Enhancing Posterior Based Speech Recognition Systems

THÈSE N^{O} 4218 (2008)

PRÉSENTÉE LE 14 NOVEMBRE 2008

À LA FACULTE SCIENCES ET TECHNIQUES DE L'INGÉNIEUR

Laboratoire de l'IDIAP

SECTION DE GÉNIE ÉLECTRIQUE ET ÉLECTRONIQUE

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

POUR L'OBTENTION DU GRADE DE DOCTEUR ÈS SCIENCES

PAR

Hamed KETABDAR

B.Sc. in electronical engineering, Sharif University of Technology, Téhéran, Iran
et de nationalité iranienne

acceptée sur proposition du jury:

Prof. Ph. Renaud, président du jury
Prof. H. Bourlard , directeur de thèse
Prof. B. Byrne, rapporteur
Prof. R. Rose, rapporteur
Prof. J.-Ph. Thiran, rapporteur

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Lausanne, EPFL
2008

Abstract

The use of local phoneme posterior probabilities has been increasingly explored for improving speech recognition systems. Hybrid hidden Markov model / artificial neural network (HMM/ANN) and Tandem are the most successful examples of such systems. In this thesis, we present a principled framework for enhancing the estimation of local posteriors, by integrating phonetic and lexical knowledge, as well as long contextual information. This framework allows for hierarchical estimation, integration and use of local posteriors from the phoneme up to the word level. We propose two approaches for enhancing the posteriors. In the first approach, phoneme posteriors estimated with an ANN (particularly multi-layer Perceptron - MLP) are used as emission probabilities in HMM forward-backward recursions. This yields new enhanced posterior estimates integrating HMM topological constraints (encoding specific phonetic and lexical knowledge), and long context. In the second approach, a temporal context of the regular MLP posteriors is post-processed by a secondary MLP, in order to learn inter and intra dependencies among the phoneme posteriors. The learned knowledge is integrated in the posterior estimation during the inference (forward pass) of the second MLP, resulting in enhanced posteriors. The use of resulting local enhanced posteriors is investigated in a wide range of posterior based speech recognition systems (e.g. Tandem and hybrid HMM/ANN), as a replacement or in combination with the regular MLP posteriors. The enhanced posteriors consistently outperform the regular posteriors in different applications over small and large vocabulary databases.

Keywords: Posterior Based ASR, Artificial Neural Networks, Local Posteriors, Context, Phonetic and Lexical Knowledge, Enhanced Posteriors

Résumé

L'utilisation des probabilités a posteriori locales de phonèmes a été explorée de plus en plus ces dernières années afin d'améliorer les systèmes de reconnaissance de la parole. La solution hybride de modèle de markov caché / réseau de neurones artificiels (HMM/ANN) et Tandem sont au jour d'aujourd'hui les exemples les plus réussis de tels systèmes. Dans cette thèse, nous présentons un dispositif afin d'améliorer l'estimation des a posteriori locaux, en intégrant des connaissance phonétiques et lexicales ainsi que des informations contextuelles relativement étalées dans le temps. Ce dispositif permet une estimation, une intégration et une utilisation hiérarchique des locaux a-posteriori depuis le phonème jusqu'au niveau du mot. Nous proposons deux approches pour améliorer les a posteriori. Dans la première approche, les phonèmes a posteriori estimés avec un ANN (en particulier un perceptron multi-couches - MLP) sont utilisés comme probabilité d'émission dans les récursions avant et arrière du HMM. Ceci provoque de nouvelles estimations d'a posteriori améliorées intégrant les contrainte topologiques d'un modèle de Markov caché (incluant des connaissances phonétiques et lexicales spécifiques), et un contexte temporelle de longue durée. Dans la seconde approche, un contexte temporel des a posteriori MLP classiques est post-traité par un MLP secondaire, de manière à apprendre les inter et intra-dépendances parmis les phonemes a posteriori. La connaissance apprise est intégrée dans l'estimation a posteriori pendant l'inférence (la passe en avant - forward pass) du second MLP, provoquant des a posteriori améliorés. L'utilisation de ces a posteriori locaux améliorés est étudié dans une large gamme de systèmes de reconnaissance de la parole basés sur les a posteriori (par exemple Tandem, HMM/ANN hybride), comme remplacement ou en combinaison avec les a posteriori MLP classiques. Les estimations améliorées des a posteriori

surpassent de manière consistante les a posteriori MLP classiques dans différentes applications, ceci sur des bases de données à petites et grandes tailles de vocabulaire.

Mots-clés : ASR basé sur les probabilités a posteriori, réseaux de neurones artificiels, probabilité locale a posteriori, information contextuelle, connaissance phonétique et lexicale, estimations améliorées des probabilités a posteriori

Contents

List of Figures

List of Tables

Acknowledgements

It is a pleasure to thank many colleagues and friends who provided great support for this thesis. Specially, I would like to thank my supervisor, Prof. Hervé Bourlard, for teaching me principles of automatic speech recognition, providing me with his deep insight and sophisticated ideas, and supporting this work over the period of last 4 years. It was a honor to be supervised by him. I hope I have learnt his advises well and always remember them in my future research career. I also specially thank Samy Bengio, former Idiap member, for close collaboration in this work. During the last 4 years, I could benefit a lot from excellent scientific and friendly atmosphere in Idiap. I take this opportunity to thank my colleagues Mathew Magimai Doss, Hynek Hermansky, Joel Pinto, and Jithendra Vepa for very helpful discussions.

This work was supported by the Swiss National Science Foundation through the National Center of Competence in Research (NCCR) on Interactive Multimodal Information Management (IM2) project, as well as the EU AMIDA (Augmented Multi-party Interaction with Distance Access) project.

Chapter 1

Introduction

1.1 Objective of the Thesis

Automatic speech recognition (ASR) is the task of processing spoken input and transcribing it into text for a computer application. The speech recognition engine combines acoustic evidences with lexical and grammatical knowledge to decode the message in the utterance. The acoustic evidence is estimated by an acoustic model which is trained on features extracted from the signal. The estimation of these acoustic evidences is based only on limited acoustic information in one or a few speech frames. However, acoustic information related to a speech sub-word unit (e.g. a phoneme) is spread over a long temporal context. In addition, there are linguistic sources of knowledge about duration of phones, co-articulation between phones, and lexical use of phones in the words. The main objective of this thesis is enhancing the estimation of these local acoustic evidences by integrating phonetic and lexical knowledge, as well as long contextual information. The proposed approaches provide a general framework for hierarchical estimation, enhancement and use of local acoustic evidences.

1.2 Posterior Based Speech Recognition Systems

Generative models such as Gaussian mixture models (GMMs) have been widely used as acoustic models in ASR as a part of hidden Markov model (HMM) configuration. However, over the past 15 years discriminant acoustic models, such as artificial neural networks (ANNs), have been increasingly investigated for estimating more discriminant acoustic evidences in the form of local posterior probabilities. ANNs take the advantage of discriminative training, model accuracy, and computational efficiency. Discriminative training in ANNs allows for minimizing the classification error, while maximizing discrimination between the correct output class and rival ones. ANNs do not require detailed assumptions about the form of the statistical distribution to be modeled, yielding more accurate acoustic models. They estimate posterior probability of output classes (usually sub-word units) conditioned on the input acoustic patterns, resulting in useful pattern recognizers. In this thesis, we mainly focus on acoustic evidences estimated by ANNs in the form of posterior probabilities.

In ASR systems using ANNs, local posterior probabilities have been usually estimated for sub-word (phone[1]) units. The term 'local' indicates that the posteriors are estimated for a local frame of speech, although the acoustic information or conditions for the estimation of local posteriors may not be limited only to a local frame[2]. We show local posteriors in the form of $p(q_t^i|A)$, where q_t^i is the event of having a speech unit i (e.g. a phone) at frame t, and A represents all conditions (e.g. model parameters, acoustic information) for the estimation of posterior. These local posterior probabilities have usually been used either as local scores (measures) or as features in frame synchronous ASR systems. Hybrid hidden Markov model / artificial neural network (HMM/ANN) approaches [1] were among the first to use posterior probabilities as local scores. In these approaches, ANNs and more specifically multi-layer Perceptrons (MLPs) are used to estimate the state emission proba-

[1]The acoustical realization of phonemes are called phones.

[2]In the remainder of the thesis, whenever we use the term posterior, we mean 'local posterior' unless otherwise mentioned.

bilities required in HMMs. Hybrid HMM/ANN method allows for discriminant training, as well as for the possibility of using small acoustic context by presenting few frames at MLP input. Regarding the use of posterior probabilities as features, one successful approach is Tandem [2]. In Tandem, a trained MLP is used for estimating local phone posteriors. These posteriors, after some transformations, are used as acoustic feature inputs to a standard HMM/GMM module. Tandem takes the advantage of discriminative acoustic model training, as well as being able to use the techniques developed for standard HMM/GMM systems.

In addition to the use of ANNs, some other posterior estimation approaches based on discriminant transformation of acoustic evidences have been proposed [3, 4]. In this thesis, we mainly concentrate on posteriors estimated by ANNs.

1.3 Motivation of the Thesis

Several approaches in the direction of using local phone posteriors as local scores or as features have recently been shown to have a significant potential for improving state-of-the-art ASR systems. However, further progress in that direction will depend on enhancing these posterior probability estimates. Normally, the estimation of local phone posteriors is based only on one or limited number of spectral feature frames. In this thesis, we refer to these posteriors as “MLP posteriors” or “regular posteriors”. However, phone information in the speech signal is not limited to just a few spectral feature frames. In general, any evidence of the way in which the information about underlying linguistic process is distributed in the signal is of importance and can be useful for estimating more informative posteriors. Information about phones is spread over a long temporal context (at least an interval of 200-300 ms) and there are no distinct boundaries between phones [5, 6], therefore taking into account long contextual information can be useful. With the same motivation, some linguistic knowledge such as duration of phones, co-articulation between phones (phonetic knowledge) and the lexical use of phones in a word can be useful for improving posterior

estimates. However, such linguistic and long contextual information is not usually taken into account in the regular posterior estimation.

There have been few recent studies for estimating posterior probability of a word hypothesis, given all acoustic observations of the utterance, by using forward-backward algorithm [7] through HMM or word graphs [8, 9, 10]. However, these studies are mainly focused on estimating word hypothesis posteriors (and not local posteriors) to measure confidence level of recognizer output.

1.4 Contribution of the Thesis

In this thesis, we present a principled framework for enhancing the estimation of local posteriors (particularly phone posteriors) by integrating long acoustic context, as well as phonetic and lexical knowledge. However, as opposed to the previously mentioned approaches [8, 9], the goal here is to provide local enhanced posteriors which can be used in frame synchronous posterior based ASR applications. The input in our approaches is regular phone posteriors estimated by an MLP, and the outcome is "local enhanced posteriors" of *phones*[3] at the *frame* level. Many posterior based ASR algorithms are based on phone evidences at the frame level. Therefore, the resulting frame based (local) enhanced posteriors can be used in a wide range of posterior based ASR systems (e.g. Tandem and hybrid HMM/ANN), as a replacement or in combination with the regular MLP posteriors in a straightforward manner. This posterior estimation/integration approach provides a theoretical framework for hierarchical estimation, integration and use of posteriors, from the state up to the phone and word units.

We propose two approaches for integrating linguistic (phonetic, lexical) and contextual knowledge in the posterior estimation:

[3] Although as it is shown in Chapter 7, we can also use our approach for local word posterior estimation.

- The first approach uses a HMM to integrate the prior phonetic and lexical knowledge. The phonetic and lexical knowledge is encoded in the topology of the HMM. The integration is realized by using the regular MLP posteriors as state emission probabilities in the HMM forward-backward recursions [7]. This yields new enhanced posterior estimates taking into account the encoded knowledge in the topology of the HMM.

- The second approach uses a secondary neural network (MLP) to post-process a temporal context of regular phone posteriors, and learn long term intra and inter dependencies between regular phone evidences (posteriors) estimated initially by the first MLP. These long term dependencies are phonetic knowledge. The learned phonetic knowledge is integrated in the phone posterior estimation during the inference (forward pass) of the second MLP, resulting in enhanced posteriors.

We present different aspects and applications of these new enhanced posterior estimates for improving speech recognition systems. We have shown that the enhanced posteriors can be used as replacement or in combination with the regular posteriors in frame synchronous ASR systems:

- **Enhanced posteriors as features:** Enhanced posteriors can be used as features for training/inference in ASR. We show that the enhanced posteriors can be used alone or in combination with the regular posteriors as more informative acoustic features, in a configuration similar to Tandem system. Compared to the use of regular posteriors, we have achieved consistent frame and word recognition improvement with the new Tandem configuration on different small and large vocabulary databases.

- **Enhanced posteriors as local scores:** Enhanced posteriors can be also used as local scores in ASR. We study the enhanced posteriors for estimating HMM state emission probabilities and decoding, in a configuration similar to hybrid HMM/ANN system. We have observed consistent improvement in frame, phone and word recognition on different databases, and also interesting results on the robustness of the

performance with respect to tuning ad-hoc parameters (e.g. phone and word insertion penalties). Moreover, posterior probabilities have been used as local scores in confidence measurement. These measures are typically based on accumulating local phone posteriors within a hypothesis, followed by the normalization with respect to the length of the hypothesis. We propose to use the enhanced posteriors as more informative local scores instead of the regular posteriors in the confidence measurement. It is shown that the use of the enhanced posteriors leads to consistently better confidence level estimation.

- **Multi-stream enhanced posteriors:** The estimation of enhanced posteriors can be extended to the case of multi-stream features. In this case, new enhanced posteriors are estimated through a multi-stream HMM by combining multiple streams of features, as well as integrating prior linguistic and contextual knowledge.

- **Higher level posterior estimation:** The theoretical framework of estimating enhanced local posteriors can be extended to higher level (e.g. word) posteriors. The estimation and use of local word posteriors is investigated through the practical case of keyword spotting problem. We estimate a keyword and a garbage unit posterior at every frame. These frame level posteriors are used to make a decision (vote) on detection of the keyword at each frame. The frame level decisions (votes) are then accumulated (by counting) to make a global decision on having the keyword in the utterance. In this way, the contribution of possible outliers is minimized as opposed to the conventional Viterbi decoding approach [11]. A strong outlier can change only few frame level decisions (votes), while it can significantly affect a global likelihood score obtained by Viterbi decoding.

- **Comparing regular and enhanced posteriors:** We also studied the difference (deviation) between the regular and enhanced posteriors. The deviation can provide a frame based measure on match/mismatch between the data and prior knowledge at different levels. The use of the deviation measure is investigated through the practical case of Out-of-Vocabulary (OOV) word detection.

Simply stated, we propose here to replace or complement the use of regular MLP posteriors by the new enhanced estimates of these posteriors, and we show some important practical cases. One can think of other applications of local posteriors in ASR, and simply use the new posteriors instead of the regular ones.

1.5 Organization of the Thesis

The thesis is organized in 9 chapters. Chapter 2 gives a general overview of state-of-the art speech recognition systems. Different components of a traditional speech recognition system including feature extraction, acoustic modeling and decoding are briefly described. The chapter continues with the review of estimation and use of local posterior probabilities in ASR. We study ANNs as acoustic models for phone posterior estimation, and their advantages over other conventional acoustic models. We present the state-of-the-art applications of posteriors in ASR as local scores in a hybrid HMM/ANN configuration, as well as discriminant features for HMMs.

In Chapter 3, we present our framework for the integration of phonetic, lexical and long contextual knowledge in the local posterior estimation. We propose two approaches for enhancing posterior estimates.

Chapter 4 describes the use of the enhanced posteriors as features for HMMs. We show that the enhanced posteriors can be used as replacement or in combination with the regular posteriors as acoustic features for a standard HMM/GMM back-end. In Chapter 5, we study the use of the enhanced posteriors as local scores in posterior based ASR. In the first part of this chapter, we study enhanced posteriors for modeling the HMM state emission probabilities in ASR decoding. The second part of the chapter studies the use of enhanced posteriors in confidence measurement. In all the studies, we compare the performance of the enhanced posteriors with the regular posteriors for frame, phone and word recognition over different small and large vocabulary databases.

Chapter 6 presents the extension of the enhanced posterior estimation framework to the multi-stream processing. We study the estimation of enhanced posteriors through a multi-stream HMM by combining multiple features, as well as integrating prior linguistic and contextual knowledge.

In Chapter 7, we extend our studies to higher level (e.g. word) posterior estimation. We present initial investigations on the estimation and use of local word posteriors through the keyword spotting problem. A new scoring approach for keyword spotting is studied and compared with the conventional keyword spotting approaches.

In Chapter 8, we investigate the difference (deviation) between the enhanced and regular posteriors, in order to detect inconsistencies between data and model (prior knowledge). As a particular case, the application of the deviation measure in detecting out-of-vocabulary words is studied.

Chapter 9 summarizes the work presented in this thesis, and draw final conclusions along with some future research directions.

Chapter 2

Overview of Speech Recognition Systems

An automatic speech recognition system (ASR) is a system which transforms a raw speech signal into a sequence of descriptors that characterize the spoken utterance for a computer, in order to perform appropriate actions. Usually the objective is to recognize a sequence of words. Given an utterance in terms of acoustic sequence $X = \{x_1, ..., x_t, ..., x_T\}$, this objective is formulated statistically as finding the word sequence $\hat{W}$ which is most likely to have produced X [7, 12]:

$$\hat{W} = \arg\max_{W} P(W|X, M) \tag{2.1}$$

where W is a word sequence out of the set of all possible word sequences, and M represents the set of parameters of the model which is estimated from training data. Practically, direct estimation of the probability $P(W|X, M)$ is not feasible. However, $P(W|X, M)$ can be rewritten by applying Bayes rule:

$$P(W|X,M) = \frac{p(X|W,M)P(W|M)}{p(X|M)} \tag{2.2}$$

Therefore, (2.1) can be rewritten in maximum likelihood form:

$$\hat{W} = \arg\max_{W} \frac{p(X|W,M)P(W|M)}{p(X|M)} \tag{2.3}$$

The term $p(X|M)$ is common to all the word hypotheses, thus it can be dropped. The set of parameters of the model M can be decomposed into two parts, acoustic model parameters M_a, and language model parameters M_l. The speech recognition system is parameterized by these two sets of parameters, $M = \{M_a, M_l\}$. M_a and M_l are assumed to be independent and estimated separately [12, 13, 14]. Therefore, the term $p(X|W,M)$ can be written as $p(X|W,M_a)$ and referred as acoustic model term. Further, the term $P(W|M)$ which is the prior probability of word sequence is written as $P(W|M_l)$, and given by the language model. (2.3) can thus be rewritten as:

$$\hat{W} \approx \arg\max_{W} \left\{p(X|W,M_a)P(W|M_l)\right\} \tag{2.4}$$

The parameters of the acoustic model, M_a, are estimated based on maximum likelihood (ML) criterion [7, 12] using a training database and its transcription:

$$\hat{M}_a = \arg\max_{M_a} \prod_{X} p(X|W,M_a) \tag{2.5}$$

where the product is over the whole utterances X in the database, and $\hat{M}_a$ is the optimum set of parameters. In the state-of-the-art ASR systems, typically hidden Markov models (HMMs) are used for acoustic modeling [12, 13, 15, 16]. The language model parameters, M_l are estimated by counting the frequency of word sequences [13, 17]. The language model can be an n-gram [13, 18], where n=2, and n=3 (bi-gram and tri-gram, respectively) are the most common cases.

