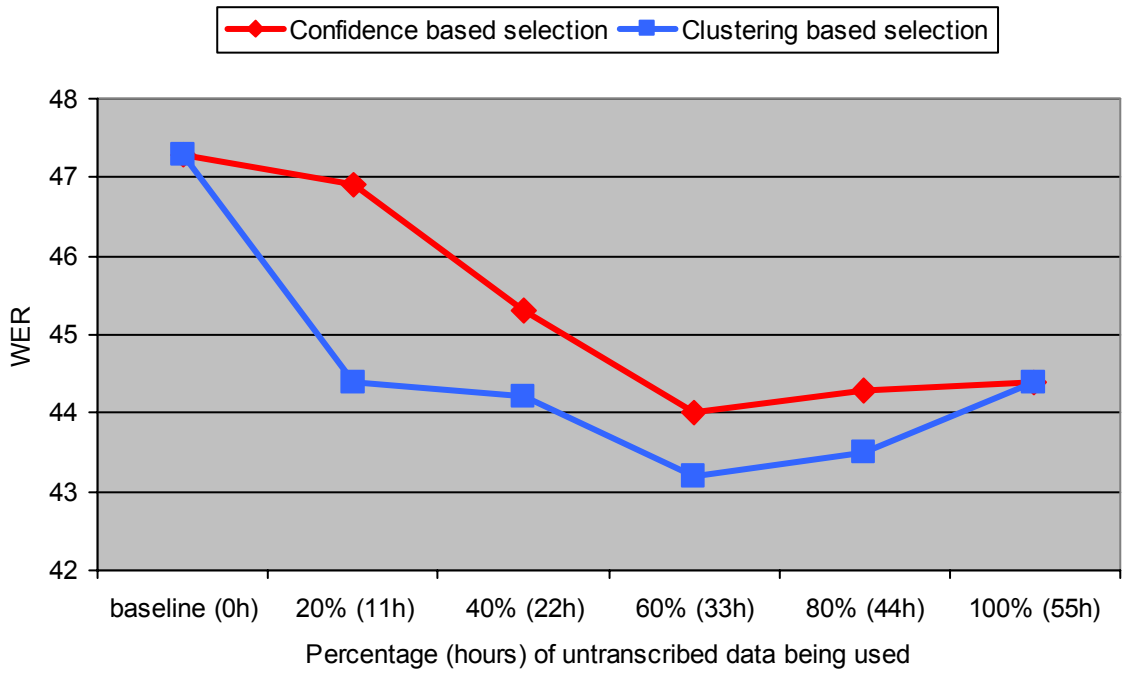

Figure 7.2 Comparison of conventional confidence based and the proposed clustering based data selection approaches in unsupervised acoustic model training. The performances are measured by Word Error Rate (WER). The amount of un-transcribed speech data selected for model re-training is increased by 11 hours (or 20% of the total un-transcribed speech) every time. 256 clusters are used in frame level clustering, and 64 clusters are used in utterance level clustering as required by the proposed data selection approach.

Figure 7.2 shows that our new approach is consistently superior to traditional method. When the first 20% un-transcribed data are added to the training set, our approach reduced the word error rate to 44.36% from the baseline of 47.31%, and began to outperform the model trained using all the un-transcribed data. In contrast, traditional method only reduced the error rate to 46.89% with the same amount of data. This indicates that the most suitable un-transcribed data for model training cannot be identified if we ignore its distribution in feature space. We further conducted

significance test, paired *t*-test, on performance differences between the two selection methods. We obtained that *t*= 3.20 and *P-value*=0.049. Therefore, at 0.05 level, the clustering based data selection method significantly outperforms traditional method.

Both approaches reach their best performance, 43.18% and 44.02% respectively, when using 33 hours or 60% of the un-transcribed data. After that, adding more un-transcribed data to acoustic model training starts to deteriorate recognition performance rather than to improve it. This phenomenon demonstrates the importance of data selection in unsupervised acoustic model training.

7.2.2 Unsupervised Acoustic Model Training for Mandarin Speech Recognition

Dataset and Configurations

The second experiment is part of our research for GALE Mandarin speech recognition project. The transcribed set is *1997 Mandarin Broadcast News* which has about 30 hours speech data. The un-transcribed set for unsupervised training is *GALE BN03 Mandarin Dataset* which has 1800 hours raw speech. For the test set, we use *RT-04 Eval* set which is one of the standard test sets in GALE project. 39 dimension MFCC feature vector (12 MFCC + 12 delta-MFCC + 12 delta-delta-MFCC + 3 power) is calculated for each frame. No any other frond-end processing technique is used. CMU-ISL audio segmenter [Jin and Schultz, 2004] is adopted to segment raw audio that results in 980K utterances.

The phone set contains 179 phonemes, created by converting Mandarin tonal syllable into initial/final combinations. Please note that the tone information is explicitly modeled into final phonemes. For example, *an3* denotes a phoneme with the third tone. A 3-state left-to-right HMM is adopted to model each speech unit. In context dependent training stage, these phonemes are transformed to triphones and then tied together to make 2000 senones. Each senone is modeled using a mixture of 64 Gaussians, giving a total of 128K Gaussians for acoustic modeling.

The language model was trained from the collection of a variety of available Chinese text datasets, including Giga-word II, TDT-2, TDT-3 and TDT-4, most of which can be obtained from LDC (http://www.ldc.upenn.edu/). Language model training is performed using CMU SLM toolkit (http://www.speech.cs.cmu.edu/SLM_info.html) [Clarkson and Rosenfeld, 1997]. There are 64000 unigrams, 16.6M Bigrams and 20.7M trigrams in the final language model. The dictionary for testing uses the same lexicon of language model.

Please note that in our experiments, the performance of Mandarin recognizer is measured by *character error rate* as required by GALE project instead of Word Error Rate. We keep using Neural Network based confidence annotator to measure the correctness of decoding hypothesis. Please refer to Section 7.2.1 for detail.

Baseline

The baseline acoustic model is built by only using the 30 hours *1997 Mandarin Broadcast News* for training, Character error rate for this model is 29.90%

Comparison of Clustering Based Data Selection with Confidence Based Data Selection

In this experiment, we use 512 clusters in frame level clustering. Since the amount of un-transcribed speech is too large, it is impossible to perform clustering with all the frame vectors in memory. So we sample the frame vectors by a ratio of 20:1, that is for every 20 frame vectors, only one is picked up for frame level clustering. Apparently this can cause degradation of clustering performance. The number of clusters in utterance level clustering is set to 128.

The baseline model trained from transcribed data is applied to decode the 980K un-transcribed utterances for generating sentence hypotheses for them. The two data selection approaches are then performed to identify the most usable data based on their separate criteria. We compare the performances of these two approaches by varying the amount of un-transcribed speech they selected. The two approaches are tested on 50 hours, 100 hours and 200 hours respectively. Result is plotted in Figure 7.3.

Figure 7.3 shows that adding un-transcribed speech to acoustic model training has the potential to significantly improve recognition performance. For example, when augmenting transcribed set (30 hours) with 100 hours un-transcribed speech, the character error rates are down to 25.61% and 25.15%, for confidence based selection and clustering based selection, respectively. Both of them achieve more than 14% relative reduction of CER. Figure 7.3 also shows that our clustering based approach consistently outperforms traditional method. Paired t-test is conducted to further evaluate the performance difference between the two methods. The test is calculated on the character error rates of acoustic models trained using 50, 100 and 200 hours selected un-transcribed data. We obtained that t=4.86 and P-*value*=0.040, which shows that at 0.05 level, the clustering based data selection method significantly outperforms confidence based method.

Another considerable phenomenon is that the difference of performance between the two data selection approaches becomes smaller when using more un-transcribed data. As we discussed, conventional confidence based selection focuses on fulfilling the requirement of *correctness* while ignores the *diversity* of selected utterances. With more data added for training, this weakness is alleviated to some extent. However, this also increases the chance of utterances with incorrect transcripts being selected, and thus causes the degradation of recognition performance. For example, the CERs of the models trained using 200 hours un-transcribed data are higher than those using 100 hours. Therefore, the better strategy, as pursued by our clustering based approach, is to consider the two requirements together in data selection: the *correctness* of transcripts and the *diversity* of selected data.

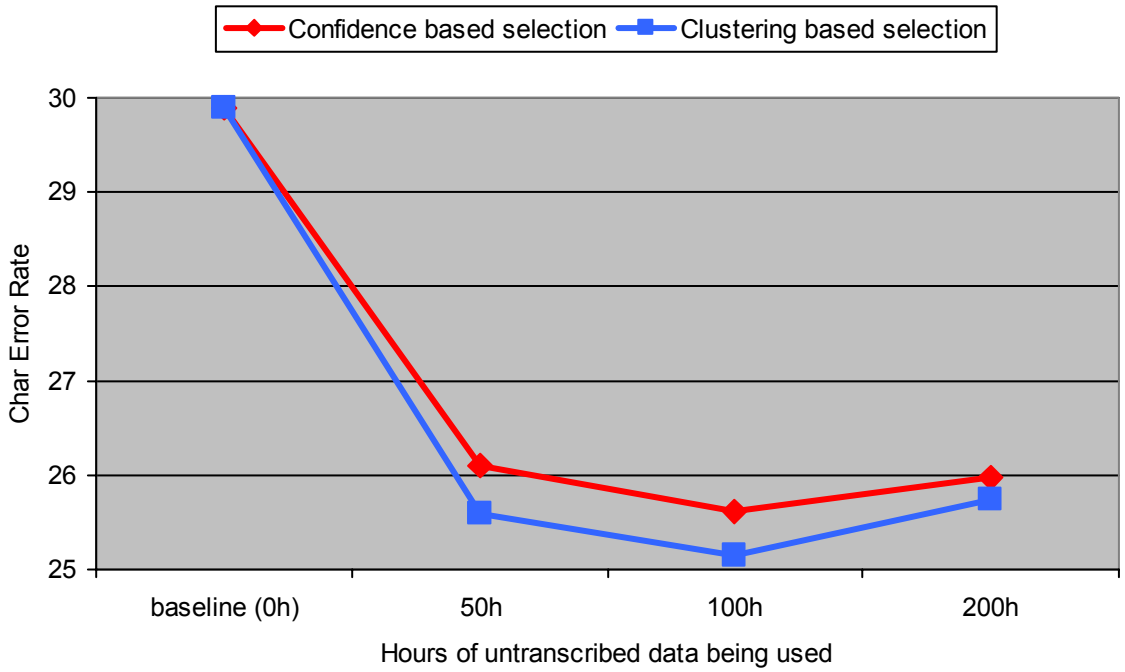

Figure 7.3 Comparison of conventional confidence based and the proposed clustering based data selection approaches in unsupervised acoustic model training. The two approaches are applied to select 50/100/200 hours un-transcribed speech data respectively based on their separate criteria. The selected un-transcribed data are then added to transcribed set for acoustic model training. The performances are measured by Character Error Rate (CER). 512 clusters are used in frame level clustering, and 128 clusters are used in utterance level clustering.

7.3 Experiment II: Lightly Supervised Acoustic Model Training

This section presents the experimental results of the clustering based data selection approach in lightly supervised acoustic model training. Different from unsupervised training which handles un-transcribed speech, lightly supervised training works on closed captioned speech which transcripts is manually generated but contains considerable amount of transcription errors. Instead of using a confidence scoring model, conventionally, the correctness of transcripts is measured by comparing closed caption with decoding hypothesis via word alignment and voting. The fragments agreed by both are assumed to be correct and selected for new model training.

In conventional approach, the data selection is performed at word level. Specifically, for an utterance, the alignment between closed caption and decoding hypothesis may only agree on a fragment of the utterance rather than the whole utterance. So conventionally, an utterance, generated by audio segmenter, is further broken down into words for data selection. To increase the robustness of training, practically it often requires that the length of a selected fragment, in terms of contiguous words or phonemes, must be grater than a minimum value.

In contrast, we use utterance as the unit for our clustering based data selection due to the concern that some important information e.g. co-articulation may lose when splitting a naturally spoken utterance into words. Figure 7.4 illustrates the selection process adopted in out experiments, in which the adopted confidence metric is the character error rate calculated from the alignment of closed captioned sentence and hypothesized sentence.

Dataset and Configurations

The experiment is carried on Mandarin speech corpora. The transcribed set is *1997 Mandarin Broadcast News* which has about 30 hours speech data. The closed captioned dataset is *TDT-4 Mandarin Broadcast News* which contains about 160 hours of broadcast news and broadcast conversation speech data. This corpus also provides time stamped recognition results decoded by a third-party speech engine. The results are then used in our experiment to extract audio segmentation information. *RT-04 Eval* set is used as the testing set. We adopt 39 dimensional MFCC as the acoustic feature.

The set-up of this experiment is same to that in experiment of unsupervised acoustic model training (Section 7.2.2). The phone set contains 179 phonemes. A 3-state left-to-right HMM is adopted to model each speech unit. Phonemes are further transformed into 2000 senones, each of

which is modeled using a mixture of 64 Gaussians, giving a total of 128K Gaussians for acoustic modeling. We use the same language model described in Section 7.2.2, which has 64000 unigrams, 16.6M bigrams and 20.7M trigrams. The dictionary for testing uses the same lexicon of language model. Recognition performance is measured by character error rate.

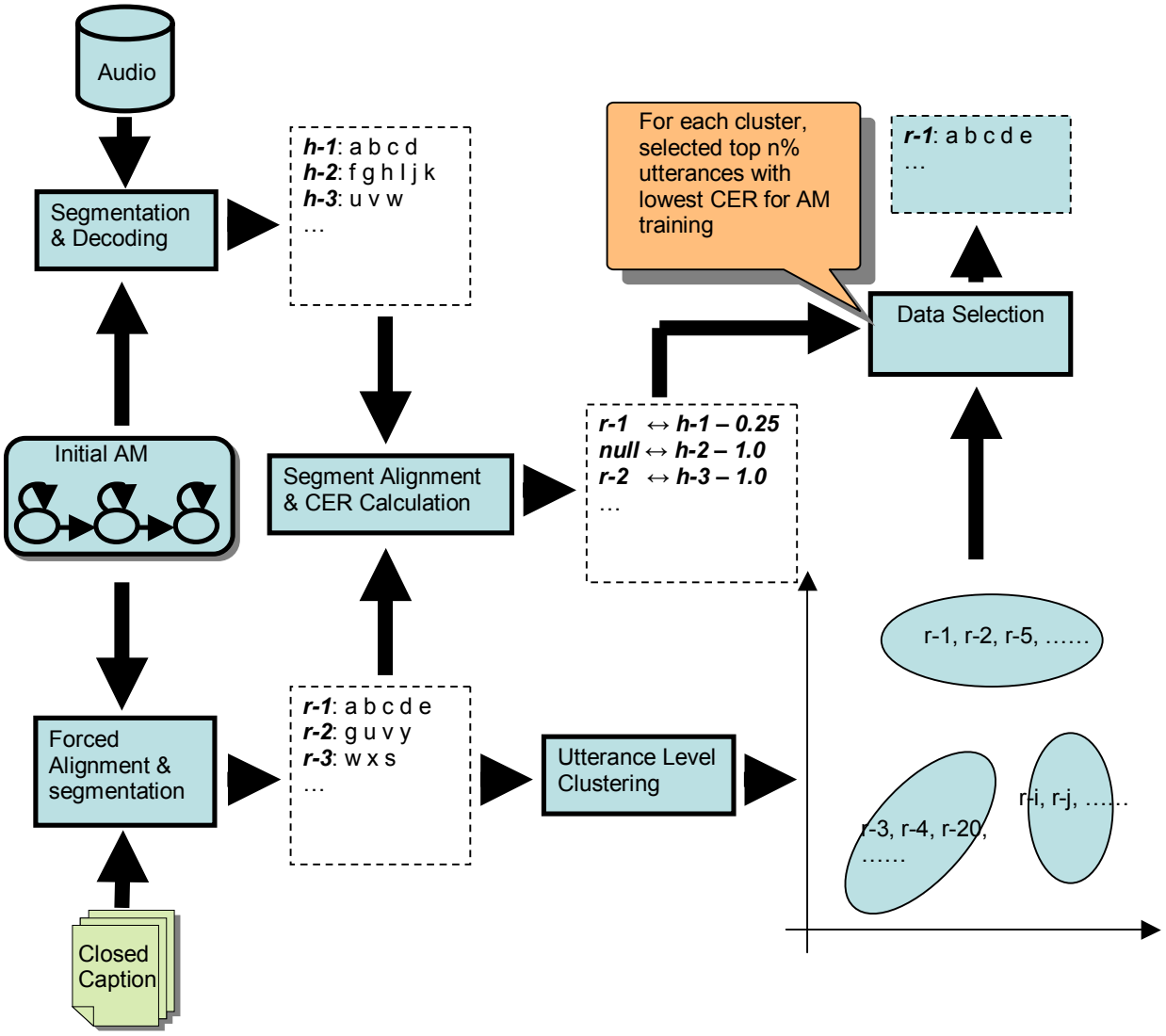

Figure 7.4 Clustering based data selection approach used in lightly supervised acoustic model training. Input audio is first segmented into utterances and decoded by initial acoustic model. On the other hand, paragraph based closed captions are also segmented into utterances by running forced alignment. We then perform an utterance to utterance alignment between closed captions and hypotheses with respect to their differences on word and time stamp. Character error rate

(CER) is calculated for each closed captioned utterance using corresponding hypothesis as the reference. Clustering based data selection is then performed, in which CER is adopted as the confidence metric to measure the correctness of closed captioned utterance.