In the following, we briefly summarize different components of an ASR system. Figure 2.1 shows a block diagram of a standard speech recognition system. The first major step is deriving a parameterized version of the input speech signal which discards redundant parts of the signal and only represents information about the spoken message. This step is called feature extraction. The process of feature extraction is described in Section 2.1.1. The outcome of the feature extraction step is a sequence of acoustic feature vectors $X = \{x_1, ..., x_t, ..., x_T\}$. These feature vectors are modeled in the next module which is acoustic modeling, in order to obtain reference models. This involves estimating M_a parameters. The acoustic models can be made for words or shorter speech units (sub-word units) such as phonemes. A phoneme is the smallest unit of speech that affects the meaning of a word and distinguishes one word from another in a given language. Since there can be many words in a large vocabulary ASR system, it may not be practically possible to build models for words, due to the lack of enough training data. Therefore, usually sub-word (phoneme) models are used. The transcription of words in terms of sub-word units is specified by the lexicon of the ASR system. Section 2.1.2 describes the acoustic modeling details. Finally, the last module of an ASR system is decoding. In this module, the acoustic model scores are combined with lexical knowledge and language model in order to find the most likely sequence of words. Decoding is explained in Section 2.1.3.

Overview of the speech recognition systems is followed by the study of artificial neural networks (ANNs) as more discriminant acoustic models. ANNs are able to estimate more discriminant acoustic evidences in the from of local posterior probabilities. The estimation and use of these posterior probabilities are reviewed in Section 2.2.

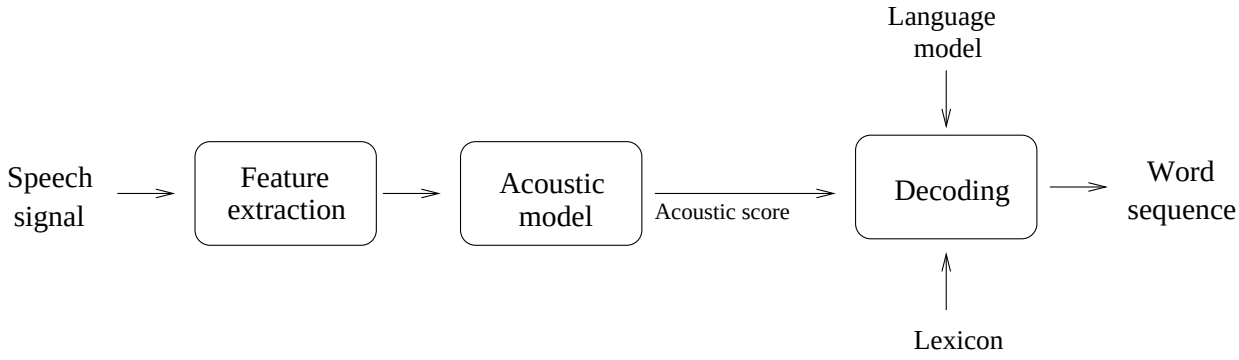


Figure 2.1: A standard speech recognition system.

2.1 Components of Speech Recognition Systems

2.1.1 Feature Extraction

Feature extraction is the process of extracting a limited amount of useful information from speech signal, while discarding redundant and unwanted information. In the other words, feature extraction maintains much of the characteristics of the original speech and eliminates much of extraneous information.

The speech signal is produced by the excitation of time varying vocal tract system by a time varying signal. The excitation is generated by air flow through the vocal cords. As the excitation acoustic waves pass through the vocal tract, the frequency content (spectrum) is modulated by the resonances of the vocal tract. The shape of vocal tract (the bandwidth and location of resonances) is more representative of the sound (phoneme) being produced, while the excitation signal is more delivering speaker dependent and higher level linguistic information. In speech recognition, we are preliminary interested in the sounds being pronounced rather than high level linguistic information. Therefore, the feature extraction module is usually designed to preserve the information related to vocal tract shape, while discarding the excitation information. Ideally, features used in ASR should be highly dis-

criminative between different sub-word (phoneme) classes, have low speaker variability, and be invariant to degradations caused by channel and noise.

Different feature representations have been developed to emphasize on the mentioned desirable properties of the features. The features used in state-of-the-art ASR systems are usually derived from spectral representation of speech signal obtained by short-time Fourier transform (STFT). The spectral representation of the speech signal is much more informative for speech sound discrimination than the time domain signal. Most of ASR systems parameterize the spectral envelope in the form of a few (10-14) dimensional feature vector. The spectral envelope can be characterized by linear prediction (LP) parameters and their transformations, or by cepstrum [19]. The cepstrum de-convolves the vocal tract response from the excitation.

The speech signal is non-stationary. To enable the use of spectral analysis algorithms (such as Fourier transform) which assume signal stationarity, short segments of the signal (10-30 ms) are used to derive short-term features for ASR. The signal can be assumed stationary within these short segments. These short segments are called frames. The signal dynamics are then represented by a sequence of the short-term feature vectors $X = \{x_1, ..., x_t, ..., x_T\}$ with each vector representing a sample from the actual underlying process. In order to segment the signal into frames, a short duration window (in which the signal can be assumed stationary) is applied to the signal. The window is applied by multiplying the speech signal with a window function $w(t)$ of length L. The length L corresponds to the frame size. Multiplying the signal with the window in time domain corresponds to convolving the frequency response of the window with the speech spectrum. The window should be selected in a way to minimize its effect on the original speech spectrum. The Hamming window [20] is commonly used for windowing in short-time speech signal processing. Usually having an overlap between the frames, called as frame shift is necessary. This is because the Hamming window tappers at the edges, resulting in attenuation of the samples at the edges. Standard ASR systems use frame size of 25-30 ms with a frame shift of 10-20 ms.

The most commonly used short-time features in ASR are MFCC [21] and PLP-derived cepstral coefficients [22]. Both of these approaches are based on spectral analysis of speech signal and using knowledge about human speech perception. In the present work, we have used PLP-derived cepstral coefficients to develop the baseline systems. In the PLP approach, first the short-time Fourier transform of the windowed signal is computed. Then the energy in each band of a filter bank defined based on bark scale [23, 24] is obtained. The resultant filter bank energies are multiplied by an equal-loudness curve, and to simulate the power law of hearing [25], cube-root compression is applied to the output amplitudes. The final smooth spectrum is transformed by IFFT, and using auto-regressive modeling [19, 26] the PLP coefficients are obtained.

The features so far obtained have only information about the present frame and do not carry any temporal or dynamic information. In fact, frames are treated independently to simplify the computation. However, strong correlation exists across frames mainly due to co-articulation [5, 27]. In order to take into account the time correlation and time trajectory of features, the first order temporal derivatives (Δ) and second order temporal derivatives ($\Delta\Delta$) of the acoustic vectors are commonly used as additional acoustic parameters [28, 29]. Therefore, assuming the use of 13 spectral-based coefficients, 13 Δ and 13 $\Delta\Delta$ coefficients are appended to each feature vector, forming a final 39 dimensions feature vector.

The acoustic features have considerable variability. These variabilities lead to high variance in the feature space of a sound. Reducing the variability of features can greatly simplify the ASR problem. In order to deal with the variabilities, the features are post processed before the statistical inference. This post processing usually include vocal tract length normalization (VTLN) [30, 31], cepstral mean subtraction and variance normalization [32, 33, 34].

The final outcome of the feature extraction is transforming speech signal into a sequence of acoustic feature vectors $X = \{x_1, ..., x_t, ..., x_T\}$.

2.1.2 Acoustic Modeling

The acoustic modeling process builds reference models for speech units using the extracted features as statistical samples. As previously described, the parameters of the acoustic model, M_a, are estimated based on maximum likelihood (ML) criterion [7, 12] using a training database and its transcription:

$$\hat{M_a} = \arg\max_{M_a} \prod_{X} p(X|W, M_a) \tag{2.6}$$

where the product is over all utterances X in the database. In this equation, the aim is to find M_a which maximizes the likelihood of the training set. A popular iterative algorithm for estimating M_a is expectation-maximization (EM) [35, 36]. In EM, a few unobserved latent variables are considered in the parameter set M_a to simplify the otherwise intractable problem. In each iteration of EM, the parameter set M_a is re-estimated using the previous parameter estimates $\acute{M}_a$ such that the likelihood of the training set is increased. Each EM iteration has two steps, estimation and maximization. EM alternates between performing the expectation step, which estimates an expectation of the likelihood by including the latent variables (as if they were observed), and a maximization step, which estimates the maximum likelihood estimates of the parameters by maximizing the expected likelihood obtained in the expectation step. The parameters obtained in the maximization step are then used to begin another expectation step, and the process is repeated. The two steps when repeated tend to increase the likelihood of the data (guarantied not to decrease the likelihood). Proof of convergence for EM algorithm can be found in [35, 36].

In speech recognition systems, hidden Markov models (HMMs) are typically used for acoustic modeling [7, 12, 13, 15, 16]. In the following, we describe their usage in ASR systems.

Hidden Markov Models (HMMs) for ASR

Hidden Markov models (HMMs) are parametric stochastic models for sequences. They can be used to represent distributions over sequences in which context can be represented by discrete states. An HMM represents a stochastic process generated by an underlying Markov chain composed of a number of states, and a set of observation distributions associated to these states. Generally, some physical interpretations can be assigned to the states. In standard ASR systems, the states usually represent phonemes (or parts of phonemes).

The probability of the observation sequence X having the word sequence W and the hidden Markov model M_a (as acoustic model) can be written as:

$$\begin{aligned} p(X|W, M_a) &= \sum_S p(X, S|W, M_a) \\ &= \sum_S p(X|S, W, M_a) P(S|W, M_a) \end{aligned} \tag{2.7}$$

where S is an HMM state sequence, and $\sum_S$ shows summation over all possible state sequences S in W. In the reminder of this chapter, we often drop the M_a keeping in mind that all the equations are written assuming the hidden Markov model M_a. There are two terms in the equation, $p(X|S, W)$ and $P(S|W)$, which are solved separately. $p(X|S, W)$ can be estimated as:

$$p(X|S, W) = p(x_1, ..., x_T|S, W) \tag{2.8}$$

To make the estimation of the above term practically feasible, we assume that the probability of the current observation x_t depends only on the current state s_t, and is independent of all other observations and states (c.i.i.d assumption). We also assume that the probability

is not time dependent. These assumptions result in:

$$p(X|S,W) \approx \prod_{t=1}^{T} p(x_t|s_t) \tag{2.9}$$

The second term, $P(S|W)$, is resolved based on the assumption that the current state s_t depends only on the previous state s_{t-1} (first order Markov assumption):

$$\begin{aligned} P(S|W) &= P(s_1, ..., s_T|W) \\ &\approx P(s_1) \prod_{t=2}^{T} P(s_t|s_{t-1}) \end{aligned} \tag{2.10}$$

Based on (2.9) and (2.10), the likelihood $p(X|W)$ can be rewritten as:

$$p(X|W, M_a) = \sum_{S} \left\{ P(s_1)p(x_1|s_1) \prod_{t=2}^{T} P(s_t|s_{t-1})p(x_t|s_t) \right\} \tag{2.11}$$

In practice, the above equation is evaluated using a computationally efficient procedure known as Baum-Welch or forward-backward algorithm [36]. (2.11) can be also approximated by computing the likelihood of the best state sequence using Viterbi algorithm [11, 37, 38, 39, 40] as follows:

$$p(X|W, M_a) = \max_{S} \left\{ P(s_1)p(x_1|s_1) \prod_{t=2}^{T} P(s_t|s_{t-1})p(x_t|s_t) \right\} \tag{2.12}$$

Considering (2.11), every state is parameterized in terms of two probability distributions, namely, state-transition probabilities a_{ij}:

$$a_{ij} = P(s_t = j|s_{t-1} = i) \tag{2.13}$$

and emission probability density function $b_j(x_t)$:

$$b_j(x_t) = p(x_t|s_t = j) \tag{2.14}$$

Therefore, the HMM topology consists of the states emission probability density functions $b_j(x_t)$, and state transition probabilities a_{ij}. The acoustic model parameter set M_a in (2.6) consists of the parameters of the emission density functions and the state transition probabilities. These parameters are obtained by training over a database. The most efficient training algorithm is the above mentioned Baum-Welch algorithm (also known as forward-backward algorithm). This algorithm is a special case of expectation-maximization (EM) algorithm (previously discussed). The approximation of this algorithm by embedded Viterbi algorithm can be also used for training HMM parameters. Training an HMM is an iterative procedure. The training starts with an initial estimation of the parameters, then an iterative re-estimation of the parameters is done in such a way that it yields better models. A description of EM algorithm for training HMM parameters can be found in [7, 41].

Typically, HMMs are built for sub-word units such as phonemes [38]. The sub-word units HMMs are connected together to form word HMMs. The lexicon of an ASR system contains the transcription of words in terms of sub-word units. In the state-of-the-art ASR systems, these sub-word units are usually phonemes. A phoneme is defined as the smallest sound within a particular language which affects the meaning of a word and distinguishes one word from another. The acoustical realization of phonemes are called phones. Although it is theoretically possible to build acoustic models directly for words, in practice it is difficult to have sufficient training samples (realization of each word) in a large vocabulary system. Therefore, the practical solution is to train phoneme models, and connect theses models based on the lexicon to create word models. There are two types of phoneme models, context independent (CI) phonemes, and context-dependent (CD) phonemes. In context-independent modeling, each phoneme model is trained independent of the others. CI modeling can not take into account the effect of co-articulation which extends over a single phoneme. In order to model co-articulation, context-dependent (CD) phoneme models

are used [42, 43]. CD phonemes can capture most local co-articulations. Context-dependent models are created based on the current CI phoneme model and typically with one preceding and one succeeding context phones. The number of CD phoneme models are much more than CI phoneme models. This may result in insufficient data for training CD models. In order to overcome this problem, parameter tying techniques are used [44, 45]. Parameter tying allows some parameters of different CD models to be shared, hence reducing the actual number of parameters which should be trained.

There are two popular models for estimating the HMM emission probabilities $p(x_t|s_t)$, namely Gaussian mixture model (GMM) and artificial neural networks (ANNs). Using GMMs for modeling the emission probabilities is described in the following. The use of ANNs will be discussed in Section 2.2.2.

HMM/GMM Configuration

Gaussian (normal density) and mixture of Gaussian probability distribution functions are commonly used models for the observations associated to each state of a continues density HMM. An HMM configuration in which the emission probability is estimated by a Gaussian mixture model (GMM) is called HMM/GMM. The emission probability density $b_j(x_t) = p(x_t|s_t = j)$ in HMM/GMM system is given by:

$$b_j(x_t) = \sum_{l=1}^{L_j} c_{jl} N(x_t; \mu_{jl}, \Sigma_{jl}) \tag{2.15}$$

where L_j is the number of mixtures for state j, c_{jl} is the weight of l^{th} mixture component such that $\sum_{l=1}^{L_j} c_{jl} = 1$, and $N(x_t; \mu_{jl}, \sum_{jl})$ is a multivariate Gaussian with μ_{jl} and $\sum_{jl}$ as mean vector and covariance matrix respectively.

2.1.3 Decoding

The last component of the automatic speech recognition system is decoding. The decoder combines the acoustic model likelihood and language model probabilities to output the word sequence. As studied at the beginning of this chapter, decoding in statistical speech recognition involves a search for the best possible word sequence $\hat{W}$ given acoustic observation sequence X:

$$\begin{aligned} \hat{W} &= \arg\max_{W} P(W|X, M) \\ &\approx \arg\max_{W} \left\{ p(X|W, M_a) P(W|M_l) \right\} \end{aligned} \tag{2.16}$$

$P(X|W, M_a)$ is the acoustic model term and is estimated as explained through (2.7)-(2.11). $P(W|M_l)$ is the language model term and is obtained by counting word sequences over the training database. The use of the language model is dependent on the task in hand. A language model can be very complex for large vocabulary conversational speech recognition problem, while it can be very simple (uniform) for a task like digit recognition. The language model is usually a Markov model. For a given sequence of $\mathcal{M}$ words, the language model probability can be estimated as:

$$P(W|M_l) = \prod_{m=1}^{\mathcal{M}} P(w_m|w_{m-1}, ..., w_1) \tag{2.17}$$

where w_m is the m^{th} word in the sequence. Usually it is assumed that the current word w_m is only dependent to the n preceding words, with $n = 2$ called as a bi-gram and $n = 3$ as a tri-gram language model.

Including state sequence S in $p(X|W, M_a)$, (2.16) can be rewritten using (2.7) as:

$$\hat{W} = \arg\max_{W} \left\{ P(W|M_l) \sum_{S} p(X|S, W, M_a) P(S|W, M_a) \right\} \tag{2.18}$$

The above equation involves summing over all possible state sequences. This can be computationally very expensive specially when the number of words increases. The sum operation can be approximated with the $\max$ operation in order to reduce the computational load:

$$\hat{W} = \arg\max_{W} \left\{ P(W|M_l) \arg\max_{S} \{ p(X|S, W, M_a) P(S|W, M_a) \right\} \tag{2.19}$$

In the other words, the best path is searched among all possible state sequences. This is called Viterbi decoding [11]. The term $p(X|S, W, M_a)P(S|W, M_a)$ can be estimated using (2.9, 2.10). The language model term $P(W|M_l)$ is estimated based on (2.17).

2.2 Posteriors in Speech Recognition Systems

Using posterior probabilities has become popular and frequently investigated in the past two decades for improving ASR systems. Posterior probabilities have been used for lattice rescoring and system combination, confidence measurement, and also as features or local scores in ASR systems.