Baseline

The baseline acoustic model is built by only using the 30 hours *1997 Mandarin Broadcast News* for training, Character error rate for this model is 29.90%.

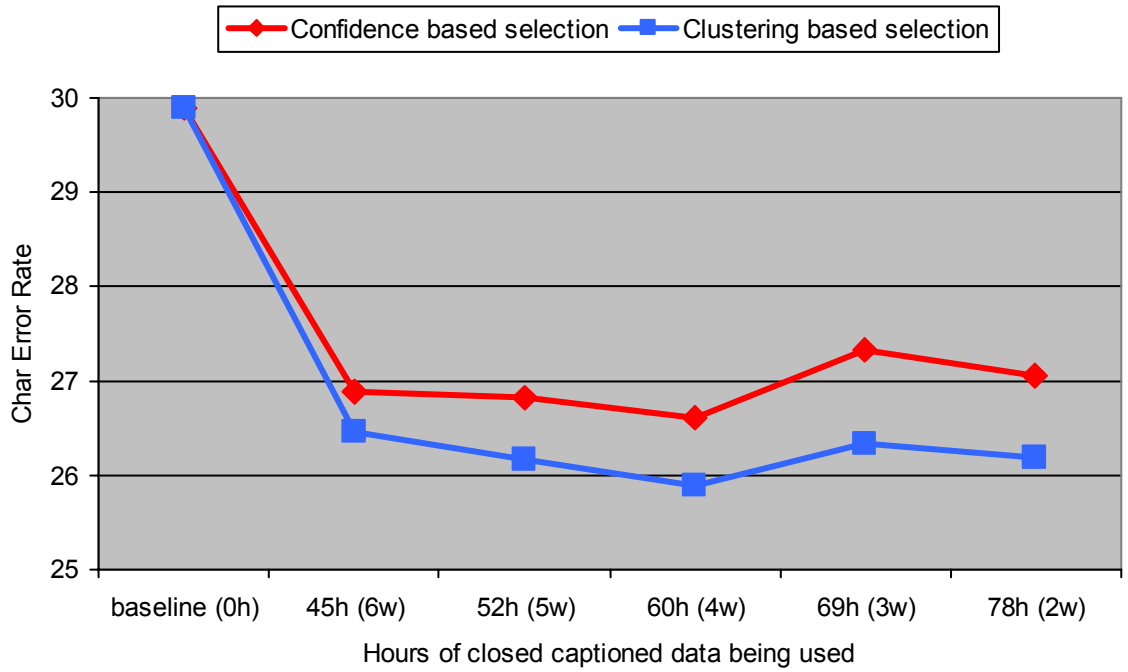

Figure 7.5 Comparison of conventional confidence based and the proposed clustering based data selection approaches in lightly supervised AM training. The performances are measured by Character Error Rate (CER). For conventional approach, we change the minimum number of words in well-matched fragments from 2 to 6. Correspondingly, the proposed approach selects and utilizes same amount of captioned data, measured in number of hours for model training. 256 clusters are used in frame level clustering, and 64 clusters are used in utterance level clustering.

Comparison of Clustering Based Data Selection with Confidence Based Data Selection

Our clustering based data selection approach is compared with conventional approach that performs selection at word level. For the fragment selected by conventional approach, we require that the length in terms of word must be greater than a pre-determined minimum value. The less

the value is, the more the number of fragments is selected. We vary the threshold from 2 words to 6 words which results in the amount of selected speech data changes from 45 hours (corresponding to 6 word) to 78 hour (corresponding to 2 word). To make a fair comparison, our clustering based data selection approach also identifies and utilizes the same amount of data for acoustic model training. The cluster number used in frame and utterance clustering are 256 and 64 respectively. Please note that our approach is conducted on utterance level and it doesn't demand that every word in the selected utterance has to be consented by both closed caption and recognition hypothesis. Figure 7.5 plots the performances of the two selection approaches.

Figure 7.5 shows that like un-transcribed data, closed captioned speech also has the potential to significantly improve the performance of acoustic modeling. When using 60 hour closed captioned speech, the character error rate is reduced by a big margin to 25.90% (using clustering based data selection) and 26.61% (using confidence based data selection) from the baseline of 29.90%. In addition, the results demonstrate again the effectiveness of clustering based data selection approach which consistently outperforms confidence based approach for all the five different training conditions. Paired t test is also conducted and results in that t=7.25 and *P-value*=0.002. Therefore, we conclude that our proposed method improves recognition accuracy significantly over traditional method on the lightly supervised acoustic model training task.

7.4 Summary

This chapter presented a novel clustering based data selection approach that considers how to identify correctly transcribed data as well as how to increase the diversity of data. We applied the approach to unsupervised training and lightly supervised training, and evaluated it with a variety of datasets. Experimental results demonstrated the effectiveness of this approach, which consistently outperforms conventional confidence scoring method in all the three speech recognition tasks.

8 Unsupervised Acoustic Model Training Based on Frame Level Boosting Algorithm

In previous chapters, we have discussed supervised frame level Boosting algorithm for utilizing transcribed speech data, and clustering based selective training strategy for identifying usable un-transcribed raw speech data. This chapter investigates if the Boosting algorithm can be extended to unsupervised training, so that the two approaches, data selection and Boosting, can be integrated together for better handling different types of training data.

We first discuss the generalization of Boosting algorithm from a supervised training method to unsupervised training method. We then evaluate the unsupervised Boosting with ICSI meeting dataset. The experimental result shows that the cooperation of data selection and unsupervised Boosting can significantly improve recognition performance.

8.1 Unsupervised Boosting Algorithm

Boosting is a supervised learning algorithm that emphasizes the "hard-to-learn" examples by increasing their weights in the training of new model, so the new model has the potential to correct the classification errors made by previous models. As shown in Chapter 4 and 5, Boosting algorithm has been demonstrated as an effective acoustic model training approach in using transcribed speech data.

The major obstacle to generalize Boosting algorithm to unsupervised training is that, for un-transcribed raw speech data, the lack of transcripts makes it intractable to determine if a recognition result is correct or not. So we are unable to follow the above criterion which pays more attention to the misrecognized utterances in new model training.

To address this problem, [Buc et al, 2001; Bennett et al, 2002] proposed an unsupervised multi-class Boosting training scheme which uses a different criterion: the unlabeled examples that the members of ensemble can't reach unanimity are given higher weights. This is essentially a confidence based approach in which the ensemble *unanimity* can be interpreted as a confidence metric to measure the correctness of classification. We further extend the method to unsupervised acoustic model training by using Minimum Bayes Risk (MBR) [Goel and Byrne, 2000; Goel et al, 2000; Goel et al, 2004] to seek the most agreed hypothesis as the reference [Zhang et al, 2005].

Suppose we have a hypothesis set $H = \{h_i \mid 1 \le i \le K\}$, where h_i is a sentence hypothesis generated as the recognition result of utterance $\mathbf{x}$. In the view of MBR, the best hypothesis is the one with the least expected error under a loss function as follows:

$$h^* = \arg\min_{h \in H} \sum_{i=1}^{K} l(h, h_i) P(h_i \mid \mathbf{x}) \tag{8-1}$$

where $l(h, h_i)$ describes a task performance oriented loss function, such as *edit* distance between sequences, and $P(h_i \mid \mathbf{x})$ denotes a posterior distribution on sentence hypothesis. The MBR can be thought of as selecting a *consensus* hypothesis: For each hypothesis, Eq. (8-1) selects the one that is closest on an average to all the likely hypothesis and alignments. The closeness is measured under the loss function of $l(h, h_i)$.