In [8, 9, 10], different methods for estimating posterior probability of a word hypothesis, given all acoustic observations of the utterance are proposed. These posteriors are estimated on HMMs or word graphs by the forward-backward (Baum-Welch) algorithm [7], and used for hypothesis confidence measurement. Such confidence scores have been used to combine the output of multiple recognition systems using the ROVER technique [46]. Word hypothesis posterior probabilities can be also used for post processing the word lattice generated by a viterbi decoder [47].

In [3, 4], a method based on using GMMs for estimating posteriors has been proposed. In this method, a large number of Gaussians are pooled from an acoustic model trained with maximum likelihood (ML) criterion. The likelihoods estimated using these Gaussians are normalized (assuming equal priors) to obtain a sparse set of posteriors. The dimension-

ality of this set is reduced by a transformation learned along with minimum phone error (MPE) training [48]. The MPE criterion evaluates the phone accuracy in the word context. In [49], the likelihoods estimated by GMMs (trained on acoustic data) are turned to posteriors trough conditional random fields. In this work, we are mainly concerned about the approaches using ANNs for posterior estimation.

Posterior probabilities have been also used as acoustic features or local scores in Tandem [2] and hybrid HMM/ANN [1] systems. In this case, usually 'local' posterior probabilities are estimated. The term 'local' indicates that the posterior is estimated for a speech unit at the current local frame, although the information and conditions for the estimation of the posterior may not only be limited to the local frame[1]. In this thesis, we are mainly focused on the estimation and use of local posterior probabilities. These local posterior probabilities are estimated for different speech units, and more particularly for sub-word units such as phones. The posterior probabilities have been usually estimated using artificial neural networks (ANNs), specifically multi-layer Perceptrons (MLPs). In these approaches, a limited context (e.g. 9 frames) of spectral features is presented at the input of the MLP. Each output of the MLP is associated with a particular context-independent phone, and estimates local posterior probability of the phone. The MLP is discriminatively trained to find a mapping between the spectral features at the input, and the phone targets at the output. Compared to other statistical estimators, ANNs have several advantages including discriminative training and model accuracy [1]. Discriminative training allows for minimizing the classification error, while maximizing discrimination between the correct output class and rival ones. ANNs do not require detailed assumptions about the form of the statistical distribution to be modeled, yielding more accurate acoustic models.

As mentioned, local posterior probabilities have been usually used either as local scores (measures) or as features in speech recognition systems. Hybrid hidden Markov model / artificial neural network (HMM/ANN) approaches [1] were among the first ones to use posterior probabilities as local scores. In these approaches MLPs are used to estimate the state

[1]In the remainder of the thesis, whenever we use the term 'posterior', it means 'local posterior' unless otherwise mentioned.

emission probabilities required in HMM. Hybrid HMM/ANN method allows for discriminant training, as well as for the possibility of using small acoustic context by presenting few frames at MLP input. We will describe the hybrid HMM/ANN system in more detail later in Section 2.2.2. In addition, local posterior probabilities can be used for measuring the confidence level of recognizer outputs [50, 51].

Posterior probabilities have been also used as acoustic features in ASR. The most successful approach for using posteriors as features is Tandem [2]. In Tandem, a trained MLP is used for estimating phone posteriors. These posteriors, after some transformations, can be used as input acoustic features to a HMM/GMM module. The MLP can be considered as doing optimal feature extraction using nonlinear discriminant analysis. Tandem technique takes the advantage of discriminative acoustic model training, as well as being able to use the techniques developed for standard HMM/GMM systems.

In the following, we review the estimation and use of local posterior probabilities in ASR systems. In this work, we are mainly concerned about the approaches using ANNs for posterior estimation. We start with an overview of ANNs.

2.2.1 Artificial Neural Networks as Statistical Estimators

An artificial neural network (ANN) is a mathematical model that is inspired by the way biological nervous systems, such as the brain process information. It is basically a dense interconnection of simple, non-linear computational elements of the type shown in Figure 2.2. In this computational element, there are N inputs $\{I_1, I_2, ..., I_N\}$, which are multiplied by weights $w_1, w_2, ..., w_N$, thresholded, and then nonlinearly compressed to give the output a, defined as:

$$a = f(\sum_{i=1}^{N} w_i I_i - \phi) \tag{2.20}$$

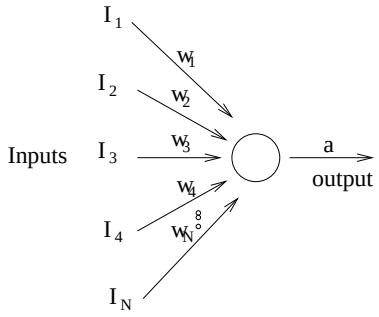


Figure 2.2: Basic computational element of an ANN. Inputs are multiplied by weights $w_1, w_2, ..., w_N$, thresholded, and then nonlinearly compressed to give the output a.

where ϕ is an internal threshold or offset, and f is a nonlinearity such as sigmoid functions or a hard limiter. Using sigmoid nonlinearities are more dominant, since these functions are continues and differentiable.

The topology of the ANN is specified by how these computational units are connected. There are few standard and well known topologies including single/multi-layer Perceptrons (SLP/MLP), recurrent networks (Hopfield) [52, 53], and self organizing networks (Kohonen) [54, 55]. Concerning the speech recognition problem, the main focus has been on using multi-layer Perceptrons (MLPs), although recurrent networks have also been investigated. In the single/multi-layer Perceptrons, the outputs of one or more simple computational elements at one layer form the inputs to a new set of simple computational elements of the next layer. Figure 2.3 shows a single layer Perceptron (top) and a multi (three) layer Perceptron (bottom). The single layer Perceptron has an input and an output layer. The multi-layer Perceptron has a hidden layer between the input and output layers. It can be proven that a SLP can separate static patterns into classes with class boundaries characterized by hyperplanes in the input feature space. An MLP can realize an arbitrary set of decision regions in the input feature space.

MLPs can been used to classify phones or words in ASR systems. In these approaches, a few frames of acoustic features are presented at the input layer. The hidden layer enables the correlation between the elements of the input feature vector to be modeled. The output

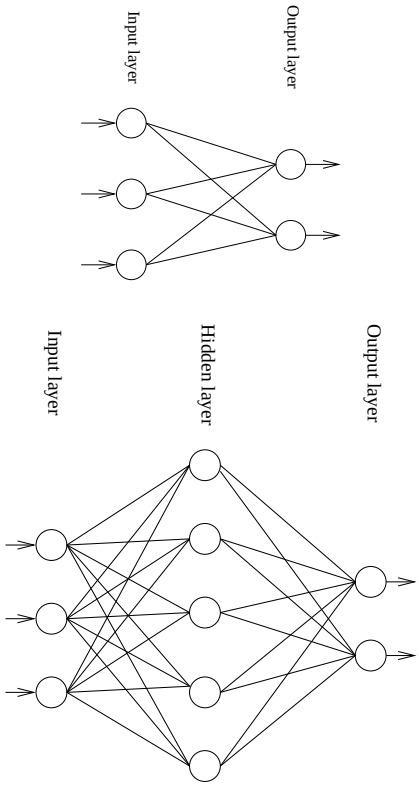


Figure 2.3: (top) Single layer Perceptron (SLP), and (bottom) multi-layer Perceptron (MLP).

layer consists of a separate output unit for each phonetic class. The MLP can learn the mapping between the acoustic features at the input and phone classes at the output. During the inference, the MLP performs phone classification or phone posterior estimation (as we will discuss later in this section) based on the acoustic features presented at its input. In this thesis, all MLPs are trained with 9 frames of acoustic feature vectors as input. The output is generally the number of context-independent phones for a given task.

Training a neural net involves determining the weighting coefficients and the offset threshold for each computational element, in order to minimize the error between the predicted output vector (output of the ANN) and the desired output vector. This supervised training is based on a labeled set of training data. A labeled set of training data is an association between a set of $\mathcal{N}$ input vectors x_1, x_2, ..., $x_\mathcal{N}$ and a set of $\mathcal{N}$ output vectors y_1, y_2, ..., $y_\mathcal{N}$, where $x_1 \rightarrow y_1$, $x_2 \rightarrow y_2$, ..., $x_\mathcal{N} \rightarrow y_\mathcal{N}$. The training of the ANN can be done efficiently by

error back propagation algorithm [56] that uses a gradient decent approach [57] to iteratively minimize a cost function. The performance of the network is measured by the cost function which is usually a differentiable function of the network outputs. For instance, a supervised training criterion can be the mean square error (MSE) [58]. In this case, the objective of the learning is to minimize the sum of squares of differences between network inputs and desired outputs. Cross entropy has also been used as a criterion for training ANNs [59]. The desired vector is usually one-hot-encoding in which the target phone class is assigned a value of 1.0 and all others 0.0. The target phone classes can be obtained either by manual segmentation of the training data (practical for small training sets), or using an already trained ASR system and Viterbi algorithm (also referred to as forced alignment).

Given the training data and the segmentation, the training of an ANN starts with initialization of the weights and an initial learning rate[2]. The update of the ANN weights is made after every training example (i.e. online training). A separate data set which is not part of the training data is used for cross validation in order to avoid over training of the ANN. After each iteration, the performance is evaluated over both cross validation data and training data. If the performance improves on the cross validation data then the training is continued, otherwise the learning rate is reduced by a factor of two for the next iteration. The training continues until the learning rate falls below a certain threshold.

ANNs used in classification mode (under certain conditions) can provide the estimates of posterior probabilities of output classes conditioned on the input [1, 59, 60, 61]. Here we describe the proof in [60, 61]. For continuous valued acoustic input vectors, the mean square error (MSE) criterion which is usually minimized during ANN training can be expressed as follows:

$$E = \int p(x) \sum_{k=1}^{K} \sum_{l=1}^{K} P(q^k|x) \left[g_l(x) - d_l(x)\right]^2 dx \tag{2.21}$$

[2]A learning rate determines how much the weights and node biases can be modified. The higher the learning rate (max. of 1.0) the faster the network is trained. However, the network has higher risk of being trained to a local minimum solution. A local minimum is a point at which the network stabilizes on a solution which is not the most optimal global solution.

where x is the input vector, q^k is the k^{th} ANN output (k^{th} class), $g_l(x)$ represents the observed output for class q^l given x at the input, and $d_l(x)$ represents the associated (desired) output. K is the total number of classes. We also have:

$$p(x) = \sum_{i=1}^{K} p(q^i, x) \tag{2.22}$$

(2.21) and (2.22) gives:

$$E = \int \sum_{i=1}^{K} \left[\sum_{k=1}^{K} \sum_{l=1}^{K} [g_l(x) - d_l(x)]^2 P(q^k|x) \right] p(q^i, x) dx \tag{2.23}$$

We assume that $d_l(x) = \delta_{kl}$ if $x \in q^k$. After some more algebraic manipulation of (2.23) we have:

$$\begin{aligned} E &= \int \sum_{i=1}^{K} \left[\sum_{l=1}^{K} (g_l(x) - P(q^l|x))^2 \right] p(q^i, x) dx \\ &+ \int \sum_{i=1}^{K} \left[\sum_{l=1}^{K} P(q^l|x)(1 - P(q^l|x)) \right] p(q^i, x) dx \end{aligned} \tag{2.24}$$

Minimization of E is achieved by choosing network parameters to minimize the first expectation term in the above equation. The second term is independent of the network outputs, and it does not contribute in the minimization of the cost function. In fact, The first expectation term is the MSE between the network output $g_k(x)$ and the posterior probability $P(q^k|x)$. This shows that a discriminant function obtained by minimizing the MSE has the property of being the best approximation to the Bayes probabilities in the sense of mean square error. A similar proof for the case of using cross entropy cost function can be found in [59]. In general, two conditions have to be satisfied to have an ANN as posterior probability estimator: (1) The network must have enough parameters (complex enough) to be trained to a good approximation of the mapping function between input and output classes, (2) A global error minimum criterion should be used to train the network. Mean squared

error and relative entropy are error criteria that satisfy this condition. However, it is experimentally shown that a network trained on huge amount of speech data is still able to approximate posterior probability of output classes, even if the error criteria is not exactly a global minimum error [1].

In ASR systems using ANNs, posterior probabilities have been usually estimated for context independent phones. In this case, typically an MLP is provided with one or a limited number of spectral feature vectors at its inputs, and it estimates local posterior $p(q_t^i|x_t)$ at the outputs. q_t^i is the event of having phone i at time t, and x_t is a spectral feature vector at time t. In this thesis, all the 'local' posteriors are shown by *small* 'p'. Usually more than one frame of acoustic features (small context) is presented at the input of the MLP, thus it estimates $p(q_t^i|x_{t-c}^{t+c})$, where c is typically equal to 4. x_{t-c}^{t+c} represents a short temporal context obtained by concatenating acoustic feature vectors in $\{x_{t-c}, ..., x_t, ..., x_{t+c}\}$. This is in fact very limited context [3].

Compared to the conventional (e.g. Gaussian mixture) estimators, ANNs have several advantages for probability estimation [1, 61]:

- Discrimination: ANNs trained for classification provide the possibility of discriminant learning. This means that ANNs trained using common cost functions such as least mean square error or relative entropy minimize classification error rate while maximizing discrimination between the correct output class and rival ones. As it was shown earlier, ANNs can provide good estimate of the posterior probability of output classes conditioned on the input patterns, resulting in useful pattern recognizers.

- Model accuracy: The use of ANNs results in more accurate acoustic models, because ANNs do not require detailed assumptions about the form of the statistical distribution to be modeled. In the other words, since ANNs can incorporate multiple constraints and find optimal combinations of constraints, there is no need for strong

[3] In the sequel of this thesis, and for simplicity sake, we will often write MLP posterior outputs as $p(q_t^i|x_t)$, though keeping in mind that they are often estimating $p(q_t^i|x_{t-c}^{t+c})$ if small acoustic context is provided at the input of MLP.

assumptions about the statistical distribution of the input features or about high order correlation in the input data. In theory, this can be discovered automatically by the ANNs during training. In most of conventional approaches, the number of significant mixtures for representing the distribution, or type of features (discrete or continues) should be often considered. These types of assumption are not required with an ANN estimator. ANNs with one hidden layer and enough complexity (enough hidden nodes) can approximate any continuous function.

- Bounded output: Posterior probability estimates (outputs of ANN) are independent of the input space dimension allowing for comparisons between different features. In contrast, the magnitude of the likelihoods depends on the size of the feature space.

- Context sensitivity: Usually a small context (more than one frame) of spectral features is presented at the input of ANN. This provides a simple mechanism for incorporating acoustic context into the statistical formulation. In case of using several acoustic vectors at the input of an MLP, local correlation of acoustic vectors can be taken into account in the probability distribution.

- Computational efficiency: Due to highly parallel and regular structures of ANNs, efficient parallel and hardware implementations of ANNs are feasible.

2.2.2 Posteriors as Local Scores

As we have discussed, ANNs can be used to classify speech units such as phones or words. This is the way ANNs were initially used for simple ASR problems. However, ANNs classifying complete temporal sequences have not been successful for continuous speech recognition, since the number of possible word sequences in an utterance is generally infinite. On the other hand, HMMs provide a reasonable structure for representing sequences of speech sounds or words. Assuming such a structure, one principled use for ANNs might be providing the scores (distance measures) for the local match (emission probabilities) in HMMs. Over the past 10-15 years, a number of systems referred to as hybrid HMM/ANN systems

[1, 61] have been developed. In these systems, ANNs have been discriminatively trained to estimate state emission probabilities for hidden Markov models (HMMs). It is shown that hybrid systems can be both effective in terms of accuracy and efficient in terms of CPU and memory run-time requirements. In hybrid HMM/ANN configuration, each state s^k of a set of HMM states $S = \{s^1, ..., s^k, ..., s^{N_s}\}$ is associated with one of MLP outputs representing posterior probability $p(q_t^i|x_t)$ of phone i at time t. The input to MLP is x_t which is a spectral feature frame at time t. q_t^i is the event of having phone i at time t. As mentioned before, usually more than one frame of acoustic features (small context) is presented at the input of the MLP. In standard HMM/ANN systems, these local posteriors are usually turned into "scaled likelihood" by dividing MLP outputs by their respective a priori probability $p(q_t^i)$ (estimated on the training data), i.e. $\frac{p(q_t^i|x_t)}{p(q_t^i)}$. Applying Bayes rule we have:

$$\frac{p(x_t|q_t^i)}{p(x_t)} = \frac{p(q_t^i|x_t)}{p(q_t^i)} \tag{2.25}$$

The left hand side of (2.25) is called scaled likelihood. In hybrid HMM/ANN based ASR, these scaled likelihoods are treated as HMM state emission probabilities.

The parameters set in hybrid HMM/ANN ASR consists of transition probabilities, parameters (weights) of the trained ANN, and the prior probability of each output unit (phone) of the ANN. The priors can be obtained from the segmentation of the training data. They are used for turning MLP output posterior probabilities to scaled likelihoods in (2.25). The HMMs used in hybrid systems usually have fixed state transitions of 0.5, however, they can be trained using Viterbi or forward-backward algorithms. ANN weights are commonly trained using already available phone segmentation of the training data. As described before, this segmentation is obtained either from hand labeling of the training data, or using an already well trained HMM based recognizer. Hand labeling of the database is practical for small training sets. For large databases, Viterbi algorithm is applied to an already trained ASR in the forced alignment mode. The ANN training process is similar to the explanation in Section 2.2.1. In addition, similar to HMM/GMM systems, the hybrid HMM/ANN based ASR system can be trained by forward-backward training [62] or Viterbi

training [63]. In practice, the embedded Viterbi training with initial phone segmentation is used to train hybrid HMM/ANN systems. In the embedded Viterbi training, after each complete pass of ANN training, the segmentation is updated using Viterbi decoding in the forced alignment mode.

The advantages of the hybrid HMM/ANN approach for speech recognition are mainly due to the use of ANNs for estimating HMM emission probabilities. Therefore, hybrid HMM/ANN ASR benefits from discriminative training, model accuracy, better robustness to insufficient training data, and finally the ability to model acoustic correlation (using limited contextual inputs). Hybrid systems have been shown to have comparable or better performance to GMM-based systems for many corpora, and are argued to give simpler systems and training procedures.

Posterior probabilities have been also used as local scores for measuring the confidence level of the recognizer output. In many ASR applications, it would be useful to have a mechanism for measuring the confidence or correctness of the recognizer output. This process is called confidences measurement. In the confidence measurement process, typically a word or phone hypothesis is obtained from a lattice or recognizer output. A score representing the level of confidence on the correctness of this hypothesis is then computed using different techniques [8, 50, 64, 65]. ANNs can be suitable for confidence measurement since, by definition, posterior probabilities also measure the probability of being correct [50, 66]. There have been some research on using the output of ANNs (phone posterior probabilities) for confidence measurement [10, 50, 66, 67]. In these approaches, the outputs of the ANN are used directly as local scores to compute the confidences level for a phone or word hypothesis. The confidence level is typically estimated by accumulating the local phone posterior probabilities within the hypothesis boundary, followed by normalization with respect to the length of the hypothesis. These confidence measurement approaches are described in more detail in Chapter 5.

2.2.3 Posteriors as Features

MLP-estimated posterior probabilities can be also used as acoustic features for HMMs. In this case, the MLP is considered as performing some kind of "optimal" feature extraction (using nonlinear discriminant analysis). As described before, when ANNs are used to estimate the posterior probabilities of a set of phone units, they allow discriminative training in a natural and efficient manner. They also do not make strict assumptions about the statistics of input features, and have been found well able to cope with highly correlated and unevenly distributed features such as spectral energy features from several adjacent frames [1, 61]. In the case of multiple features (e.g., multi-band and multi-stream speech recognition), the MLP can also be used as a convenient way to integrate multiple features and generate the most compact and most discriminant representation to be used in standard HMMs.

The previously discussed hybrid HMM/ANN framework [1] replaces the GMM acoustic model with an ANN, discriminatively trained to estimate the posterior probabilities of each phone class given the data. Because of the different probabilistic basis (likelihoods versus posteriors) and different representations for the acoustic models (means and variances of mixture components versus network weights), techniques developed for one domain are often difficult to transfer to the other. The relative dominance of likelihood-based systems has resulted in the availability of sophisticated tools such as HTK [68] offering advanced, mature, and integrated system parameter estimation procedures. On the other hand, discriminative acoustic model training and certain combination strategies facilitated by the posterior representation are much more easily implemented within the hybrid framework. Combining ANN and GMM modeling within a single system holds the potential of combining the advantages of both systems, and several groups have pursued variants of this theme. The most popular approach among them is called Tandem [2]. In Tandem, these two approaches are successfully combined by using the output of an ANN classifier as the input features for the Gaussian mixture models of a conventional speech recognizer. An ANN classifier is first trained to estimate context-independent phone posterior probabili-

ties. The probability vectors are then treated as normal feature vectors and used as the input for a conventional HMM/GMM system. The resulting system which effectively has two acoustic models in tandem - first a neural network then a GMM - performs better than conventional HMM/GMM baselines.

The overall Tandem system is illustrated in Figure 2.4. First, a neural network acoustic model (a conventional MLP) is trained to estimate the posterior probabilities of each possible sub-word unit (particularly, context-independent phones). The network is trained by back-propagation with a minimum-cross-entropy criterion [59] to one-hot targets obtained from either hand labeling or a forced alignment of the training data generated using an earlier acoustic model. The input to the network is a context window of a few successive frames of the feature vector. Typically a context window of 9 frames, corresponding to 90 ms of audio at a 10 ms frame rate is used. The output of the neural network is a vector of posterior probabilities $p(q_t|x_t)$ at time t, with one element for each phone. Such a vector is generated for context windows centered on each input feature vector. In hybrid HMM/ANN approach, these would go directly to an HMM decoder to find the word sequence, but instead in Tandem approach they are used as the feature inputs for a HMM/GMM back-end module. Typically, the number of phones is between 30 and 50, so the total dimensionality of the feature space is the same as with normal acoustic features augmented by deltas and double-deltas. Because the posterior probabilities have a very skewed distribution, it is advantageous to warp them into a different domain, for instance by taking their logarithms. An alternative to this is to omit the final nonlinearity in the output layer of the neural network. Normally, the softmax nonlinearity (exponentials normalized to sum to 1) is used as the output nonlinearity. Skipping the softmax nonlinearity is very similar to taking the logarithm of the subsequent probabilities. The features constituted by the log-posteriors have the rather unusual property of tending to contain one large value (corresponding to the current phone) with all other values much smaller. Applying a global de-correlation via the Karhunen-Loeve (KL) transform improves system performance, presumably by improving the match of these features to the Gaussian mixture models. The result of the KL transformation is used as feature inputs for a conventional HMM based speech recognizer,

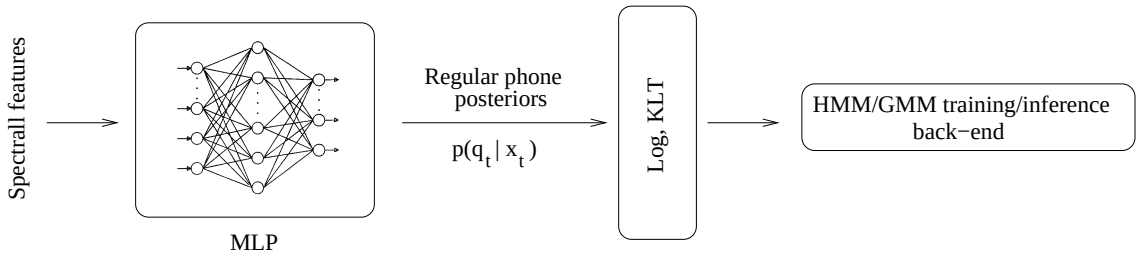


Figure 2.4: Standard approach for deriving and using Tandem features. The phone posterior vectors $p(q_t|x_t)$ are estimated using MLP. $p(q_t|x_t)$ is a vector of phone posterior probabilities at time t. These posteriors are gaussianized and decorrelated using log and KL transforms. The result of the transformation is used as acoustic features for training and inference in a standard HMM/GMM back-end.

which relearns the associations to sub-word units.

In Tandem system, the acoustic feature space is re-mapped by the discriminatively-trained ANN in such a way that regions around key phonetic boundaries are magnified, while regions that correspond to a single phone are compressed. This minimizes the effects of non-phonetic variations such as speaker characteristics and noise. This soft re-mapping retains some information from the original signal that the subsequent Gaussian mixture model in the HMM/GMM recognizer can usefully exploit. The gains of the Tandem approach arise from the combination of discriminative modeling (in this case via the ANN) which allows parameters to focus on critical regions, as well as generative distribution modeling by the GMMs, which are better suited to model a large number of classes.

Input to Tandem can be any data that are believed to provide a relevant evidence for the classification. In its simplest form mentioned, Tandem takes as input a super frame of typical speech features such as 9 frames of concatenated PLP static and dynamic features. Often, Tandem inputs are concatenated outputs from other sub-band classifiers such as TRAP [69], HATS [70] or MRASTA [71]. TRAP has been also reported to be efficient in combining different features and for alleviating irrelevant information [72] [73].

2.3 Summary and Conclusions

In this chapter, we have studied the fundamental components of the state-of-the art ASR systems: feature extraction, acoustic modeling and decoding. Feature extraction transforms the speech signal to acoustic features by removing the redundant information and preserving the information about the lexical content. The feature extraction methods are mainly based on spectral analysis of speech signal. Acoustic modeling module uses the acoustic features and statistical approaches to build acoustic models for sub-word (phone) units. HMMs are the most dominant tools for acoustic modeling in ASR. The decoder module combines the acoustic scores obtained from the acoustic model with the lexical and language model information to decode the word sequence in the utterance.

In the second part of the chapter, we have studied the state-of-the-art approaches for the estimation and use of posterior probabilities as more discriminant evidences in ASR. We studied ANNs as the most dominant tool for local phone posterior estimation, and we reviewed the potential advantages of using ANN models in ASR. We described the use of posterior probabilities as features, as well as local scores. The hybrid HMM/ANN framework uses the posteriors as state emission probabilities in the HMM configuration. Tandem approach uses phone posteriors as acoustic features for training and inference in a HMM/GMM back-end module. Both hybrid and Tandem approaches take the advantage of discriminant training, model accuracy and context sensitivity of ANNs. The estimation of phone posteriors in these approaches is only based on information in a limited number of frames. However, phone information is spread over long context in the utterance. In addition, there are some knowledge about duration of phones and their lexical usage in the words. In the next chapter, we propose some approaches for enhancing local posterior estimates by integrating phonetic/lexical and long contextual information.

Chapter 3

Enhancing Posterior Probability Estimation

In the previous chapter, we have studied the estimation and use of local[1] phone posterior probabilities in ASR systems. As we have seen, the estimation of these posteriors is based only on one or limited number of spectral feature frames. In this thesis, we refer to these posteriors as "MLP posteriors" or "regular posteriors". However, limited spectral information is not the only available source of knowledge about phones. There are other sources of knowledge which can be taken into account to estimate more informative phone posteriors. Information about phones is spread over long temporal context and there are no sharp boundaries between phones, therefore taking into account long contextual information can be useful. Information about the underlying sub-word (phone) classes extends at least over an interval of 200-300 ms. This has been demonstrated in [5] and [6] by studying the mutual information between speech features over the time. Since the derived features will be

[1]Here we emphasis again on the definition of 'local' posterior probabilities. A local posterior is a posterior probability which is estimated for the current local frame, and not for a phone/word segment or hypothesis. However, as we will see in this chapter, the information for the estimation of this local posterior can be obtained from a long context or other sources of knowledge. Therefore, the term 'local' refers to the event for which the posterior is estimated, not the conditions. We will often drop the word 'local' keeping in mind that all the posteriors are estimated for a local frame, unless otherwise mentioned.

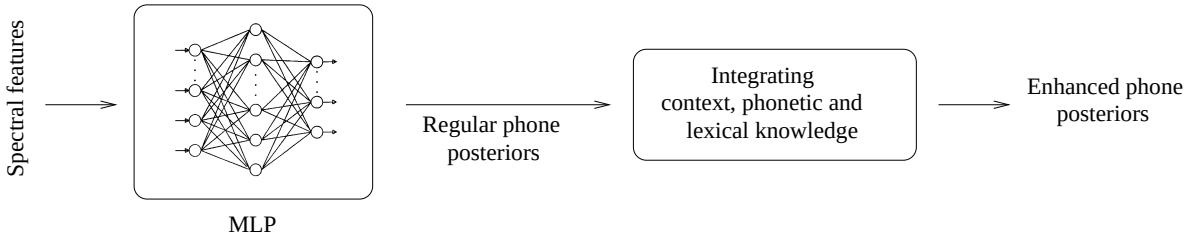


Figure 3.1: General idea: First, regular phone posteriors are estimated using an MLP, then these posteriors are post-processed in a secondary module to integrate context, phonetic and lexical knowledge. This results in enhanced phone posterior estimates.

used for classification into phone classes, it is beneficial to collect the evidence from all the data points which carry the information. In general, any evidence of the way in which the information about underlying linguistic process is distributed in the signal is of importance and can be useful. With the same motivation, some linguistic knowledge such as duration of phones (phonetic knowledge) and the lexical use of phones in a word can be useful for improving local posterior estimates.

There have been few recent studies with the goal of integrating context and prior linguistic knowledge in the posterior estimation [8, 9, 10]. In these studies, different methods for estimating posterior probability of a word hypothesis, given all acoustic observations of the utterance are proposed. These posteriors are estimated on HMMs or word graphs by the forward-backward (Baum-Welch) algorithm [7], and used for word confidence measurement. These studies are mainly focused on estimating word posteriors for the purpose of hypothesis confidence measurement.

In this chapter, we present a principled framework for enhancing the estimation of posteriors (particularly phone posteriors) by integrating long acoustic context, as well as phonetic and lexical knowledge. However, as opposed to the above approaches, the goal here is to provide local enhanced posteriors which can be used in frame synchronous posterior based ASR systems. The input in our approaches is regular phone posteriors estimated by an MLP, and the outcome is "local enhanced posteriors" of *phones*[2] at the *frame* level. Many

[2]Although as it is shown in Chapter 7, we can also use our approach for local word posterior estimation.

posterior based ASR algorithms are based on local phone posteriors. Therefore, the resulting frame based enhanced posteriors can be used in a wide range of posterior based ASR systems (e.g. Tandem and hybrid HMM/ANN), as replacement or in combination with the regular MLP posteriors in a straightforward manner. The general idea is illustrated in Figure 3.1. The regular phone posteriors estimated by a neural network (MLP) are post-processed by a secondary module to integrate context, phonetic, and lexical knowledge.

We propose two approaches for integrating phonetic, lexical and contextual knowledge in the posteriors estimation. The first approach uses a HMM to integrate the prior phonetic and lexical knowledge. The phonetic and lexical knowledge is encoded in the topology of the HMM. The integration is realized by using the regular MLP posteriors as emission probabilities in the HMM forward-backward recursions (Baum-Welch approach) [7]. This yields new enhanced posterior estimates taking into account the encoded knowledge in the topology of the HMM. The second approach uses a secondary neural network (MLP) to post-process a temporal context of regular phone posteriors, and learn long term intra and inter dependencies between regular phone evidences (posteriors) estimated initially by the first MLP. These long term dependencies are phonetic knowledge. The learned phonetic knowledge is integrated in the phone posterior estimation, during the inference (forward pass) of the second MLP, resulting in enhanced posteriors.

The proposed approaches provide a general framework for integrating acoustic context and different phonetic/lexical knowledge for improving posterior estimation in ASR, from state up to the phone and word units. In the following, we study these two approaches.

3.1 HMM-based Integration of Prior and Contextual Knowledge

Topological constraints in a HMM encode specific prior phonetic and lexical knowledge. For instance, modeling phones with a minimum number of states imposes the knowl-

edge about duration of phones, or left-to-right connection of phone models imposes specific lexical knowledge. This knowledge can be integrated in the regular MLP posteriors to get an enhanced version of these posterior estimates. This objective can be formulated as turning the regular estimate of phone posteriors $p(q_t^i|x_t)$ obtained by MLP, to a more informative posterior $p(q_t^i|x_{1:T}, M)$, where q_t^i is the event of having phone i at time t, $x_{1:T} = \{x_1, ..., x_t, ..., x_T\}$ is the acoustic context as available possibly in the whole utterance, and M is HMM model encoding specific prior knowledge. We have used HMM/ANN formalism for integrating HMM topological constraints in the MLP posterior estimates. The integration is done by using phone posteriors $p(q_t^i|x_t)$ as state emission probabilities in the HMM. Each state s^k of the set of HMM states $S = \{s^1, ..., s^k, ..., S^{N_s}\}$ (N_s total number of HMM states) is associated with one of MLP outputs representing a phone posterior probability. The state emission probabilities are used in HMM forward-backward recursions [74] to integrate HMM topological constraints (encoding specific prior knowledge). This gives the estimates of HMM state posteriors $p(s_t^k|x_{1:T}, M)$, where s_t^k is the event of having state k at time t. The state posteriors will then be integrated to enhanced phone posteriors $p(q_t^i|x_{1:T}, M)$ by accumulating posteriors of all the states modeling phone i in the HMM. In the forward-backward recursions and state posterior estimation, we have the contribution of the HMM topological constraints (prior knowledge) in addition to the MLP posteriors (emission probabilities). Therefore, the state posteriors (and consequently phone posteriors) can be interpreted as the integration of topological constraints (prior knowledge) in the MLP posteriors. In the following, we first review the forward-backward recursions for conventional likelihood based HMM systems, then we study forward-backward recursions for the case of modeling state probability distributions with MLP outputs.

According to the standard HMM formalism, the state posterior is defined as the probability of being in state k at time t, s_t^k, given the whole observation sequence $x_{1:T}$ and the HMM model M encoding specific prior knowledge (topological/temporal constraints):

$$\gamma(k, t) = p(s_t^k|x_{1:T}, M) \tag{3.1}$$

where x_t is a feature vector at time t, $x_{1:T} = \{x_1, \ldots, x_t, \ldots, x_T\}$ is an acoustic observation sequence, s_t is the HMM state at time t, which value can range from 1 to N_s (total number of HMM states), and s_t^k shows the event "$s_t = k$". In the following, we will often drop the M, keeping in mind that all recursions are processed through some prior (Markov) model M. We call $\gamma(k,t)$ as "state posterior".

The state posteriors $\gamma(i,t)$ can be estimated using forward α and backward β recursions (as referred to in HMM formalism) [74] using local emission likelihoods $p(x_t|s_t^k)$:

$$\begin{aligned} \alpha(k,t) &= p(x_{1:t}, s_t^k) \\ &= p(x_t|s_t^k) \sum_j^{N_s} p(s_t^k|s_{t-1}^j)\alpha(j,t-1) \qquad (3.2) \\ \beta(k,t) &= p(x_{t+1:T}|s_t^k) \\ &= \sum_j p(x_{t+1}|s_{t+1}^j)p(s_{t+1}^j|s_t^k)\beta(j,t+1) \qquad (3.3) \end{aligned}$$

thus yielding the estimate of $p(s_t^k|x_{1:T}, M)$:

$$\gamma(k,t) = p(s_t^k|x_{1:T}, M) = \frac{\alpha(k,t)\beta(k,t)}{\sum_j \alpha(j,T)} \qquad (3.4)$$

Similar recursions, also yielding to "state posteriors", can also be developed for systems based on local posterior probabilities, such as hybrid HMM/ANN systems using MLPs to estimate HMM emission probabilities [1]. Each HMM state s^k is associated with one MLP output $p(q^i|x_t)$ representing posterior probability for phone i over time. In standard HMM/ANN systems, these local posteriors are usually turned into "scaled likelihood" by dividing MLP outputs by their respective a priori probability $p(q_t^i)$, as estimated on the training data, i.e. $\frac{p(q_t^i|x_t)}{p(q_t^i)}$. The scaled likelihoods are used as state emission probabilities in

HMM/ANN ASR. For HMM state k at time t associated with the phone i we have:

$$\frac{p(x_t|s_t^k)}{p(x_t)} = \frac{p(s_t^k|x_t)}{p(s_t^k)} \tag{3.5}$$

The scaled likelihood at the left hand side of (3.5) is used in standard HMMs since, during recognition, $1/p(x_t)$ is simply a normalization factor independent of the state s_t^k.

In [62], it was shown that these scaled likelihoods can be used in "scaled alpha" $\alpha^{scale}(k,t)$ and "scaled beta" $\beta^{scale}(k,t)$ recursions to yield state posterior estimates.

To use scaled likelihoods, we start by defining scaled α as:

$$\alpha^{scale}(k,t) = \frac{p(x_{1:t}, s_t^k)}{\prod_{\tau=1}^{t} p(x_\tau)} \tag{3.6}$$

We note here that this is simply a *definition*. Thus, the product in the denominator does not imply that we have made any explicit temporal independence assumption. In fact, all the recursions used below, will never make any additional temporal independence assumption than the usual state conditional independence assumption.