In the case of ensemble based unsupervised training, the *pseudo-transcripts* of an un-transcribed utterance can be determined by using MBR to select the one with least risk from the hypothesis set generated by the ensemble. An alternative way to determine the pseudo-transcripts is to use ROVER combination, which differs MBR in that it can create a new hypothesis rather than select one from existing hypotheses. However, as a mixture of words from different hypotheses, the result created by ROVER can not maintain the acoustic consistency owned by individual hypothesis. Moreover, ROVER is often found to increase deletion errors in order to reduce substitution and insertion errors. As we know, acoustic model training relies on an alignment between speech signal and phoneme sequence which is derived from transcripts. Too many word omissions can lead to the corruption of phoneme model learning.

The frame level unsupervised Boosting algorithm, which performs optimization and resample at frame level, is formulated in Table 8.1. Please refer to Chapter 5 for the detail of frame level Boosting algorithm and the notations used in the algorithm.

8.2 Experiment of Unsupervised Boosting Algorithm on ICSI Meeting Dataset

The unsupervised Boosting algorithm is evaluated on ICSI meeting dataset which has 75 meeting or 70 hours speech data. We follow the same set-up of data selection experiment described in Chapter 7. We use 10 meetings as the transcribed set for initial acoustic training, 61 meetings as

the un-transcribed set for unsupervised training, 3 meetings as the hold-out set for recognizer tuning, and 1 meeting as the test set.

Input:

- Ψ_T : Transcribed set that $\Psi_T = \{(\mathbf{x}_i, y_i) \mid 1 \le i \le M\}$ where $\mathbf{x}_i = (x_{i,1}, x_{i,2}, ..., x_{i,T})$ is the sequence of acoustic feature vectors for utterance *i* and y_i is the corresponding transcript.
- Ψ_U : Un-transcribed set that $\Psi_U = \{(\mathbf{x}_j, ?) \mid M+1 \le j \le N\}$.

Initialize:

- Train an initial acoustic model λ_0 from Ψ_T.
- Decode un-transcribed utterance $(\mathbf{x}_j, ?) \in \Psi_U$ using λ_0, generating hypothesis h_j so that $(\mathbf{x}_j, ?) \Rightarrow (\mathbf{x}_j, h_j)$, and then form a new set $\Psi_U^* = \{(\mathbf{x}_j, h_j) \mid M+1 \le j \le N\}$.
- Let $\Psi_0 = \Psi_T \cup \Psi_U^* = \{(\mathbf{x}_i, y_i) \mid 1 \le i \le N\}$ where $y_i = h_i$ for $M+1 \le i \le N$.

For k = 1 to K:

- Train a new acoustic model λ_k from data set Ψ_{k-1}.
- Determine the value of $r_u(i,t)$ for each frame $x_{i,t}$ of each utterance $\mathbf{x}_i \in \Psi_{k-1}$. Run forced-alignment if necessary.
- Generate hypothesis set $H_{\mathbf{x}_i} = \{h\}$ for each utterance $\mathbf{x}_i \in \Psi_{k-1}$ using λ_k, and compute posterior probability $P_{\lambda_k}(\theta_u(i,t) \mid \mathbf{x}_i)$ for every possible $\theta_u(i,t)$ at frame *t*.
- Compute pseudo loss

$$\varepsilon_k = \frac{1}{2 \mid \Psi_{k-1} \mid} \sum_{\mathbf{x}_i \in \Psi_{k-1}} \frac{1}{T_i} \sum_{t=1}^{T_i} \frac{l_{i,t}}{\mid \theta_u(i,t) \mid}$$

  where

$$l_{i,t} = \sum_{\theta_u(i,t) \neq r_u(i,t)} [1 - P_{\lambda_k}(r_u(i,t) \mid \mathbf{x}_i) + P_{\lambda_k}(\theta_u(i,t) \mid \mathbf{x}_i)].$$

- Set $c_k = \varepsilon_k / (1 - \varepsilon_k)$
- Calculate new weight for each frame *t* of each utterance $\mathbf{x}_i \in \Psi_{k-1}$ that

$$w_{i,t} = \frac{1}{\mid \theta_u(i,t) \mid} \sum_{\theta_u(i,t) \neq r_u(i,t)} c_k^{\frac{1}{2}[1 + P_{\lambda_k}(r_u(i,t) \mid \mathbf{x}_i) - P_{\lambda_k}(\theta_u(i,t) \mid \mathbf{x}_i)]}$$

- Resample training data according to normalized $w_{i,t}$, forming new training set Ψ_k. For un-transcribed utterance $\mathbf{x}_i \in \Psi_U$, use MBR (Eq. 8-1) to set the pseudo-transcripts y_i with the hypothesis most agreed among existing ensemble $\{\lambda_1, \lambda_2, ..., \lambda_k\}$.

In generalization:

- The hypothesis to a new utterance $\mathbf{x}$ is determined by using ROVER to combine the sentence hypotheses generated by the *K* acoustic models.

Table 8.1 Unsupervised frame level Boosting algorithm for using both transcribed and un-transcribed speech data

Baseline

An initial acoustic model is trained solely from the 10 transcribed meetings (Model 1 in Table 8.2). All the un-transcribed speech data are decoded by using the initial model. Clustering based data selection algorithm described in Chapter 7 is then performed to identify the most usable un-transcribed data. According to the experiment shown in Section 7.2.1, Chapter 7, the best performance is achieved by adding 33 hours speech data to the training (Model 2 in Table 8.2). The unsupervised Boosting is then carried out on the combination of transcribed speech and 33 hours selected un-transcribed speech. In addition, we also trained an "oracle" model using the correct transcripts of all speech data. The transcripts are manually generated by human transcriber and double checked to maintain quality. In contrast, in the training of model 1 and model 2, the correct transcripts are only available to a small portion of the speech data, and decoding hypotheses, which unavoidably contain recognition errors, are adopted as the transcripts for un-transcribed part. Thus it is not surprising to see the oracle model performs significantly better than model 1 and model 2. The oracle model is used in our experiment as an indicator to show us the best performance that unsupervised acoustic model training can realize. Table 8.2 shows the word error rates of these three models.

Acoustic Model Trained from	Word Error Rate
Model 1: Transcribed Data	47.31%
Model 2: Transcribed + Selected Un-transcribed	43.18%
Model 3: *oracle*	30.69%

Table 8.2 Performances of baseline systems for unsupervised Boosting training. The first one is trained using transcribed data only. Clustering based data selection approach is performed to selected 33h un-transcribed speech for augmenting transcribed data. The second model is then trained from the combination of transcribed data and selected un-transcribed data. The third model is trained using correct transcripts of all speech data.

Performance of Unsupervised Boosting

The unsupervised Boosting training is carried out on the basis of clustering based data selection. The unit $\boldsymbol{u}$ in $r_u(i,t)$ and $\theta_u(i,t)$ is set to context independent phoneme as in the experiment of

supervised Boosting training. A total number of 9 models are trained using the algorithm shown in Table 8.1. Table 8.3 presents the word error rates of the algorithm varying with the ensemble size from 1 to 9. ROVER is used to combine the sentence hypothesis predicted by each model to form the final system output.

Size	N=1	N=2	N=3	N=4	N=5	N=6	N=7	N=8	N=9
WER	43.18%	41.85%	40.38%	39.91%	39.40%	39.00%	38.67%	38.42%	38.47%

Table 8.3 Performance of unsupervised Boosting algorithm on ICSI meeting dataset

Table 8.3 shows that unsupervised Boosting algorithm demonstrates significant improvements over the baseline. Word error rate is down to 38.42% from 43.18% when 8 acoustic models are generated and combined, which represents an 11.0% relative reduction on the recognition errors. If compared to the word error rate of 47.31%, the performance of the model solely trained on transcribed data, the relative reduction is 18.8%. The encouraging results manifest the potential of the unsupervised Boosting as a useful method for improving the quality of acoustic model training by exploiting both transcribed and un-transcribed data. However, comparison to the oracle model trained using correct transcripts shows that the performance of unsupervised training still lags far behind. This suggests further research on unsupervised learning technique is highly desired.

8.3 Summary

This chapter investigated the generalization of supervised Boosting algorithm to unsupervised acoustic modeling, as well as its combination with clustering based data selection strategy for handling different types of training data. Significant improvement of recognition performance was observed in experiments with ICSI meeting dataset. On the basis of data selection, unsupervised Boosting reduced the word error rate by relatively 18.8%.

9 Summary and Conclusions

This thesis presents our research on investigating approaches that utilize the characteristics of various speech data to improve the performance of speech recognition systems. We focus on how to effectively exploit available transcribed data and how to explore informative un-transcribed data for acoustic model training. In the remainder of this chapter, we summarize the major findings of this thesis and discuss open questions and promising directions for further research.

9.1 Major Contributions

Frame Level Boosting Algorithm

We investigate Boosting algorithm, an ensemble based supervised training approach, to iteratively create multiple acoustic models with complementary error patterns by updating the distribution of training data. We proposed a frame level Boosting algorithm. In contrast to conventional approach based on a sentence level objective function, our approach aims at the minimization of word or sub-word recognition errors which is more effective to improve training performance. Moreover, our approach enables acoustic model training more focus on the misrecognized part of an utterance rather than, as adopted by conventional approach, giving the whole utterance an equal weight without discriminating recognition error from correct result. Substantial improvement of recognition performance is obtained in our experiments with CMU Communicator dataset. Compared to single acoustic model system, our approach reduces the word error rate by relatively 22%. Compared to conventional utterance level Boosting, our approach reduces the word error rate by relatively 6%.

Improved Hypothesis Combination Scheme

We proposed a Neural Network based combination scheme which improves standard hypothesis combination in two perspectives: 1. N-best list re-ranking is investigated to identify more reliable recognition hypothesis as the input of combination; 2. the desired word is determined by a two-stage scheme considering insertion detection and word scoring separately, and Neural Network is employed to incorporate a number of informative features into the decision process. In contrast, standard ROVER only uses a simple voting strategy that linearly combines two features: frequency of occurrence and confidence score. Considerable improvement is obtained in our

experiments with CMU Communicator dataset. Compared to standard ROVER, our approach reduces the word error rate by relatively 4%. If performed with frame level Boosting, the overall reduction of word error rate is relatively 25%.

Clustering Based Data Selection Strategy

Conventional data selection approaches only focus on how to ensure if the selected examples have correct transcripts, while ignore other issues. These strategies often result in only the examples that match well to the current model being selected and thus lead to a suboptimal model. To address the problem, we proposed a novel data selection strategy that considers two requirements together: 1. the correctness of transcripts; 2. the diversity of selected data i.e. its coverage to acoustic and phonetic phenomenon. We applied the strategy to both unsupervised and lightly supervised acoustic model training by implementing an utterance space clustering scheme which helps to automatically capture the variability and complexity of human speech. Consistent improvements are obtained in our experiments on English and Mandarin speech recognitions with un-transcribed and closed captioned speech corpora including ICSI meeting dataset, GALE BN03 un-transcribed dataset and TDT-4 closed captioned dataset.

Unsupervised Boosting Algorithm in Acoustic Model Training

Traditionally, Boosting is a supervised training approach that requires the class label to be given for each example. We investigate the generalization of Boosting algorithm to unsupervised acoustic model training. To do this, Minimum Bayes Risk (MBR) decoding procedure is used to determine the pseudo-transcripts for each un-transcribed utterance by selecting the hypothesis with the least expected error. On the basis of MBR, a frame level unsupervised Boosting training algorithm is proposed, and further combined with clustering based data selection approach to form a unified framework for better handling different types of training data. Significant improvement of recognition performance is observed in our experiments on ICSI meeting dataset.

9.2 Some Further Directions

While the algorithms developed in this thesis have been quite successful at improving speech recognition performance, there is still room for additional improvement.

In our frame level Boosting algorithm, the sub-word hypothesis for each frame is estimated by a Viterbi based forced alignment program. We noticed that the performance of Boosting training is impacted considerably by the result of forced alignment. Therefore, investigation on a robust forced alignment algorithm is highly desirable.

In addition, the sampling method used in frame level Boosting algorithm is exclusively depends on the weight calculated in the training process. As we have known, there is other information helpful for sampling, such as the type and duration distribution of a phoneme. For example, to increase the importance of consonant, we could lengthen the duration of consonant while shorten the duration of vowel.

The conversion of an utterance to a fixed size vector is the key issue for performing clustering based data selection. Please note an utterance can carry multiple information, e.g. acoustic info, phonetic info, semantic info, speaker info, etc.. Currently, we only exploit acoustic and phonetic information. We believe the proposed data selection approach can further benefit from the incorporation of other useful information.

K-Means is adopted in our experiments as the major clustering algorithm. This is equal to assume the distribution of utterance vectors follow the distribution of Gaussian Mixtures. Apparently, this is a strong assumption. There exist other options such as bottom-up hierarchical clustering techniques which are worthy to be investigated.

References

[Asami etal, 2005] Asami, T., Iwano, K., and Furui, S., "Stream-Weight Optimization by LDA and Adaboost for Multi-Stream Speaker Verification", Proc. of European Conference on Speech Communication and Technology, 2005.

[Bahl et al, 1980] Bahl, L.R., Bakis, R., Cohen, P.S., Cole, A.G., Jelinek, F., Lewis, B.L. and Mercer, R.L., "Further Results on the Recognition of a Continuously Read Natural Corpus", Proc. of IEEE International Conference on Acoustics, Speech, and Signal Processing, 1980.

[Banerjee et al, 2004] Banerjee, S., Cohen, J., Quisel, T., Chan, A., Patodia, Y., Bawab, Z. A., Zhang, R., Black, A., Stern, R., Rosenfeld, R., Rudnicky, A. I., Rybski, P., and Veloso, A., "Creating Multi-Modal, User-Centric Records of Meetings with the Carnegie Mellon Meeting Recorder Architecture", Proc. of ICASSP 2004 Meeting Recognition Workshop.

[Baum, 1972] Baum, L., "An Inequality and Associated Maximization Technique in Statistical Estimation of Probabilistic Functions of Markov Processes", Inequalities, v. 3,p. 1-8, 1972.

[Bennett et al, 2002] Bennett, K. P., Demiriz, A., and Maclin, R., "Exploiting Unlabeled Data in Ensemble Methods", Proc. of SIGKDD International Conference on Knowledge Discovery and Data Mining, 2002.

[Blum et al, 1998] Blum, A., and Mitchell, T., "Combining Labeled and Unla-beled Data with Co-Training", Proc. of the 11th Conference on Computational Learning Theory, 1998.

[Bourlard et al., 1996] Bourlard, H., Dupont, S., Hermansky, H., and Morgan, N., "Towards Sub-Band Based Speech Recognition", Proc. of European Signal Processing Conference, pages 1579-1582, 1996.

[Bourlard and Dupont, 1997] Bourlard, H., and Dupont, S., "Sub-Band Based Speech Recognition", Proc. of IEEE International Conference on Acoustics Speech and Signal Processing, pages 1251-1254, 1997.

[Breiman, 1996] Breiman, L., "Bagging Predictors", Machine Learning, 24(2): 123-140, 1996.

[Breiman, 1998] Breiman, L., "Arcing Classifiers", The Annals of Statistics, 26(3): 801-849, 1998.

[Breiman, 2001] Breiman, L., "Random Forests," Machine Learning, vol. 45, no.1, pp. 5–32, 2001.

[Buc et al, 2001] Buc, F. D., Grandvalet Y., and Ambroise, C., "Semi-Supervised MarginBoost", Proc. of 9th Conference on Neural Information Processing Systems (NIPS), 2001.

[Carpenter et al, 2001] Carpenter, P., Jin, C., Wilson, D., Zhang, R., Bohus D., and Rudnicky, A. I., "Is This Conversation on Track?", Proc. of European Signal Processing Conference, 2001.

[Chase, 1997] Chase, L., "Error-Responsive Feedback Mechanisms for Speech Recognition", Ph.D. Thesis, Carnegie Mellon University, April 1997 .

[Chen et al, 2004] Chen, L., Lamel, L. Gauvain, J. L., "Lightly Supervised Acoustic Model Training Using Consensus Networks", Proc. of IEEE International Conference on Acoustics Speech and Signal Processing, 2004.

[Chien et al, 1997] Chien, J. T., Lee C. H., and Wang, H. C., "A hybrid Algorithm for Speaker Adaptation Using MAP Transformation and Adaptation," IEEE Signal Processing Letters, vol. 4, no. 6, p. 167-169, June 1997.