Starting from (3.5), we can express the scaled α recursion as follows:

$$\begin{aligned}
\alpha^{scale}(k,t) &= \frac{p(x_t|s_t^k)}{p(x_t)} \sum_j p(s_t^k|s_{t-1}^j) \frac{p(x_{1:t-1}, s_{t-1}^j)}{\prod_{\tau=1}^{t-1} p(x_\tau)} \\
&= \frac{p(x_t|s_t^k)}{p(x_t)} \sum_j p(s_t^k|s_{t-1}^j) \alpha^{scale}(j, t-1) \\
\alpha^{scale}(k,t) &= \frac{p(s_t^k|x_t)}{p(s_t^k)} \sum_j p(s_t^k|s_{t-1}^j) \alpha^{scale}(j, t-1)
\end{aligned} \tag{3.7}$$

Similarly, we can define the scaled β and β recursion as follows:

$$\begin{aligned}\beta^{scale}(k,t) &= \frac{p(x_{t+1:T}|s_t^k)}{\prod_{\tau=t+1}^{T} p(x_\tau)} \\ &= \sum_j \frac{p(s_{t+1}^j|x_{t+1})}{p(s_{t+1}^j)} p(s_{t+1}^j|s_t^k)\beta^{scale}(j,t+1)\end{aligned} \tag{3.8}$$

Given that all values required in (3.7) and (3.8) are available from the MLP output, another estimate of the state posteriors $p(s_t^k|x_{1:T},M)$, denoted here as $\gamma^{scale}(k,t)$, can thus be obtained as:

$$\begin{aligned}\gamma^{scale}(k,t) &= p(s_t^k|x_{1:T},M) \\ &= \frac{p(s_t^k,x_{1:T})}{p(x_{1:T})} \\ &= \frac{p(x_{t+1:T}|s_t^k)p(s_t^k,x_{1:t})}{p(x_{1:T})} \\ &= \frac{p(x_{t+1:T}|s_t^k)p(s_t^k,x_{1:t})\prod_{\tau=1}^{T} p(x_\tau)}{p(x_{1:T})\prod_{\tau=1}^{T} p(x_\tau)} \\ &= \frac{p(x_{t+1:T}|s_t^k)p(s_t^k,x_{1:t})\prod_{\tau=1}^{T} p(x_\tau)}{p(x_{1:T})\prod_{\tau=1}^{t} p(x_\tau)\prod_{\tau=t+1}^{T} p(x_\tau)} \\ &= \frac{\alpha^{scale}(k,t)\beta^{scale}(k,t)\prod_{\tau=1}^{T} p(x_\tau)}{p(x_{1:T})} \\ &= \frac{\alpha^{scale}(k,t)\beta^{scale}(k,t)\prod_{\tau=1}^{T} p(x_\tau)}{\sum_j p(x_{1:T},s_t^j)} \\ &= \frac{\alpha^{scale}(k,t)\beta^{scale}(k,t)}{\sum_j \alpha^{scale}(j,T)}\end{aligned} \tag{3.9}$$

Again, in theory, we have:

$$\gamma(k,t) = \gamma^{scale}(k,t) = p(s_t^k|x_{1:T},M) \tag{3.10}$$

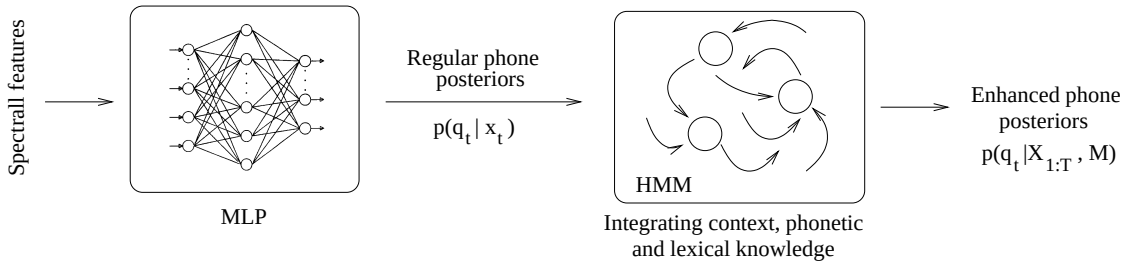


Figure 3.2: HMM-based enhanced posterior estimation: First, regular phone posterior vectors $p(q_t|x_t)$ are estimated using an MLP. These posteriors are used as emission probabilities in HMM recursions to estimate state posteriors. The HMM state posteriors are then integrated into enhanced phone posterior vectors $p(q_t|x_{1:T}, M)$.

In this work, we always use hybrid HMM/ANN configuration for the estimation of HMM state posterior probabilities. This means that the MLP posteriors (after turning to scaled likelihoods), are used as emission probabilities in the forward-backward recursions. All the computations (including forward-backward recursions) are implemented using logarithmetic to avoid numerical errors.

The estimated state posteriors are then used to estimate phone posteriors. The enhanced phone posteriors $p(q_t^i|x_{1:T,M})$ can be expressed in terms of state posteriors $\gamma(k,t)$ as follows:

$$\begin{aligned} p(q_t^i|x_{1:T}, M) &= \sum_{k=1}^{N_s} p(q_t^i, s_t^k|x_{1:T}, M) \\ &= \sum_{k=1}^{N_s} p(q_t^i|s_t^k, x_{1:T}, M)p(s_t^k|x_{1:T}, M) \\ &= \sum_{k=1}^{N_s} p(q_t^i|s_t^k, x_{1:T}, M)\gamma(k,t) \end{aligned} \tag{3.11}$$

where $p(q_t^i|x_{1:T}, M)$ is the enhanced phone posterior for phone i at time t. Probability $p(q_t^i|s_t^k, x_{1:T}, M)$ represents the probability of being in a given phone i at time t knowing to be in the state k at time t. If there is no parameter sharing between phones, this is deterministic and equal to 1 or 0. Otherwise, this can be estimated from the training

data. In this work, we assume that there is no parameter sharing between phones, thus a phone posterior is estimated by adding up all state posteriors associated with the phone in the whole model. This way, the new enhanced phone posterior estimates $p(q_t^i|x_{1:T}, M)$ integrating context and prior knowledge is obtained. In the remainder of the thesis, we call them as "HMM-based enhanced posteriors". The estimation of HMM state posteriors through forward-backward recursions is typically known for training HMM parameters in EM process. The state posteriors are used as hidden variables during the EM process to estimate HMM parameters. However, in this work the main novelty is using the estimated state/phone posteriors as features or scores for further decoding/training in another hierarchical module. As we will study in chapters 4 and 5, the state/phone posteriors can be used as features (instead of spectral features) for training a HMM/GMM back-end in Tandem system, as well as local scores for decoding in hybrid HMM/ANN system.

Figure 3.2 is showing the configuration for the HMM-based integration of prior and contextual knowledge. As it is shown, the regular phone posterior vectors $p(q_t|x_t)$ are initially estimated using an MLP. $p(q_t|x_t)$ is a vector of phone posteriors at time t with the components $p(q_t^i|x_t)$ for $i \in \{1, ..., i, ..., N_q\}$. These phone posteriors are turned into scaled likelihoods (by dividing them by the corresponding priors), and used as emission likelihoods in the HMM. The HMM state posteriors are estimated using HMM forward-backward recursions. The state posteriors are then integrated to enhanced phone posteriors $p(q_t|x_{1:T}, M)$. $p(q_t|x_{1:T}, M)$ is a vector of enhanced phone posteriors at time t. The obtained phone posteriors are more informative (enhanced) than regular MLP posteriors, since the prior knowledge (encoded in the topology of the HMM), and long acoustic context (as available in the whole utterance) is additionally taken into account to estimate them. In fact, the second module (the HMM) gets phone initial evidences (MLP posteriors) as input, and acts as a corrective filter by introducing context and prior knowledge. The corrective filter suppresses the effect of evidences not matching with prior knowledge or contextual information, and magnifies the effect of evidences matching them. The output of this corrective filter is enhanced evidences in the form of posteriors.

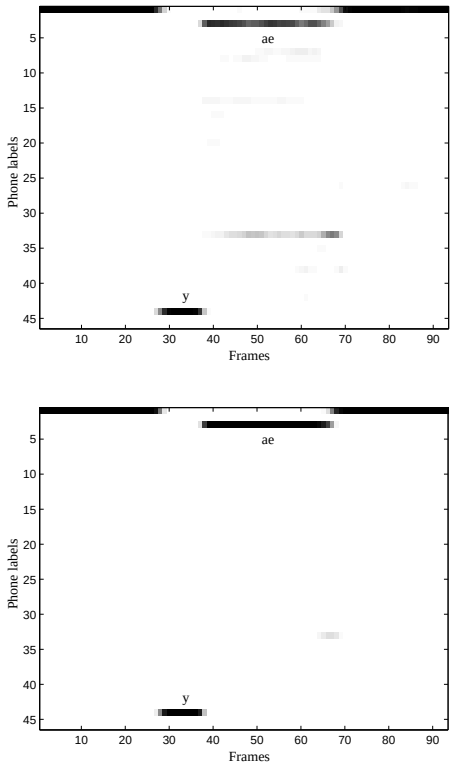
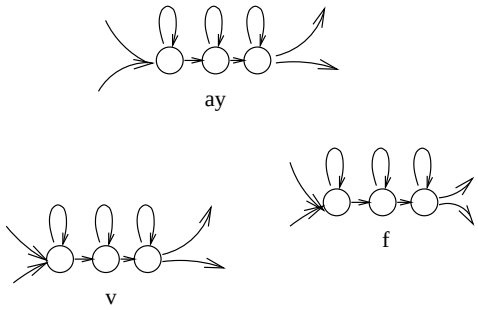


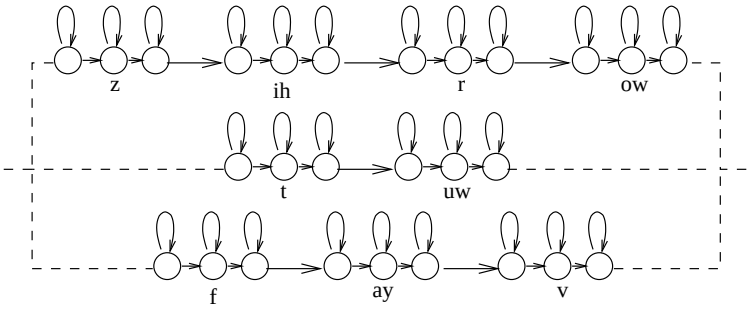
Figure 3.3: (top) MLP estimated phone posteriors, and (bottom) corresponding enhanced phone posteriors for the word 'yeah'. The y-axis is showing phone labels and x-axis is showing frames. Intensity of each block shows the posterior value. The enhanced posteriors look less noisy.

Figure 3.3 is showing a sample of regular MLP posteriors and corresponding enhanced posteriors obtained by integrating phone duration information. The enhanced posterior estimates are less noisy. The MLP posteriors at the top are used as local estimators (emission probabilities) in the HMM estimating enhanced posteriors (bottom).

The HMM module used for enhanced posterior estimation can have different topologies, thus encoding different types of prior knowledge. As the simplest case, phones can be modeled with a minimum number of states, and be connected using uniform transition probabilities (Figure 3.4.a). In this case, only the prior phonetic knowledge about minimum duration of phones is introduced in the posterior estimation. Next step is using non-uniform phone transitions estimated from a labeled data, instead of uniform transi-



a: Phonetic duraion knowledge



b: Lexical knowledge

Figure 3.4: (a) HMM configuration for integrating phonetic duration knowledge. Phones are modeled with a minimum number of states, and phone models are connected using uniform transitions. (b) HMM configuration for integrating phonetic and lexical knowledge. Word models are included in the HMM configuration.

tions. Finally, we can have a fully constrained model composed of connected word models and phone models (Figure 3.4.b). This topology integrates full phonetic and lexical knowledge in the posterior estimation. For the experiments in Chapters 4 and 5, the HMM-based enhanced posteriors are obtained by integrating the knowledge about minimum phone duration. This is achieved by using an HMM model with a topology similar to Figure 3.4.a. In this topology, phones are modeled with left-to-right connection of 3 states, imposing the constraint that every phone has to take at least 3 frames. The phone models are connected by uniform transition probabilities. On the other hand, in the experiments of Chapters 7 and 8, we have integrated certain lexical knowledge. The HMM topology for integrating the lexical knowledge is similar to Figure 3.4.b. In this topology, phone models are left-to-right

connected based on their lexical use in the words.

Although in this chapter we only study phone posterior estimation, this posterior estimation/integration approach provides a theoretical framework for hierarchical estimation, integration and use of posteriors, from the state level up to the phone and word levels. Local word posteriors can be estimated basically in the same way as state posteriors are integrated into phone posteriors. More details on word posterior estimation are studied in Chapter 7.

Besides the advantages of integrating prior knowledge for enhancing posterior estimates, it should be noticed how and to what extent the knowledge is reliable. Although the prior knowledge is assumed to be correct, but as the name "prior" suggests, there can be few cases in which the true data is not matching the prior knowledge. For example, the assumed lexical knowledge may not include some rare but truly existing pronunciation variants for a word, while such cases may appear in data. In these cases, the enhanced posteriors start deviating from the MLP posteriors and they may not represent the data correctly. Therefore, although prior knowledge helps to improve the estimation of posteriors, there can be some cases that the resulting posteriors are not matching the data. This means there is a trade-off between the smoothness obtained by integrating prior knowledge, and deviation from data. Considering this potential issue, as it is studied in Section 4.1, we propose to use HMM-based enhanced posteriors in combination with the original MLP posteriors. In this way, information in both posterior streams are preserved. A more detailed explanation will be given in Section 4.1.

3.2 MLP-Based Integration of Phonetic and Contextual Knowledge

In Section 3.1, we have studied the integration of phonetic and lexical knowledge (encoded in HMM topology) in the posterior estimation. The HMM topology specifies the linguistic

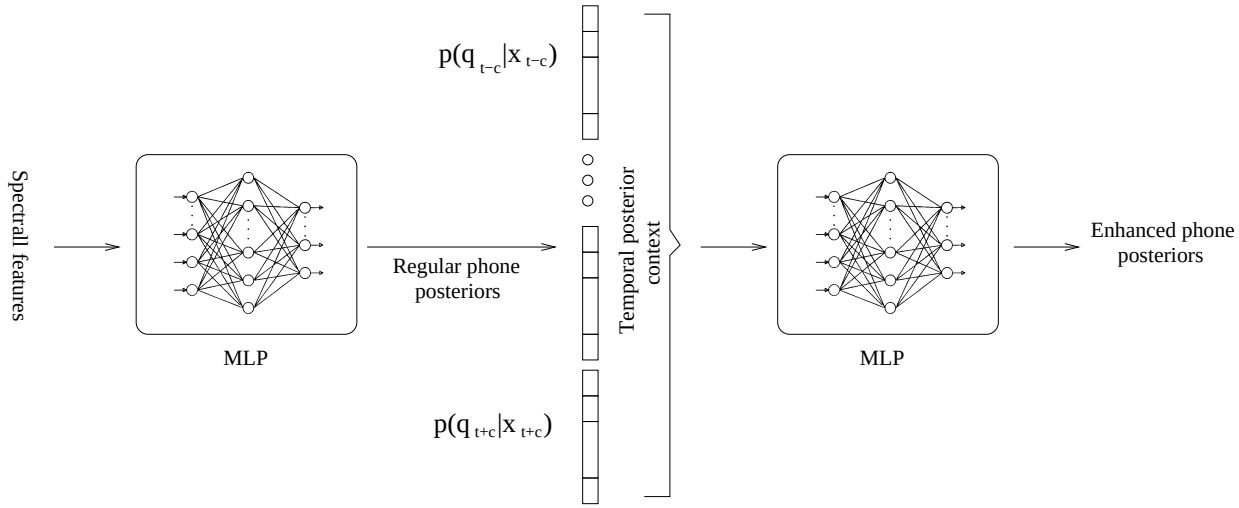


Figure 3.5: MLP-based enhanced phone posterior estimation: The first MLP is transforming acoustic (cepstral) features to regular phone posteriors. A temporal context of phone posteriors is made by concatenating posterior vectors in $\{p(q_{t-c}|x_{t-c}), ..., p(q_t|x_t), ..., p(q_{t+c}|x_{t+c})\}$. $p(q_{t-c}|x_{t-c})$ is a vector of phone posteriors at time $t-c$. The second MLP processes the temporal context of regular phone posteriors, and learns long term dependencies between phone evidences. These dependencies are phonetic knowledge. During the inference (forward pass of the second MLP), the learned knowledge is integrated in the posterior estimation, resulting in enhanced posteriors.

knowledge based on the prior assumptions about phones duration and the lexical usage of phones in the words. The alternative to this prior assumptions is learning the knowledge from data. In this section, we study a second approach for integrating phonetic knowledge which realizes the idea of learning phonetic knowledge from data. We use a secondary ANN to learn long term inter and intra dependencies between phone evidences (posteriors) in the training data. The configuration is shown in Figure 3.5. We have two MLPs in this configuration. The first MLP performs the regular phone posterior probability estimation by transforming a small context of acoustic features (cepstral features) to phone posteriors. The input to the second MLP is a temporal context of phone posteriors estimated by the first MLP, i.e. $\{p(q_{t-c}|x_{t-c}), ..., p(q_t|x_t), ..., p(q_{t+c}|x_{t+c})\}$, where '$c$' shows a temporal context (typically 7-9). To form this input, the posterior vectors in the mentioned temporal context are concatenated. The output of the second MLP is enhanced phone posteriors for the same

set of phones as the first MLP. The phonetic class is defined with respect to the center of the temporal context. The first MLP is typically trained with the cepstral features as input and phone targets as output, while the second MLP is trained with a long context of phone posteriors as input and the same phone targets as output. The same database is used for training the two MLPs. The first MLP learns the transformation form acoustic features to phone evidences, while the second MLP gets the phone evidences as input and learns long term dependencies between phone evidences. These long term phone dependencies are phonetic information, such as phone trajectory shape, co-articulation between phones, and phone duration information. Therefore, the second MLP learns phonetic knowledge from data, and integrates this knowledge in the phone posterior estimation during the inference (forward pass). This leads to enhancement of phone posteriors. The rational behind this is that at the output of every MLP, the information stream gets simpler (converging to a sequence of binary posterior vectors), and can thus be further processed (using a simpler classifier) by looking at a larger temporal window. In the remainder of this thesis, we call the posteriors at the output of the second MLP as "MLP-based enhanced posteriors".