[Christensen et al, 2000] Christensen, H., Lindberg, B., and Andersen, O., "Employing Heterogeneous Information in a Multi-Stream Framework, Proc. of IEEE International Conference on Acoustics Speech and Signal Processing, 2000.

[Clarkson and Rosenfeld, 1997] Clarkson, P. R. and Rosenfeld, R., "Statistical Language Modeling Using the CMU-Cambridge Toolkit", Proc. of European Conference on Speech Communication and Technology, 1997.

[Collins et al., 2000] Collins, M., Schapire, R. E., and Singer, Y., "Logistic Regression, AdaBoost and Bregman Distances", Proc. of 13th Annual Conference on Computational Learning Theory, 2000.

[Cook and Robinson, 1995] Cook, G. D., and Robinson, A. J., "Utterance Clustering for Large Vocabulary Continuous Speech Recognition", Proc. of European Conference on Speech Communication and Technology, 1995.

[Cook and Robinson, 1996] Cook, G., and Robinson, T., "Boosting the Performance of Connectionist Large Vocabulary Speech Recognition", Proc. of International Conference on Spoken Language Processing, 1996.

[Cook et al, 1997] Cook, G., Waterhouse, S., and Robinson, A., "Ensemble Methods for Connectionist Acoustic Modeling", Proc. of European Conference on Speech Communication and Technology, 1997.

[Cozman et al, 2003] Cozman, F. G., Cohen, I., and Cirelo, M. C., "Semi-Supervised Learning of Mixture Models and Bayesian Networks", Proc. of 20th International conference on Machine Learning, 2003.

[Davis, S. and Mermelstein, 1980] Davis, S. and Mermelstein, P., "Comparisons of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences", IEEE Transactions on Acoustics, Speech and Signal Processing, 28(4):357-366, 1980.

[Dietterich and Bakiri, 1995] Dietterich, T. G. and Bakiri, G., "Solving Multiclass Learning Problems via Error-Correcting Output Codes", Journal of Artificial Intelligence Research, 2, 263-286, 1995.

[Dietterich, 1998] Dietterich, T. G., "Machine Learning Research: Four Current Directions", AI Magazine, 18(4): 97-136, 1998.

[Dietterich, 2000] Dietterich, T., "An Experimental Comparison of Three Methods for Constructing Ensembles of Decision Trees: Bagging, Boosting, and Randomization", Machine Learning, 40, 139–157, 2000.

[Dimitrakakis and Bengio, 2004] Mimitrakakis, C., and Bengio, S., "Boosting HMMs with an Application to Speech Recognition", Proc. of IEEE International Conference on Acoustics Speech and Signal Processing, 2004.

[Dimitrakakis and Bengio, 2004] Mimitrakakis, C., and Bengio, S., "Boosting Word Error Rates", Proc. of IEEE International Conference on Acoustics Speech and Signal Processing, 2005.

[Dupont and Ris, 2001] Dupont, S., Ris, C., "Multiband with Contaminated Training Data", Proc. of CRAC Workshop, European Conference on Speech Communication and Technology, 2001.

[Fiscus, 1997] Fiscus, J. G., "A Post-Processing System to Yield Reduced Word Error Rates: Recognizer Output Voting Error Reduction (ROVER)", Proc. of IEEE Automatic Speech Recognition and Understanding Workshop Workshop, 1997.

[Fletcher, 1953] Fletcher, H., "Speech and Hearing in Communication", Krieger, New York.

[Foo and Lim, 2002] Foo, S. W., and Lim, E., G., "Speaker Recognition Using Adaptively Boosted Decision Tree Classifier", Proc. of International Conference on Acoustics Speech and Signal Processing, 2002.

[Freund and Schapire, 1996] Freund, Y. and Schapire, R. E., "Experiments with a New Boosting Algorithm," Proceedings of the 13th International Conference on Machine Learning, 148-156, 1996.

[Freund and Schapire, 1997] Freund, Y. and Schapire, R. E., "A Decision Theoretic Generalization of On-line Learning and an Application to Boosting", Journal of Computer and System Science, 55(1): 119-139, 1997.

[Goel and Byrne, 2000] Goel, V., and Byrne, W. J., "Minimum Bayes-Risk Automatic Speech Recognition", Computer Speech and Language, Vol. 14(2), pp. 115--135, 2000.

[Goel et al, 2000] Goel, V., Kumar, S., and Byrne, W., "Segmental Minimum Bayes-Risk ASR Voting Strategies", Proc. of International Conference on Spoken Language Processing, 2000.

[Goel et al, 2004] Goel, V., Kumar, S., and Byrne, W., "Segmental Minimum Bayes-Risk Decoding for Automatic Speech Recognition", IEEE Transactions on Speech and Audio Processing, Vol 12(3), 234-250, 2004.

[Hagen et al, 1998] Hagen, A., Morris, A., and Bourlard, H., "Subband-Based Speech Recognition in Noisy Conditions: the Full Combination Approach", IDIAP Research Report 98-15, 1998.

[Hagen et al, 2000], Hagen, A., Morris, A., Bourlard, H., "From Multi-Band Full combination to Multi-Stream Full Combination Processing in Robust ASR", Proc. of ISCA Tutorial and Research Workshop ASR2000.

[Hagen and Bourlard, 2000] Hagen, A., and Bourlard, H., "Using Multiple Time Scales in the Framework of Multi-Stream Speech Recognition", Proc. of International Conference on Spoken Language Processing, 2000.

[Hagen et al., 2001] Hagen, A., Bourlard, H., and Morris, A., "Adaptive ML-Weighting in Multi-Band Recombination of Gaussian Mixture ASR", Proc. of IEEE International Conference on Acoustics Speech and Signal Processing, 2001.

[Hagen and Bourlard, 2001] Hagen, A., and Bourlard, H., "Error Correcting Posterior Combination for Robust Multi-Band Speech Recognition", IDIAP Research Report 01-10, 2001.

[Hermansky, 1990] Hermansky, H., "Perceptual Linear Predictive (PLP) Analysis of Speech", Journal of the Acoustical Society of America, 87(4):1738-1752, 1990.

[Ho, 1995] Ho, T. K., "Random Decision Forests", Proceedings of the 3rd International Conference on Document Analysis and Recognition, Montreal, 278-282, 1995.

[Ho, 1998] Ho, T. K., "The Random Subspace Method for Constructing Decision Forests", IEEE Trans. On Pattern Analysis and Machine Intelligence, 20(8): 832–844, 1998.

[Huang et al, 1993] Huang, X., Alleva, F., Hon, H., Hwang, M., Lee, K., Rosenfeld, R., "The SPHINX-II Speech Recognition System: An Overview", Computer Speech and Language, v. 2, p. 137-148, 1993.

[Hwang, 1993] Hwang, M. Y., "Subphonetic Acoustic Modeling for Speaker-Independent Continuous Speech Recognition", Ph.D. thesis, Carnegie Mellon University, 1993.

[Kamm and Meyer, 2001] Kamm, T. M., and Meyer, G. L., "Automatic Selection of Transcribed Training Material", Proc. of IEEE Automatic Speech Recognition and Understanding Workshop, 2001.

[Kemp and Waibel, 1999] Kemp, T., and Waibel, A., "Unsupervised Training of a Speech Recognizer: Recent Experiments", Proc. of European Conference on Speech Communication and Technology, 1999.

[Kirchhoff and Bilmes, 2000] Kirchhoff, K., and Bilmes, J., "Combination and Joint Training of Acoustic Classifiers for Speech Recognition", Proc. of ISCA Tutorial and Research Workshop ASR2000.

[Kirchhoff et al, 2000] Kirchhoff, K., Fink, G. A., and Sagerer, G., "Conversational Speech Recognition Using Acoustic and Articulatory Input", Proc. of International Conference on Acoustics Speech and Signal Processing, 2000.

[Kuncheva et al., 2002] Kuncheva, L. I., Skurichina, M., and Duin, R. P. W., "An Experimental Study on Diversity for Bagging and Boosting with Linear Classifiers", Information Fusion, 3 (2), 245-258, 2002.

[Kuncheva & Whitaker, 2003] Kuncheva, L. I. and Whitaker, C. J., "Measures of Diversity in Classifier Ensembles", Machine Learning , 51 , 181-207, 2003.

[Kwon and Lee, 2003] Kwon, O. W., and Lee, T. W., "Optimizing Speech/Non-speech Classifier Design Using AdaBoost", Proc. of International Conference on Acoustics Speech and Signal Processing, 2003.

[Jang and Hauptmann, 1999a] Jang, P. J., Hauptmann, A. G., "Improving Acoustic Models with Captioned Multimedia Speech". Proc. of ICMCS, Vol. 2, pp 767-771, 1999.

[Jang and Hauptmann, 1999b] Jang, P. J., Hauptmann, A. G., "Learning to Recognize Speech by Watching Television", Proc. of IEEE Intelligent Systems, Volume 14, No. 5, pp. 51 - 58, 1999.

[Janin et al, 1999] Janin, A., Ellis, D., Morgan, N., "Multi-Stream Speech Recognition: Ready for Prime Time?", Proc. of European Conference on Speech Communication and Technology, 1999.

[Janin et al, 2003] Janin, A., Baron, D., Edwards, J. A., Ellis, D., Gelbart, D., Morgan, N., Peskin, B., Pfau, T., Shriberg, E., Stolcke, A., and Wooters, C., "The ICSI meeting corpus". Proc. of International Conference on Acoustics, Speech, and Signal Processing, 2003.

[Jelinek, 1998] Jelinek, F., "Statistical Methods for Speech Recognition", MIT Press, 1998.

[Jin and Schultz, 2004] Jin, Q. and Schultz, T., "Speaker Segmentation and Clustering in Meetings". Proc. of International Conference of Spoken Language Processing, 2004.

[Juang and Katagiri, 1992] Juang, B. H., and Katagiri, S., "Discriminative Learning for Minimum Error Classification", IEEE trans. on Signal Processing, Vol. 40, No. 12, 1992.

[Lamel et al, 2000] Lamel, L., Gauvain, J. L., Adda, G., "Lightly Supervised Acoustic Model Training," Proc. ISCA ITRW ASR2000, pp. 150-154, 2000.

[Lamel et al, 2002] Lamel, L., Gauvain J., and Adda, G., "Unsupervised Acoustic Model Training", Proc. of International Conference on Acoustics Speech and Signal Processing, 2002.

[Lee, 1988] Lee, K., "Large Vocabulary Speaker-Independent Continuous Speech Recognition: The SPHINX System". Ph.D. Thesis, Carnegie Mellon University, 1988.

[Lee, 1990] Lee, K., "Context-Dependent Phonetic Hidden Markov Models for Continuous Speech Recognition". Proc. of IEEE Transactions on Acoustics, Speech, and Signal Processing, pp 599-609, 1990.

[Li et al, 2002] Li, X., Singh, R., Stern, R. M., "Lattice Combination for Improved Speech Recognition", Proc. of International Conference on Spoken Language Processing, 2002.

[Li et al, 2003] Li, S., Z., Zhang, D., Ma, C., Shum, H. Y., and Chang, E., "Learning to Boost GMM Based Speaker Verification", Proc. of European Conference on Speech Communication and Technology, 2003.

[Liscombe et al, 2005] Liscombe, J., Riccardi, G., Tur, D. H., "Using Context to Improve Emotion Detection in Spoken Dialog Systems", Proc. of European Conference on Speech Communication and Technology, 2005.

[Ma and Matsoukas, 2007] Ma, J. and Matsoukas, S., "Unsupervised Training on a Large Amount of Arabic Broadcast News Data", Proc. of International Conference on Acoustics Speech and Signal Processing, 2007.

[Mairesse and Walker, 2005] Mairesse, F., and Walker, M., "Learning to Personalize Spoken Generation for Dialogue Systems", Proc. of European Conference on Speech Communication and Technology, 2005.

[Mangu et al, 2000] Mangu, L., Brill, E., and Stolcke, A., "Finding Consensus in Speech Recognition: Word Error Minimization and Other Applications of Confusion Networks", Computer, Speech and Language, 14(4):373-400, 2000.

[Mason et al., 1999] Mason, L., Baxter, J., Bartlett, P. L., and Frean, M., "Boosting Algorithms as Gradient Descent in Function Space". Technical Report, RSISE, Australian National University, 1999.

[Meyer, 2002] Meyer, C., "Utterance-Level Boosting of HMM Speech Recognizers", Proc. of International Conference on Acoustics Speech and Signal Processing, 2002.

[Moreno et al, 2001] Moreno, P. J., Logan, B., and Raj, B., "A Boosting Approach for Confidence Scoring", Proc. of European Conference on Speech Communication and Technology, 2001.

[Moreno et al, 2003] Moreno, P. J., and Agarwal, S., "An Experimental Study of EM-based Algorithms for Semi-Supervised Learning in Audio Classifi-cation", Proc. of ICML-2003 Workshop on Continuum from La-beled to Unlabeled Data, 2003.

[Neto and Meinedo, 2000] Neto, J. P., and Meinedo, H., "Combination of Acoustic Models in Continuous Speech Recognition Hybrid Systems", Proc. of International Conference on Spoken Language Processing, 2000.

[Ney, 1990] Ney, H., "Acoustic Modeling of Phoneme Units for Continuous Speech Recognition", Proc. of European Signal Processing Conference, 1990.

[Nguyen and Xiang, 2004] Nguyen, L. and Xiang, B., "Light Supervision in Acoustic Model Training", Proc. of International Conference on Acoustics Speech and Signal Processing, 2004.

[Nigam et al, 2000] Nigam, K., McCallum, A. K., Thrun, S., Mitchell, T., "Text Classification from Labeled and Unlabeled Docu-ments using EM", Machine Learning, 39(2/3): 103-134 2000.

[Rabiner, 1989] Rabiner, L. R., "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition", Proceedings of the IEEE 77(2):257-286, 1989.

[Rabiner and Juang, 1993] Rabiner, L. and Juang, B., "Fundamentals of Speech Recognition", Prentice Hall Signal Processing Series, Englewood Cliffs, 1993.

[Ravishankar, 1996] Ravishankar, M., "Efficient Algorithms for Speech Recognition", Ph.D Thesis, Carnegie Mellon University, 1996.

[Riccardi and Hakkani-Tur, 2003] Riccardi, G. and Hakkani-Tur, D., "Active and unsupervised learning for automatic speech recognition", Proc. of the European Conference on Speech Communication and Technology, 2003.

[Rochery et al, 2002] Rochery, M., Schapire, R., Rahim, M., Gupta, N., Riccardi, G., Bangalore, S., Alshawi, H., and Douglas, S., "Combining Prior Knowledge and Boosting for Call Classification in Spoken Language Dialogue", Proc. of International Conference on Acoustics Speech and Signal Processing, 2002.

[Rosenfeld, 2000] Rosenfeld, R., "Two Decades of Statistical Language Modeling: Where Do We Go from Here", Proc. of IEEE, vol. 88, no. 8, pp. 1270-1278, Aug. 2000.

[Rudnicky et al, 1999] Rudnicky, A. I., Thayer, E., Constantinides, P., Tchou, C., Shern, R., Lenzo, K., Xu, W., Oh, A., "Creating Natural Dialogs in the Carnegie Mellon Communicator System", Proc. of European Conference on Speech Communication and Technology, 1999.

[Schapire et al., 1997] Schapire, R. E., Freund, Y., Bartlett, P., and Lee, W. S., "Boosting the Margin: A New Explanation for the Effectiveness of Voting Methods", Proceedings of the 14th International Conference on Machine Learning, 1997.

[Schapire, 1999] Schapire, R. E., "A brief Introduction to Boosting", Proc. of the 16th International Joint Conference on Artificial Intelligence, 1999.

[Schapire and Singer, 1999] Schapire, R. E. and Singer, Y., "Improved Boosting Algorithms Using Confidence Rated Predictions", Machine Learning, 37(3): 297-336, 1999.

[Schluter, 2000] Schluter, R., "Investigations on Discriminative Training Criteria", Ph.D. Thesis, Aachen, Germany, 2000.

[Schluter et al, 2001] Schluter, R., Macherey, W., Muller, B., Ney, H., "Comparison of Discriminative Training Criteria and Optimization Methods for Speech Recognition". Speech Communication, Vol. 34, pp. 287-310, 2001.

[Schwenk, 1999] Schwenk, H., "Using Boosting to Improve a Hybrid HMM Neural Network Speech Recognizer", Proc. of International Conference on Acoustics Speech and Signal Processing, 1999.