We have experimentally analyzed the role of the second ANN in the hierarchy [75, 76]. The mapping function which is learned by the MLP is nonlinear, thus the analysis of second MLP role is not straightforward. A single layer Perceptron (SLP) can be a reasonable approximation for investigating the role of the second MLP, and can be considered as a multi-dimensional linear matched filter for temporal trajectories of phone posteriors [77]. Therefore, we replace the second MLP with a SLP, in order to analyze the role of the second ANN in the configuration shown in Figure 3.5. The single layer Perceptron can be viewed as a multi-dimensional matched filter derived jointly for all the phones by minimizing an error criteria. Figure 3.6 shows the matched filters for the phones /iy/ and /b/. The matched filters are obtained from the weights of the trained SLP. The weights of the SLP are the coefficients of the linear mapping function. The vertical axis shows the coefficients of the linear mapping function (weights of SLP), and the horizontal axis shows the temporal context in frames (with zero indicating current frame). The width of the matched filter captures the duration of the phone (contribution across time), and height captures

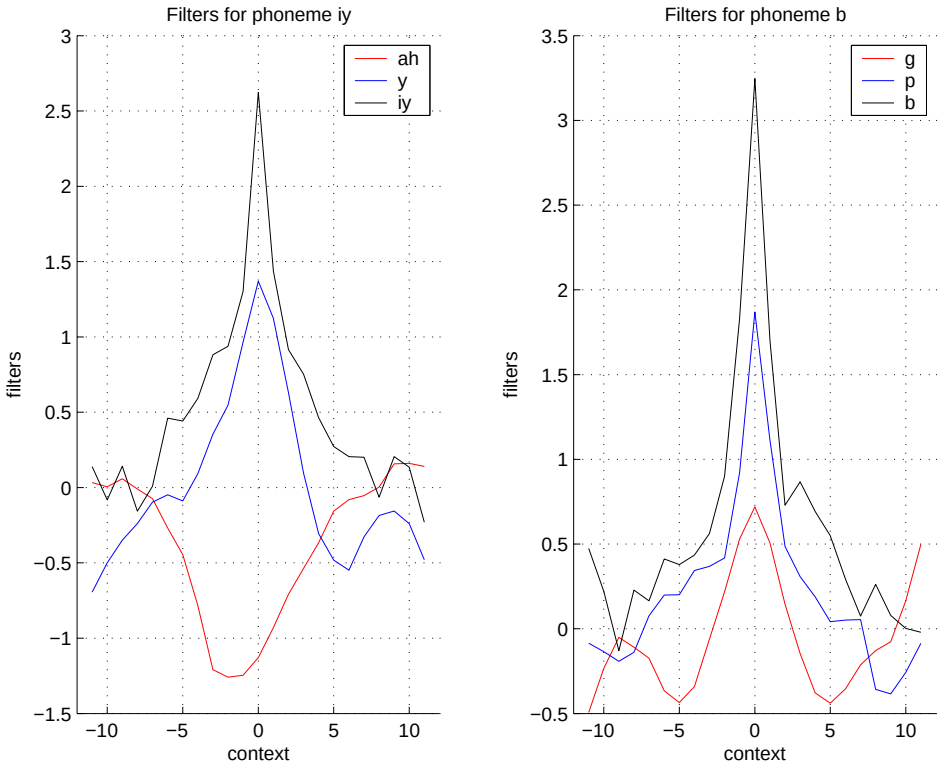


Figure 3.6: The matched filters for phones /iy/ and /b/. The plot also shows top three contributing phones in the filter. The SLP matched filter of a phone (e.g. /iy/) captures the contribution of different phone posteriors at the input of the SLP (in the window duration of 20 frames) to the posterior probability of phone /iy/. Phone /iy/ has a negative contribution from the phone /ah/. In the matched filter for the phone /b/, there is a contribution from phones /p/ and /g/, which is consistent with the production of /b/.

the contribution of different phones in the production of the current phone (contribution across phones). The analysis of the matched filters obtained after training the SLP shows that the matched filter for a specific phone (e.g. /iy/) captures the contribution of different regular phone posteriors at the input of SLP to estimate the posterior probability of the phone /iy/. These contributions are consistent with the production of this phone. The analysis indicates that the second ANN has learned the long term inter an intra dependencies between the regular posteriors. These dependencies are mostly phonetic information such as phone posterior trajectory shape, co-articulation between phones, and phone duration information.

Figure 3.7 is showing an example of initial and corresponding enhanced posteriors. The enhanced (second MLP) posteriors are less noisy than the initial (first MLP) posteriors. The second MLP acts as a filter which smooth out evidences not matching the learned phonetic knowledge. Ideally, this approach can be used for post-processing the output of any posterior estimator to integrate higher level knowledge (e.g. phonetic knowledge).

In the MLP-based integration of the phonetic and lexical knowledge, the risk of using the knowledge which is not matching the reality of data is less than HMM-based integration. It is due to the fact that the knowledge is learned from the data, instead of being obtained from prior assumptions. This leads to some differences in the way we use HMM-based and MLP-based enhanced posteriors for speech recognition systems. It will be studied in more detail in Chapters 4 and 5.

In this work, the second MLP has been trained on the same database as the first MLP. An alternative (although not experimented here) will be to use the second MLP for (task) adaptation purposes. For instance, the first MLP can be trained on a general English database, while the second MLP is trained on a second database of specific accent or dialect. In this case, the first MLP acts as a general phone posterior estimator, and the second MLP adapts the posterior estimation for the specific task.

3.3 Summary and Conclusions

In this chapter, we studied the integration of phonetic/lexical and contextual knowledge in the local posterior estimation. We discussed that the regular estimation of local posteriors does not take into account long contextual and phonetic/lexical knowledge. In both hybrid HMM/ANN and Tandem approaches, posteriors are estimated using ANNs (more specifically MLPs), based only on the acoustic information in a local frame or a limited number of local frames. However, information about phones are extended over long temporal context. Phones have specific duration constraints (phonetic knowledge), follow specific sub-lexical

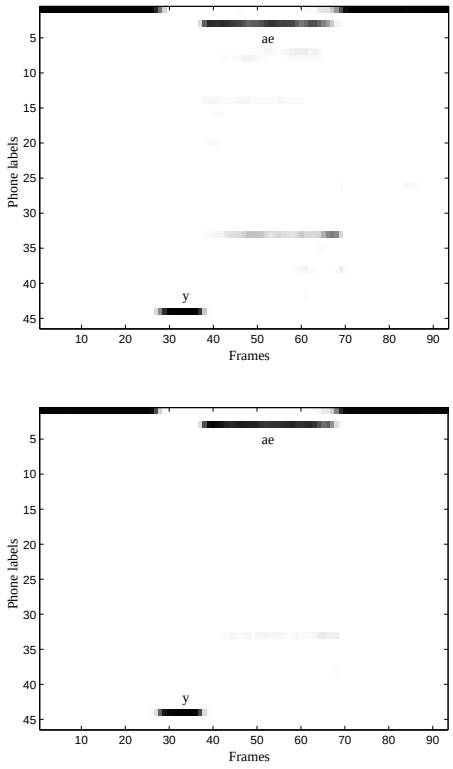


Figure 3.7: (top) Initial posteriors estimated by the first MLP, and (bottom) enhanced phone posteriors estimated by the second MLP, integrating phonetic knowledge. The utterance contains the word 'yeah'. The y-axis is showing phone labels and x-axis is showing frames. The intensity inside each block is showing the posterior value. The new enhanced posteriors are less noisy.

and lexical rules (lexical knowledge), etc. These long contextual and linguistic sources of knowledge can help in providing more informative phone posterior estimates. In this chapter, we have presented a principled framework for enhancing the estimation of posteriors by integrating long acoustic context, as well as phonetic and lexical knowledge.

We proposed and discussed two approaches for integrating context and phonetic/lexical knowledge. The first approach uses a HMM module to integrate this additional knowledge. The knowledge is encoded in the topology of the HMM. The regular MLP posteriors are used in HMM forward-backward recursions to integrate context and prior phonetic/lexical knowledge, yielding the enhanced phone posterior estimates. In the second approach, a

secondary MLP is used to post-process a temporal context of regular MLP posteriors, and learn long term dependencies between these posteriors. These long term dependencies are phonetic knowledge. During the inference (forward pass of the second MLP), the learned knowledge is integrated in the phone posterior estimation, resulting in enhanced phone posteriors at the output of the second MLP. In the HMM-based enhanced posterior estimation, the prior phonetic/lexical knowledge is provided by our assumptions about duration of phones and the lexical use of phones in the words. On the other hand, in the MLP-based enhanced posterior estimation, the knowledge is learnt from data.

The resulting local enhanced posteriors can be used in a wide range of frame-synchronous posterior based ASR systems (e.g. Tandem and hybrid HMM/ANN), as replacement or in combination with the regular MLP posteriors in a straightforward manner.

Chapter 4

Enhanced Posteriors As Features

As discussed in Section 2.2.3, local posterior probabilities have been used as more discriminant features in speech recognition systems. The most well known sample of these systems is Tandem [2]. In Tandem approach, posterior probabilities are used as features for training and inference in a standard HMM/GMM back-end. As the enhanced phone posteriors are estimated at every frame, they can be also used in frame synchronous posterior based ASR systems such as Tandem. In this chapter, the use of the enhanced posteriors as features in Tandem configuration is investigated. We propose new Tandem configurations for HMM-based and MLP-based enhanced phone posteriors. We show that using the enhanced posteriors as features, or as complementary features can improve the performance of Tandem system. Since HMM-based and MLP-based enhanced posteriors have different properties, we study their cases separately.

In the following, Section 4.1 describes the use of the HMM-based enhanced posteriors as features. The performance of the HMM-based posteriors is studied in terms of their consistency, frame recognition and word recognition. We show that the HMM-based enhanced posteriors can be used in combination with the regular MLP posteriors for improving the performance of Tandem configuration. Section 4.2 describes the same studies for the MLP-

based enhanced posteriors. The MLP-based enhanced posteriors can be used instead or in combination with the regular MLP posteriors for improving Tandem configuration.

4.1 HMM-Based Enhanced Posteriors

In Section 3.1, we have studied the integration of prior and contextual knowledge using a HMM. This integration leads to estimating more informative posteriors. We also mentioned to the issue of integrating partially incorrect prior knowledge leading to deviation from the data. Considering this, a safe compromise is using the enhanced posteriors as complementary features along with the original MLP posteriors. In the other words, the enhanced posteriors should be combined with the MLP posteriors. Considering a configuration similar to Tandem, the combined evidences are then used as features for training and inference in a HMM/GMM back-end. In this way, the raw evidences (MLP posteriors) representing the data are preserved, while there is also access to the posteriors enriched by the prior knowledge and context. We have studied addition (average) and concatenation as the combination rules. In case of addition (average), the combined evidence is written as:

$$Comb_t^i = \frac{p(q_t = i|x_t) + p(q_t = i|x_{1:T}, M)}{2} \tag{4.1}$$

where $Comb_t^i$ shows the combined evidence for phone i at frame t. In case of concatenation rule, the MLP and enhanced posterior vectors at frame t are concatenated. The dimension of the resulting vector is reduced by applying KLT transform.

Figure 4.1 is showing a diagram of the normal Tandem system using MLP posteriors as features, and Tandem system using enhanced posteriors as complementary to the MLP posteriors. The emission probabilities in the HMM module which integrates prior knowledge are provided by the MLP. The enhanced posteriors are obtained by post-processing MLP posteriors in the HMM to integrate prior and contextual knowledge. In the following

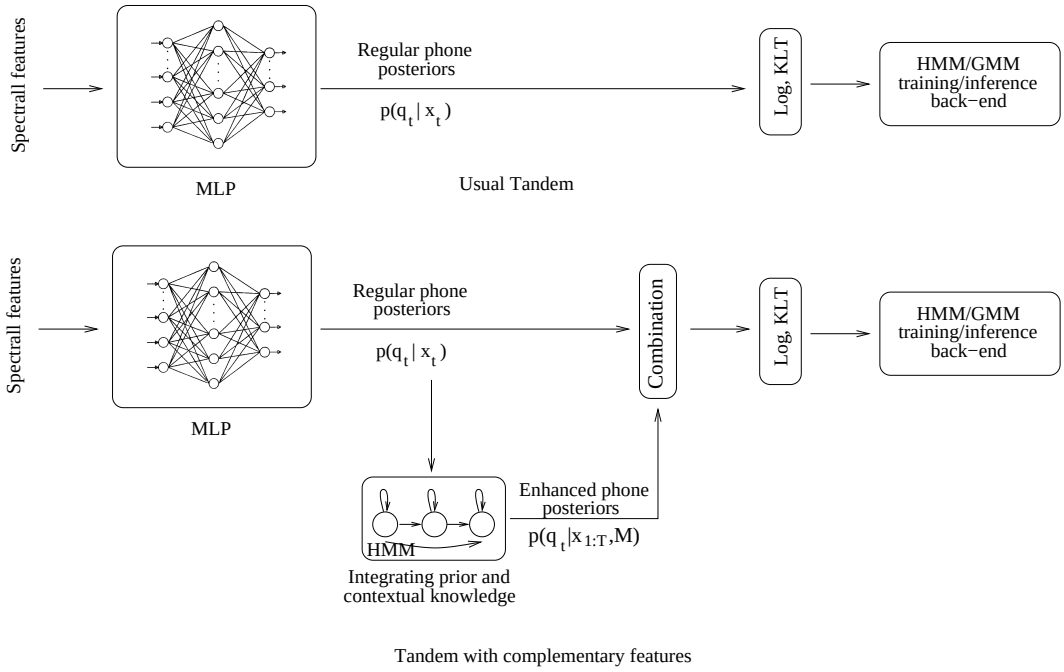


Figure 4.1: (top) Usual Tandem, and (bottom) Tandem system using enhanced posteriors as complementary features. Usual Tandem uses MLP posteriors (after some transformations) as features. The new Tandem system uses a combination of the MLP and enhanced posteriors as features. In the new Tandem configuration, enhanced posteriors are estimated using a HMM module integrating phone duration information. The enhanced posteriors are then combined with the MLP posteriors, some transformations applied, and the resulting features are used for training and inference in a HMM/GMM back-end.

experiments, the knowledge integrated by the HMM is minimum phone duration information. The topology of this HMM consists of phones modeled with 3 states, imposing a minimum phone duration of 3 frames. The phone models are connected with uniform transitions. A minimum phone duration equal to 3 frames is a usual assumption in many traditional speech recognition systems, and also matches the phone duration statistics we obtained from the phonetic segmentation of data. Longer or shorter minimum durations degrade or do not improve the results. We have also tried other topologies integrating different types of prior knowledge. The minimum phone duration was the best concerning the overall word recognition performance. However, this does not limit the general ability of the HMM-based enhancement method for integrating other adequate types of knowledge in different applications.

We have used OGI Numbers'95 database [78, 79], and a reduced vocabulary version of the DARPA Conversational Telephone Speech-to-text (CTS) task (1000 words) [80] for the experiments. CTS is the main task in the experiments of this chapter. For the OGI Numbers'95 database, the training set contains 3233 utterances spoken by different speakers (approximately 1.5 hours) and the validation set consists of 357 utterances (used during MLP training). The test set contains 1206 utterances. The vocabulary consists of 31 words (including silence) with a single pronunciation for each word. There are 27 context-independent phones including silence. The acoustic vector x_t is the PLP cepstral coefficients [22] extracted from the speech signal using a window of 25 ms with a shift of 12.5 ms, followed by cepstral mean subtraction. At each time frame t, 13 PLP cepstral coefficients, their first-order and second-order derivatives were extracted, resulting in 39 dimensional acoustic vector. For the estimation of regular MLP phone posteriors, we trained an MLP with 351 input nodes (9 frames of acoustic features), 1200 hidden units and 27 output units corresponding to the 27 context-independent phones. In all the experimental setup in this chapter, the structure of MLP is obtained using cross-validation. After training, the phone posteriors for the training set and test set were estimated and scaled by their respective priors (estimated from the training segmentation) to obtain scaled-likelihoods.

For the conversational telephone speech (CTS) database, training set contains 16 hours of male CTS speech randomly selected from the Fisher Corpus and the Switchboard Corpus. The tuning/test set was a subset selected from the the NIST 2003 evaluation set. Only those utterances that covered the top most frequent 1000 words with lower than 10% out-of-vocabulary rate were selected, resulting in 2.5 hours of data which was further divided into a 1.2 hour tuning set and a 1.3 hour test set. The tuning and test sets contained similar ratio of the number of utterances from Fisher corpus to the number of utterances from the Switchboard corpus. There are 46 phones in this task. The acoustic features are 13 PLP coefficients concatenated with their first two derivatives. It was computed with vocal tract normalization (VTLN) [81], and mean and variance normalization. For the estimation of regular posteriors, an MLP was trained with 14.6 hours of speech with the remaining 1.4 hours of speech used as a cross-validation set to prevent over-training. The input layer

of the MLP had 351 nodes containing 9 frames of PLP features, together with their first and second order derivatives. The hidden layer had 1300 nodes and the output layer had 46 outputs. The structure of the MLP is obtained by cross validation. After training, the phone posteriors for the whole database were estimated.

For both databases, the MLP posteriors were then used to estimate the enhanced phone posteriors as explained in Section 3.1. The prior knowledge used to obtain enhanced posteriors is the phonetic duration knowledge as explained before.

We first start with the comparison of the enhanced and MLP posteriors at the frame level. Table 4.1 is showing the fame recognition results (non-italic numbers) for the enhanced and regular MLP posteriors (for the two databases). All the error rates are expressed in percentage. For both databases, the enhanced posteriors show lower frame error rates than the MLP posteriors. We have also performed the statistical significance test [82] in order to verify the reliability of improvements obtained by the enhanced posteriors as compared to the regular posteriors. This test shows if the improvements are due to a genuine advantage of one system over the other, or just an effect of chance. The test is based on a bootstrap method for assigning measures of accuracy to statistical estimates, and it gives a bootstrap estimate of the probability of error reduction (improvement). The results of the test are shown inside parentheses in Table 4.1. The probability of improvement has been expressed in percentage as the confidence on the reliability of improvements. The test indicates a high confidence (100%) that the improvements in the error rates reflect a real superiority of the enhanced posteriors.

In addition, we study the entropy for each type of posteriors. The entropy can provide a measure of consistency/confusion in the posteriors. The entropy of phone posteriors is

measured at each frame, and averaged over the whole database:

$$E_t = -\sum_i p(q_t^i|x_{1:T}, M) \log_2 p(q_t^i|x_{1:T}, M) \tag{4.2}$$

$$AvE = \frac{\sum_{t=1}^{\mathcal{T}} E_t}{\mathcal{T}} \tag{4.3}$$

where E_t is the entropy of posteriors at frame t, and $\mathcal{T}$ is the total number of frames in the database. The average entropy values AvE for the enhanced and MLP posteriors are shown in Table 4.2. Enhanced posteriors have lower entropy compared to the regular MLP posteriors, indicating lower variability. Given that the enhanced posteriors have lower FER (Table 4.1) and lower entropy (Table 4.2) tends to show that they can be potentially more efficient features for further training and inference in Tandem system. Lower variability of enhanced posteriors provides the possibility of more efficient training and feature space modeling in the HMM/GMM back-end.

Database	MLP posteriors	Enhanced posteriors
CTS	35.2	**33.3** *(100.0)*
Numbers	17.6	**16.2** *(100.0)*

Table 4.1: Frame error rates (FER) on Numbers'95 and CTS tasks, for regular MLP posteriors and HMM-based enhanced phone posteriors. Enhanced posteriors have lower FER than the regular MLP posteriors. Frame error rates are obtained on cross-validation partition of the databases. The numbers in parentheses are statistical significance of improvements.

Database	MLP posteriors	Enhanced posteriors
CTS	1.64	**0.33**
Numbers	0.67	**0.18**

Table 4.2: Average entropy of enhanced and regular MLP posteriors. The measures are obtained by computing the entropy of posteriors at each frame, and averaging over the whole database. Enhanced posteriors have lower average entropy indicating higher consistency than the regular posteriors.

After the frame level studies, we investigate the performance of enhanced posteriors for

word recognition. As discussed before, for word recognition studies in Tandem configuration, the enhanced phone posteriors at each frame t are combined with the original MLP posteriors[1]. Two combination rules which are summation (average) and concatenation have been tried. The resulting combined evidences are processed by Log and KLT transforms, as done for normal Tandem feature extraction. We have also extracted the regular baseline Tandem features by performing Log and KLT transforms on the regular MLP posteriors. For comparison purpose, we also report the standard acoustic feature baseline results obtained by using the traditional PLP features (already used for MLP training) in the HMM/GMM back-end.

For each type of features (PLPs, regular Tandem and combined evidence), we trained a HMM/GMM system using HTK toolkit [68]. In case of Numbers database, 80 context-dependent phone models with 12 mixtures per state, and 3 states per phone is used. In case of CTS database, models were trained through 40 iterations: 5 iterations for the context-independent models, 5 iterations for the context-dependent models, 5 iterations for the clustered context-dependent models, and then 5 iteration each for incrementing mixtures from 1 to 32 (2, 4, 8, 16, 32). During the recognition, a bi-gram language model is used.

Table 4.3 is showing the results in terms of word error rate (WER) percentage (for the two databases), using PLPs, regular MLP posteriors, and combined evidence (MLP and enhanced posteriors). The first column shows the standard baseline PLP acoustic feature results. The second column shows the baseline Tandem where the regular MLP posteriors are used as features. The third and forth columns show the performance of the combination of the enhanced and regular posteriors (using different combination rules). The statistical significance of improvements between the baseline Tandem and the system using combined evidence is shown in the parentheses. The combined evidence is performing consistently better than the baseline Tandem and acoustic PLP features. In the CTS case, the best combination rule is concatenation, resulting in 3% relative improvement with 100% confidence. The high confidence indicates the true superiority of the new system using

[1]In practice, using HMM-based enhanced posteriors alone in the Tandem configuration did not improve word recognition performance.

combination (concatenation) of enhanced and MLP posteriors. Using enhanced posteriors (encoding prior and contextual knowledge) in combination with MLP posteriors has helped to provide better evidences for Tandem. Smaller improvement in case of using addition rule is due to imperfect combination strategy. [2] The same statistical significance test on the results for Numbers database gives probabilities of improvement ranging between 81% to 92.4% indicating moderately high confidence on the improvements.

Database	PLP	MLP posteriors	**MLP + Enh**	**MLP & Enh**
CTS	44.4	44.2	**43.8** *(76.1)*	**41.3** *(100.0)*
Numbers	7.3	4.7	**4.3** *(81.1)*	**4.3** *(92.4)*

Table 4.3: Word error rates (WER) on Numbers and CTS tasks, for MLP posteriors, and MLP posteriors combined with the enhanced posteriors, using addition (MLP+Enh) and concatenation (MLP & Enh) as combination rules. The combined evidences perform better than regular MLP posteriors in Tandem configuration.

The overall complexity of Tandem system is composed of the complexity of the MLP used for the estimation of phone posteriors plus the complexity of the HMM/GMM back-end. The parameters of MLP consist of weights and bias for the nodes. The parameters in HMM/GMM back-end consist of mean and covariance matrices for the Gaussians modeling the state emission probabilities, and also transition probabilities between the states. The total number of parameters in Tandem system rises to approximately 190 millions in our case. In this work, the HMM used for integrating prior knowledge has relatively simple topology, imposing 3 frames minimum phone duration. The HMM topology is composed of uniformly connected phone models in which each phone is modeled with 3 states, summing up to 138 (46x3) states and only 2300 parameters overally. In such a simple topology, forward-backward recursions are not computationally expensive. Taking into account the huge number of parameters (approximately 190 millions) and computationally expensive large vocabulary decoding in the HMM/GMM back-end, the additional complexity imposed by the posterior enhancement process is practically negligible. In practice, the enhance-

[2] The analysis of recognition results has shown that the addition rule introduces inconsistency due to partial mismatch between regular and HMM-based enhanced posteriors.

ment process increases the overall Tandem execution duration only by 0.35%.

4.2 MLP-Based Enhanced Posteriors

The enhanced posteriors obtained by a secondary MLP can be also used as features in Tandem configuration. In this case, unlike HMM-based enhanced posteriors, the integrated knowledge is learned from the data. Therefore, there is less risk of being biased by partially wrong prior assumptions. This allows using the enhanced posteriors as features directly (without the need for combination with the regular posteriors). In this way, the configuration for using the MLP-based enhanced posteriors would be similar to the normal Tandem configuration. The only difference is that the regular phone posteriors are replaced with the enhanced phone posteriors. We compare the performance of regular and enhanced posteriors as features in the Tandem configuration. The databases, specifications of spectral features extraction, and regular MLP posterior estimation is the same as the case of HMM-based posterior experiments (see Section 4.1).

In order to enhance phone posterior estimates for the Numbers database, a second MLP for post-processing 19 frames of regular posteriors is used (as explained in Section 3.2). It has 513 (19x27) input nodes, 1000 hidden nodes and 27 output nodes. For enhancing phone posteriors in the CTS database, a second MLP with 690 (15x46) input nodes, 2000 hidden nodes and 46 output nodes is used to post-process 15 frames of regular posteriors[3]. The size of the temporal posterior context, and the structure of the second MLP is obtained by cross validation for all the experiments. The size of the temporal context is close to the reported 200ms duration [5, 6] for phone temporal information.

As before, we start with frame level performance study of enhanced posteriors. Table 4.4 is showing frame error rates percentage of the regular and enhanced posteriors, for Numbers and CTS databases (cross validation portion). Again, lower error rates can be observed

[3]In practice, a temporal context of 9-15 frames resulted in very similar error rates.

for the enhanced posteriors in both databases. Results of the significance test (100% confidence) show high reliability of the improvements.

The same as Section 4.1, we do entropy studies on the MLP-based enhanced posteriors. Table 4.5 shows the average entropies for the enhanced and regular posteriors. Enhanced posteriors have less entropy than the regular posteriors, indicating lower variability. Given that the enhanced posteriors have lower FER (Table 4.4) and lower entropy (Table 4.5), they can be potentially more efficient features for further training and inference in Tandem system.

Database	Regular posteriors	Enhanced posteriors
CTS	35.2	**31.5** *(100.0)*
Numbers	17.6	**15.4** *(100.0)*

Table 4.4: Frame error rates (FER) on Numbers'95 and CTS tasks, for regular (first MLP) and enhanced (second MLP) phone posteriors. Enhanced posteriors have lower FER than the regular posteriors. Frame error rates are obtained on cross-validation partition of the databases.

Database	Regular posteriors	Enhanced posteriors
CTS	1.64	**1.29**
Numbers	0.67	**0.40**

Table 4.5: Average entropy of enhanced (second MLP) and regular (first MLP) phone posteriors for different databases. The measures are obtained by computing the entropy of posteriors at each frame, and taking average over the whole database. Enhanced posteriors have lower entropy indicating higher consistency than the regular posteriors.

In the word recognition studies, we compare the performance of regular and enhanced posteriors as features in the Tandem configuration. Unlike the case of HMM-based posteriors, MLP-based enhanced posteriors can be used directly as features, without being necessarily combined with regular posteriors. Details of implementation for the HMM/GMM back-end is the same as Section 4.1. For comparison purpose, we also report baseline PLP acoustic feature performance. Table 4.6 is showing the word recognition performances for PLPs,

regular posteriors (baseline Tandem) and enhanced posteriors. It can be observed that the enhanced posteriors are consistently performing better than the regular posteriors and also PLP features. As before, the numbers inside the parentheses are showing the statistical significance of the improvements obtained by enhanced posteriors as compared to the regular posteriors. The probability of improvement is high specially for the CTS database, indicating high reliability of the improvements obtained by the system using enhanced posteriors.

Database	PLP	Regular posteriors	Enhanced posteriors
CTS	44.4	44.2	**42.5** *(99.6)*
Numbers	7.3	4.7	**4.3** *(80.5)*

Table 4.6: Word error rates (WER) on Numbers'95 and CTS tasks, for regular and enhanced phone posteriors. Enhanced posteriors are obtained by post-processing regular posteriors using a secondary MLP. The phone posteriors are used in Tandem configuration for the recognition. Enhanced phone posteriors perform consistently better than the regular posteriors for the two databases.

In addition to the use of MLP-based enhanced posteriors as a replacement for the regular MLP posteriors, we have investigated their usage as complementary features to the regular MLP posteriors (as done for HMM-based enhanced posteriors). The configuration for using the combined evidences is the same as shown in Figure 4.1, except that the HMM-based enhanced posteriors are replaced with the MLP-based enhanced posteriors. The same addition and concatenation rules have been tried. Table 4.7 is showing the word recognition results when the MLP-based enhanced posteriors are used as complementary features. As illustrated in Table 4.7, usage of the MLP-based enhanced posteriors as complementary features improves the performance even more than using them instead of regular posteriors. Therefore, they perform best when they are used in combination with the regular MLP posteriors. The statistical significance of the improvements are also high specially for the CTS database.

MLP-based enhancement approach involves using a second MLP in the Tandem configuration. Thanks to the computational efficiency due to the regular and parallel structures of

Database	PLP	MLP posteriors	**MLP + Enh**	**MLP & Enh**
CTS	44.4	44.2	**41.2** *(99.9)*	**42.3** *(99.0)*
Numbers	7.3	4.7	**4.2** *(80.2)*	**4.2** *(82.4)*

Table 4.7: Word error rates (WER) on Numbers and CTS tasks, for MLP posteriors, and MLP posteriors combined with enhanced posteriors using addition (MLP+Enh) and concatenation (MLP & Enh) as combination rules. Enhanced posteriors are obtained at the output of the second MLP. Combined evidences perform better than regular MLP posteriors.

the MLPs, and more importantly taking into account the huge computational load needed for large vocabulary decoding in the Tandem configuration, the additional computational load imposed is practically negligible. The experiments have shown that the overall execution duration increases only by 0.19% when a second MLP is used.

We also further studied the strategies and possibilities of optimizing and using a simpler structure for the second MLP. This will provide the possibility of processing longer temporal context. Phone posteriors have simpler and possibly more linearly separable patterns, as compared to the acoustic features. Therefore, it is potentially possible to use a relatively simpler MLP for post-processing the phone posteriors. In our study, initially we tried to reduce the complexity of the second MLP in terms of the number of hidden nodes. The optimum complexity is obtained empirically. Reducing the complexity below this optimum slightly degrades the performance of enhanced posteriors, however they still perform better than the regular posteriors. The degradation in the performance of enhanced posteriors is small even for large decrease in the complexity of the second MLP. In addition, we have studied using a single layer Perceptron (SLP) as the second ANN. Although the obtained enhanced posteriors are performing better than the regular posteriors, their performance is slightly lower than the case of using MLP as the second ANN. It implies that there are still nonlinearly separable patterns at the output of the first MLP (regular posteriors) which can not be learned by the SLP.

Building upon the same idea of ANN hierarchy, a third MLP has also been tried in order

to post-process the output of the second MLP. Using a third MLP, the frame error rate and entropy results are improved, but no considerable improvement in word recognition is observed.

4.3 Summary and Conclusions

In this chapter, we have studied the use of the enhanced posteriors as features in the configurations similar to Tandem system. We have shown that the enhanced posteriors can be used alone or in combination with the regular posteriors as features in frame synchronous posterior based ASR. The HMM-based enhanced posteriors should be always combined with the regular posteriors in order to improve the performance of the Tandem system. On the other hand, the MLP-based enhanced posteriors can be used alone as replacement to the regular posteriors in Tandem system. We have shown that using enhanced posteriors can consistently improve the word recognition performance in the Tandem system. In addition, frame level studies of the enhanced posteriors also show higher frame recognition as compared to the regular posteriors. We also studied the entropy of the enhanced and regular posteriors. The enhanced posteriors have lower entropy indicating that they are less noisy than the regular posteriors.

Chapter 5

Enhanced Posteriors As Local Scores

In this chapter, we study the use of the enhanced posteriors as local scores (measures) in posterior based ASR. In posterior based ASR framework, local posteriors are used as local scores for decoding, as well as for estimating the confidence level of the recognizer output. The enhanced posteriors can be used instead of the regular posteriors as local scores in decoding and confidence measurement. In the following, we study the two cases, namely enhanced phone posteriors for decoding, and enhanced posteriors for confidence measurement.

5.1 Enhanced Posteriors for Decoding

As described in Section 2.2.2, phone posteriors can be used as local scores for decoding in ASR (e.g. hybrid HMM/ANN system). In this case, phone posteriors are used as state emission probabilities in the HMM configuration. In the same way, the enhanced posteriors can be also used for estimating HMM state emission probabilities. In this section, we

investigate the use of the enhanced posteriors as scores for decoding, and we compare them with the regular MLP posteriors. Since HMM-based and MLP-based enhanced posteriors have different properties, we study them separately.

In Section 5.1.1, we describe the use of HMM-based enhanced posteriors instead of the regular MLP posteriors for decoding. We show that the resulting system is more robust with respect to the changes in ad-hoc tuning parameters such as phone and word deletion penalties. Section 5.1.2 describes the use of the MLP-based enhanced posteriors as local scores for decoding. We show that the MLP-based enhanced posteriors perform consistently better in terms of frame, phone and word recognition.

5.1.1 HMM-Based Enhanced Posteriors

HMM-based enhanced posteriors can be used as local scores for decoding, in the same way as regular posteriors are used in HMM/ANN configuration. Unlike the case of using HMM-based enhanced posteriors as features (described in Chapter 4), there are few issues regarding the use of these posteriors as local scores for decoding. The main issue is the fact that the knowledge which is integrated in the enhancement process is the same as the knowledge which is taken into account in the topological constraints of the decoder. For instance, the same duration knowledge as integrated in the enhancement process is taken into account in the hybrid decoder configuration. This means that we should not expect performance improvement when the HMM-based enhanced posteriors are used for decoding, since no additional knowledge is integrated in the enhancement process. The experiments also confirm that the performance of the enhanced and regular posteriors for decoding are the same. However, there is a side advantage in using HMM-based enhanced posteriors for decoding.

The advantage is revealed when we compare the sensitivity to ad-hoc tuning factors (e.g. phone deletion penalty) for the decoder using the enhanced posteriors, and the decoder using regular posteriors. Phone deletion penalty is a tuning factor and an engineering trick

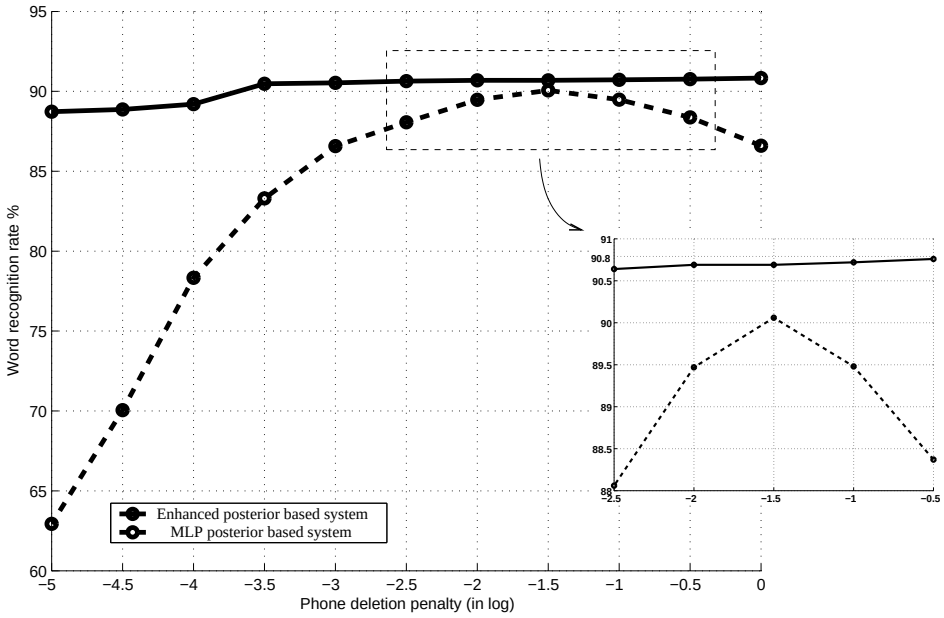


Figure 5.1: Comparing the sensitivity to tuning phone deletion penalty, for the decoder using enhanced posteriors and the one using MLP posteriors. Phone deletion penalty is varied for the two decoders and the performances are observed (on OGI Numbers'95 database). The inside diagram is a zoom of performance curves for small values of phone deletion penalty (fine tuning). The decoder using enhanced posteriors is much less sensitive to tuning ad-hoc parameters than the one using regular MLP posteriors.

which is used for numerical compensation of scores for different paths during decoding [12, 83, 84]. It can noticeably affect the recognition performance of standard HMM/ANN and HMM/GMM systems[1]. We have setup some experiments to investigate this issue.