[Schwenk and Gauvain, 2000] Schwenk, H., and Gauvain, J. L., "Combining Multiple Speech Recognizers using Voting and Language Model Information", Proc. of International Conference on Speech and Language Processing, 2000.

[Sharma, 1999] Sharma, S. R., "Multi-Stream Approach to Robust Speech Recognition", Ph.D. Thesis, OGI, 1999.

[Shipp and Kuncheva, 2002] Shipp, C. A. and Kuncheva, L. I., "Relationships between Combination Methods and Measures of Diversity in Combining Classifiers", Information Fusion, 3 (2), 135-148, 2002.

[Shire, 2000] Shire, M. L., "Discriminant Training of Front-End and Acoustic Modeling Stages to Heterogeneous Acoustic Environments for Multi-stream Automatic Speech Recognition", Ph.D. Thesis, University of California, Berkeley, 2000.

[Siegler et al, 1997] Siegler, M., Jain, U., Raj, B., and Stern, R. M., "Automatic Segmentation, Classification and Clustering of Broadcast News Audio," Proc. DARPA Speech Recognition Workshop, Feb. 1997.

[Singh et al, 2001] Singh, R., Seltzer, M., Raj, B., and Stern, R. M., "Speech in Noisy Environments: Robust Automatic Segmentation, Feature Extraction, and Hypothesis Combination", Proc. of International Conference on Acoustics Speech and Signal Processing, 2001.

[Siohan et al, 2005] Siohan, O., Ramabhadran, B., and Kingsbury, B., "Constructing Ensembles of ASR Systems Using Randomized Decision Trees", Proc. of International Conference on Acoustics Speech and Signal Processing, 2005.

[Skurichina and Duin, 2002] Skurichina, M. and Duin, R. P. W., "Bagging, Boosting and the Random Subspace Method for Linear Classifiers", Pattern Analysis & Applications, 5:121–135, 2002.

[Stolfo et al, 1989] Stolfo, S. J., Galil, Z., McKeown, K., and Mills, R., "Speech Recognition in Parallel", Proc. of Speech Natural Language Workshop, DARPA, pp 353-373, 1989.

[Therrien 1992] Therrien, C. W., "Discrete Random Signals and Statistical Signal Processing", Prentice Hall Inc., New Jersey, 1992.

[Thompson et al, 1994] Thompson, J. D., Higgins, D. G., and Gibson, T. J., "CLUSTAL W: Iimproving the Sensitivity of Progressive Multiple Sequence Alignment through Sequence Weighting, Position-Specific Gap Penalties and Weight Matrix Choice", Nucleic Acids Research, 1994 Nov 11, 22(22):4673-80.

[Tibrewala and Hermansky, 1997] Tibrewala, S., and Hermansky, H., "Sub-Band Based Recognition of Noisy Speech", Proc. of IEEE International Conference on Acoustics Speech and Signal Processing, pages 1255-1258, 1997.

[Tur et al, 2004] Tur, G., Tur, D. H., Riccardi, G., "Extending Boosting for Call Classification Using Word Confusion Networks", Proc. of International Conference on Acoustics Speech and Signal Processing, 2004.

[Tur, 2005] Tur, G., "Model Adaptation for Spoken Language Understanding", Proc. of International Conference on Acoustics Speech and Signal Processing, 2005.

[Vergyri et al, 2000] Vergyri, D., Tsakalidis, S., and Byrne, W., "Minimum Risk Acoustic Clustering for Multilingual Acoustic Model combination", Proc. of International Conference on Spoken Language Processing, 2000.

[Visweswariah et al, 2004] Visweswariah, K., Gopinath, R., and Goel, V., "Task Adaptation of Acoustic and Language Models Based on Large Quantities of Data", Proc. of International Conference on Spoken Language Processing, 2004.

[Viterbi, 1967] Viterbi, A., "Error Bounds for Convolutional Codes and an Asymptotically Optimum Decoding Algorithm", IEEE Transactions on Information Theory, v. IT-13, p. 260-269, 1967.

[Walker et al, 2003] Walker, M., Prasad R., and Stent, A., "A Trainable Generator for Recommendations in Multimodal Dialog", Proc. of European Conference on Speech Communication and Technology, 2003.

[Wang et al, 2003] Wang, D., Lu, L., Zhang, H. J., "Speech Segmentation without Speech Recognition", Proc. of International Conference on Acoustics Speech and Signal Processing, 2003.

[Weber, 2000] Weber, K., "Multiple Timescale Feature Combination towards Robust Speech Recognition", Proc. of KONVENS 2000, pp. 295-299.

[Wessel et al, 1998] Wessel, F., Macherey K., and Schluter, R., "Using Word Probabilities as Confidence Measures". Proc. of International Conference on Acoustics Speech and Signal Processing, Vol. 1, pp. 225-228, 1998.

[Wessel and Ney, 2001] Wessel, F., and Ney, H., "Unsupervised Training of Acoustic Modeling for Large Vocabulary Continuous Speech Recognition", Proc. of IEEE Automatic Speech Recognition and Understanding Workshop, 2001.

[Macherey et al, 2005] Macherey, W., Haferkamp, L., Schl¨uter R., and Ney H., "Investigations on Error Minimizing Training Criteria for Discriminative Training in Automatic Speech Recognition", Proc. of European Conference on Speech Communication and Technology, 2005.

[Wu et al, 1998] Wu, S. L., Kingsbury, B. E. D., Morgan, N., and Greenberg, S., "Incorporting Information from Syllable-Length Time Scales into Automatic Speech Recognition", Proc. of International Conference on Acoustics Speech and Signal Processing, 1998.

[Xiong and Huang, 2002] Xiong, Z., and Huang, T., "Boosting Speech/Non-Speech Classification Using Averaged Mel-frequency Cepstrum", Proc. of The 3rd IEEE Pacific-Rim Conference on Multimedia, 2002.

[Zhang and Rudnicky, 2001] Zhang, R., and Rudnicky, A. I., "Word Level Confidence Annotation Using Combinations of Features", Proc. of European Conference on Speech Communication and Technology, 2001.

[Zhang and Rudnicky, 2003a] Zhang, R., and Rudnicky, A. I., "Improving the Performance of an LVCSR System through Ensembles of Acoustic Models", Proc. of International Conference on Acoustics Speech and Signal Processing, 2003.

[Zhang and Rudnicky, 2003b] Zhang, R., and Rudnicky, A. I., "Comparative Study of Boosting and Non-Boosting Training for Constructing Ensembles of Acoustic Models", Proc. of European Conference on Speech Communication and Technology, 2003.

[Zhang and Rudnicky, 2004a] Zhang, R., and Rudnicky, A. I., "A Frame Level Boosting Training Scheme for Acoustic Modeling", Proc. of International Conference on Spoken Language Processing, 2004.

[Zhang and Rudnicky, 2004b] Zhang, R., and Rudnicky, A. I., "Apply N-Best List Re-Ranking to Acoustic Model Combinations of Boosting Training", Proc. of International Conference on Spoken Language Processing, 2004.

[Zhang et al, 2005] Zhang, R., Bawab, Z. A., Chan, A., Chotimongkol, A., Huggins-Daines, D., Rudnicky, A. I., "Investigations on Ensemble Based Semi-Supervised Acoustic Model Training", Proc. of European Conference on Speech Communication and Technology, 2005.

[Zhang and Rudnicky, 2006a] Zhang, R., and Rudnicky, A. I., "A New Data Selection Approach for Semi-Supervised Acoustic Modeling", Proc. of International Conference on Acoustics Speech and Signal Processing, 2006.

[Zhang and Rudnicky, 2006b] Zhang, R., and Rudnicky, A. I., "Investigations of Issues for Using Multiple Acoustic Models to Improve Continuous Speech Recognition", Proc. of International Conference on Spoken Language Processing, 2006.

[Zitouni et al, 2002] Zitouni, I., Kuo, H. K. J., and Lee, C. H., "Combination of Boosting and Discriminative Training Techniques for Natural Language Call Steering Systems", Proc. of International Conference on Acoustics Speech and Signal Processing, 2002.

[Zolnay et al, 2005] Zolnay, A., Schlueter, R., and Ney, H., "Acoustic Feature Combination for Robust Speech Recognition", Proc. of International Conference on Acoustics Speech and Signal Processing, 2005.

[Zweig and Padmanabhan, 2000] Zweig, G., and Padmanabhan, M., "Boosting gaussian mixtures in an LVCSR system", Proc. of International Conference on Acoustics Speech and Signal Processing, 2000.