We have used OGI Numbers'95 database [78, 79] for the experiments. Specifications of the database, spectral features and regular MLP posteriors estimation is the same as mentioned in Section 4.1. We have used a fully constrained model (as explained in Section 3.1) to get estimates of enhanced posteriors. This means that we integrate full lexical and phonetic knowledge in the posterior estimation. The obtained enhanced posteriors are then

[1]Usually this factor is tuned using a development set to get maximum performance, which does not guarantee the same improvement on the test set, specially if the conditions (e.g. noise level, task, etc.) change. Sometimes it is even tuned over the test set which is an incorrect practice as it shows optimistically biased results! In any case, there is no strong theoretical explanation for tuning, it makes the system less robust against changes and it is time consuming.

used as local scores for decoding. We have used NOWAY [85] as the hybrid decoder. For comparison, regular phone posteriors are also used in the same decoder. In order to compare the sensitivity of the systems (one using regular posteriors, and the other one using enhanced posteriors), we vary the phone deletion penalty value in the decoder and observe the change of performance for the two systems. Figure 5.1 shows the results. Comparing the two curves, we can conclude that the decoder using enhanced posteriors is much less sensitive to tuning than the one using regular posteriors (standard hybrid HMM/MLP system). HMM-based enhanced posteriors tend to have very close to binary values (similar to a decision) because they are estimated by integrating some extra knowledge, while the MLP posteriors can change more smoothly between 0 and 1. Therefore, the accumulated scores obtained by enhanced posteriors during decoding tend to be discrete, while it is continuous for the case of regular MLP posteriors. The tuning operation (which slightly changes the scores) affects the decision made based on continuous scores more than the one made based on discrete scores. This means that the decoder using enhanced posteriors is much less sensitive to tuning ad-hoc parameters.

5.1.2 MLP-Based Enhanced Posteriors

The MLP-based enhanced posteriors can be also used for decoding in the same way as regular posteriors. In this case, they are used as local scores instead of the regular posteriors in the hybrid HMM/ANN configuration. We compare the performance of regular and enhanced posteriors for decoding. The comparison is done for the OGI Numbers [78, 79] and CTS [80] databases. The specifications of databases, regular MLP posteriors, and enhanced posterior estimation are the same as mentioned in Section 4.2. We have used NOWAY [85] as the hybrid decoder for Numbers database, and JUICER [86] for CTS database. Table 5.1 is showing the word recognition performances for regular and enhanced posteriors[2]. It can be observed that the enhanced posteriors are performing noticeably better than the regu-

[2]In this case, the PLP baseline can not be reported because the PLP features can not be used as local scores for decoding.

lar posteriors for the two databases[3]. As before, the numbers in the parentheses show the statistical significance of improvements, indicating high superiority of the system using enhanced posteriors. The probabilities of improvement is over 99%.

In hybrid HMM/ANN system, the computational load is mainly due to the decoder searching for the best word sequence hypothesis in HMM. The additional computational load imposed by the MLP-based enhancement process is very small compared to the computation load needed in the decoder, and it can be practically ignored. The experiments have shown that the computation duration for the hybrid system using enhanced posteriors increase only by 1.92%.

Database	Regular posteriors	Enhanced posteriors
CTS	53.6	**49.2** *(100.0)*
Numbers	9.9	**8.8** *(99.1)*

Table 5.1: Word error rates (WER) on Numbers'95 and CTS tasks, for regular and enhanced phone posteriors. The phone posteriors are used in hybrid HMM/ANN configuration for decoding. Enhanced posteriors perform better than the regular posteriors.

Error rates	Regular posteriors	Enhanced posteriors
FER	29.9	**27.4** *(100.0)*
PER	31.2	**28.5** *(99.3)*

Table 5.2: Frame error rates (FER) and phone error rates (PER) for regular and enhanced phone posteriors, on TIMIT database. Lower FER and PER can be observed for enhanced posteriors as compared to the regular posteriors.

We have also performed phone recognition experiments to compare the enhanced and regular posteriors for phone recognition in a hybrid decoder. For the experiments, TIMIT

[3]Comparing the baseline hybrid system (Table 5.1, 2nd column) with baseline Tandem system (Table 4.3, 3rd column), we can observe that the baseline hybrid system is lower than baseline Tandem system. However, this should not be a surprise considering the fact that Tandem system benefits from sophisticated techniques for improving the performance (e.g. context-dependent phone modeling, embedded training). The advantage of hybrid system is mainly in simplicity and better generalization to new tasks.

database [87] is used. The training data set consists of 3000 utterances from 375 speakers, cross validation data set consists of 696 utterances from 87 speakers and the test data set consists of 1344 utterances from 168 speakers. There are 39 context-independent phones. The acoustic features are PLP, delta and double delta features. For estimating regular posteriors, we have used an MLP with 351 input nodes (9 frames of PLPs), 1000 hidden nodes and 39 (corresponding to the number of phones) output nodes.

In order to estimate enhanced posteriors, 19 frames temporal context of the regular posteriors is post-processed by a secondary MLP (as explained in Section 3.2). This MLP has 741 (39x19) input nodes, 1000 hidden nodes and 39 output nodes (corresponding to the number of phones). For the phone recognition, we have used NOWAY [85] as hybrid decoder. In this decoder, each phone is modeled with 3 states, and a bi-gram phone level language model is used. Frame and phone recognition results are shown in Table 5.2. The enhanced posteriors perform noticeably better than the regular posteriors for frame and phone recognition.

5.2 Enhanced Posteriors in Confidence Measurement

Posterior probabilities have also been used for the purpose of confidence measurement. A confidence measure is a score that is applied to the speech recognition output. It gives an indication of how confident we are that the unit to which it has been applied (e.g. a phrase, word, phone) is correct. A word may be hypothesized with low confidence when the word model is matched against unclear acoustics caused by disfluencies or noise, or when an out-of-vocabulary (OOV) word is encountered. Confidence measures can be used to reject those hypotheses which are likely to be erroneous (i.e., have a low confidence) in a hypothesis test. Over the last two decades, considerable research has been devoted to the development of confidence scores associated with the outputs of ASR systems [8, 64, 65, 88, 89]. A

reliable measure for the confidence of a speech recognizer output is useful in many applications. These measures have been used mostly to help spot keywords in spontaneous or read texts, and to provide a basis for the rejection of OOV words. Many other ASR applications could also benefit from knowing the level of confidence for a recognized word. For example, text-dependent speaker recognition systems could put more emphasis on words recognized with higher confidence; unsupervised adaptation algorithms could adapt the acoustic model only when the confidence level is high, human-made transcriptions could be verified by ASR systems outputting their confidence in the transcribed word sequence, etc.

In this thesis, our preliminary concern is posterior based ASR systems. Several confidence measures have been proposed for posterior based ASR, particularly hybrid HMM/ANN systems [50, 51, 66, 90, 91]. As discussed before, ANNs are capable of providing good estimates of posterior probability $p(q_t^i|x_t)$ of an HMM state/phone q^i at time t given an acoustic feature vector x_t. Hybrid HMM/ANN systems thus seem particularly well suited to generate confidence measures since, by definition posterior probabilities measure the probability of being correct. The posterior based confidence measures (PCMs) are existing at the word and at the phone levels. They are estimated based on accumulating local phone posteriors (MLP outputs) within a phone or word hypothesis boundary, followed by normalization with respect to the length of the hypothesis. This normalization counteracts the underestimate of the acoustic probabilities caused by the observation independence assumption. In this section, we study the use of the enhanced posteriors as local phone posteriors, replacing the regular MLP posteriors in this confidence measurement methodologies. Since the enhanced posteriors are expected to be more informative than the regular MLP posteriors, they can potentially lead to better (more reliable) confidence measures.

5.2.1 Phone Confidence Measures

At the phone hypothesis level, the normalized posterior based confidence measure, denoted $NPCM$ is defined as the logarithm of a global phone posterior probability computed as the product of the local phone posteriors along the optimal state sequence, and normalized by the duration of the phone hypothesis [50, 51]. For a phone hypothesis q^i, starting at frame b and ending at frame e, the confidence measure is defined as:

$$NPCM(i) = \frac{1}{e-b+1} \sum_{t=b}^{e} \log p(q_t^i|x_t) \tag{5.1}$$

The normalization is necessary due to different phone durations, as otherwise short phones would be favored.

5.2.2 Word Confidence Measures

The word confidence measures are defined in a similar manner. For a word hypothesis w, composed of a sequence of L phone hypotheses $(q^1, ..., q^l, ..., q^L)$, the $frame{-}basedNPCM(w)$ is defined as:

$$frame-basedNPCM(w) = \frac{1}{\sum_{l=1}^{L}(e_l - b_l + 1)} \sum_{l=1}^{L} \sum_{t=b_l}^{e_l} \log p(q_t^l|x_t) \tag{5.2}$$

where b_l and e_l are respectively the beginning and end frames of phone hypothesis q^l in the considered word. A second word confidence measure can be defined by doing a secondary normalization with respect to the number of phones in the hypothesized word. This measure is called $phone-basedNPCM(w)$, and defined as follows:

$$phone-basedNPCM(w) = \frac{1}{L} \sum_{l=1}^{L} \Big(\frac{1}{e_l - b_l + 1} \sum_{t=b_l}^{e_l} \log p(q_t^l|x_t) \Big) \tag{5.3}$$

There are also other alternatives to these confidence measures such as mean posterior

confidence measures (MPCMs). MPCMs at phone and word levels are computed as NPCMs in (5.1), (5.2) and (5.3), except that we compute the average of local posteriors before taking the logarithm.

For all these measures, phone and word hypothesis boundaries (b_l and e_l) and optimal state sequence are obtained using Viterbi decoding by back tracking the decoded state sequence.

5.2.3 Enhanced Posteriors: More Informative Local Evidences

Confidences measures defined in the previous section are global scores obtained by accumulating local evidences (phone posteriors). Having better (more informative) local evidences can potentially lead to better confidence measures. Our studies in Chapter 4 and Section 5.1, have shown that the enhanced posteriors lead to better performance at the frame, phone and word levels, indicating that they are better (more precise) local estimators than the regular MLP posteriors. Therefore, they can provide better local evidences for phones in the confidence measurement process. This means that using the enhanced posteriors instead of the regular MLP posteriors can potentially improve the confidence measures previously defined. In order to evaluate this idea, the local posterior estimates (MLP outputs) in the above definitions are simply replaced with the enhanced posterior estimates. In the following, the performance of the two types of posteriors (regular and enhanced) for confidence estimation is compared.

5.2.4 Experiments and Results

The confidence measures are evaluated in terms of their ability to predict whether a particular phone or word hypothesis is correct or incorrect. A hypothesis is rejected if its confidence score falls below a threshold. Two types of error can occur: Type I error corresponding to the rejection of a correct hypothesis, and type II error corresponding to the acceptation of an incorrect hypothesis. The performance of confidence measures is then

evaluated in terms of type I and type II errors, and the classification error rate (CER) is defined as:

$$CER = \frac{\text{Type I errors + Type II errors}}{\text{Total number of hypotheses in the test set}} \tag{5.4}$$

CER has been conventionally used in related posterior based confidence measure studies to evaluate the performance. For the experiments, we have used a partition of Wall Street Journal (WSJ) Database [92]. There are 45 phones and 5k words in this database. The training set size is about 70 hours and the test size is about 1.1 hours. The test set is recognized using Viterbi decoding through the best trained ASR model available for the task. The ASR model is a HMM/GMM using context-dependent models for phone acoustic modeling and a bi-gram language model for decoding. The decoding generates word and phone level hypotheses and segmentations. For the evaluation, the decoding results and reference word and phone sequences were aligned so that each hypothesis could be marked as correct or incorrect, allowing the evaluation of the performance of each of the confidence measures as hypothesis test statistics. In order to make the performance differences clear between the different confidence measures, the number of true and false word hypotheses in the test set were equalised for each condition. This was done by counting the number of false hypotheses for a condition and randomly selecting the same number from the set of true hypotheses for that condition[4]. Equalising the number of true and false hypotheses had the effect of artificially raising the recogniser error rate close to 0.5 for each condition.

Confidence levels are then estimated at the phone and word levels for each hypothesis using the described measures. For estimating regular phone posteriors, we have used an MLP with 351 (corresponding to 9 frames of 39 dimension PLP features) input, 2000 hidden, and 45 output nodes (corresponding to the number of phones). The experimental setup for the estimation of enhanced posteriors is described in the following sections. All the NPCM and MPCM confidence measures defined in (5.1-5.3) are estimated using both regular and enhanced posteriors. As discussed in Chapter 3, there are two types of enhanced posteriors,

[4]In practice, since the equalization is done at the utterance level, the number of true and false hypotheses are not exactly equal, but very close.

MLP-based enhanced posteriors and HMM-based enhanced posteriors. The experiments and comparisons are carried out for both types of enhanced posteriors as described in the following.

HMM-Based Enhanced Posteriors

In order to estimate HMM-based enhanced posteriors, phone duration information was integrated in the regular MLP posteriors. This was achieved by modeling phones with 3 states in the HMM module used for prior knowledge integration. The posterior based confidence measures are computed with both regular and HMM-based enhanced posteriors. The confidence measures are then compared with a range of thresholds to decide about acceptance/rejection of hypotheses. Finally, CER values are computed as previously described.

Phone Confidence Measures: Figures 5.2 and 5.3 are showing performance curves for NPCM and MPCM phone level confidence measures obtained using regular and enhanced posteriors. Regular posterior results are plotted in blue and enhanced posterior results are plotted in red. The horizontal axis shows the percentage of hypotheses that were rejected and is a function of the confidence threshold. The vertical axis shows the CER percentage. The area under the error curves corresponding to the enhanced posteriors is smaller (i.e. better trade-offs) compared to the ones corresponding to the regular posteriors. This is consistent for both NPCM and MPCM measures.

Word Confidence Measures: The same study is repeated for the NPCM and MPCM word confidence measures defined in (5.2, 5.3). Figures 5.4 and 5.5 are showing the results for different word confidence measures estimated using regular and enhanced posteriors. The results corresponding to regular posteriors are plotted in blue, and results corresponding to enhanced posteriors are plotted in red. Again, it can be observed that the enhanced posteriors are consistently performing better than the regular posteriors for confidence

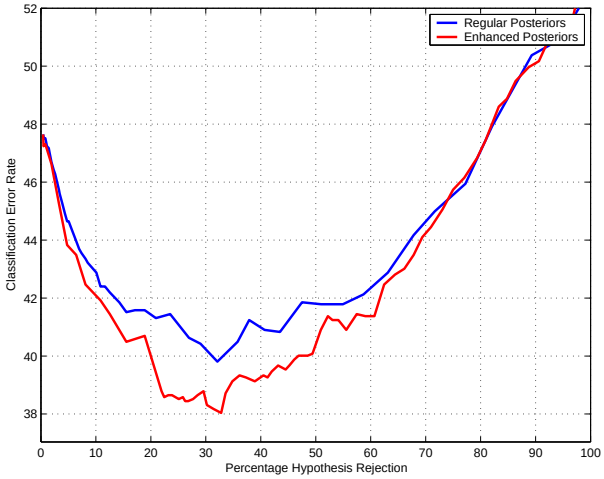


Figure 5.2: CER curves for NPCM phone hypothesis confidence measure. The y axis is showing CER percentage and the x axis is showing phone hypothesis rejection percentage. The blue curve is obtained using regular posteriors and the red curve is obtained using HMM-based enhanced posteriors.

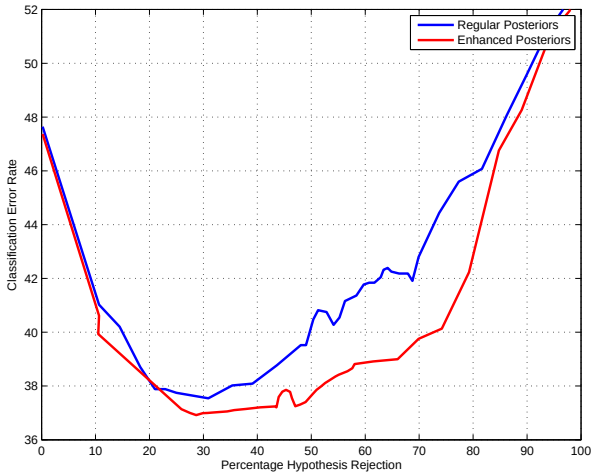


Figure 5.3: CER curves for MPCM phone hypothesis confidence measure. The conditions are the same as Fig. 5.2.

measurement. For all the measures (frame and phone-based NPCM, frame and phone-based MPCM), the area under the error curves corresponding to enhanced posteriors is smaller.

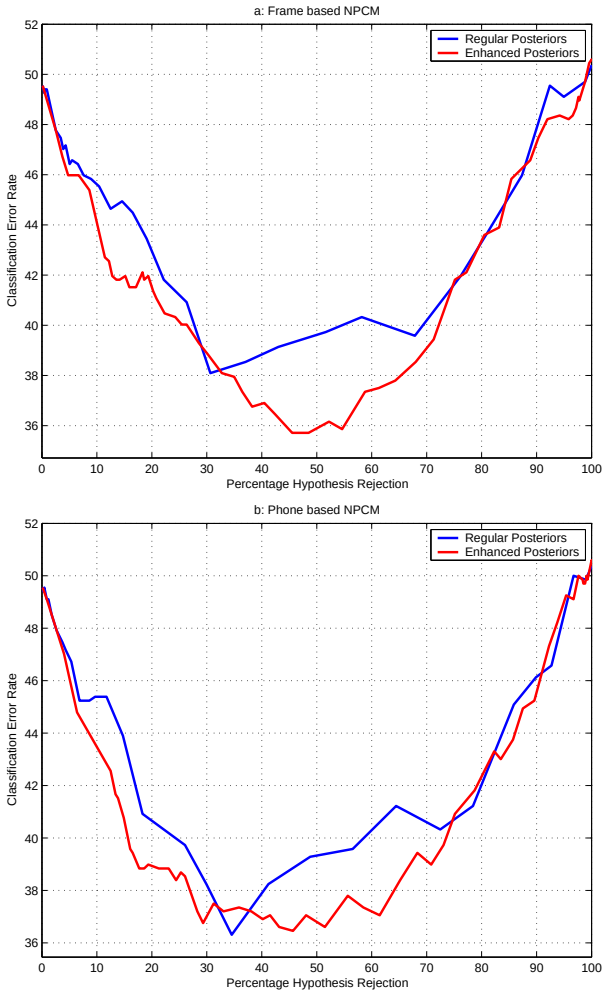


Figure 5.4: CER curves for NPCM word hypothesis confidence measures. (a) The error curves for $frame - basedNPCM$ measures, and (b) the curves for $phone - basedNPCM$ measures. The y axis is CER percentage and the x axis is word hypothesis rejection percentage. The blue curves are obtained using regular posteriors and the red curves are obtained using HMM-based enhanced posteriors.

MLP-Based Enhanced Posteriors

For estimating MLP-based enhanced posteriors, a secondary MLP was used to post-process the regular posteriors as described in Section 3.2. The temporal posterior context is made by concatenating 9 frames of regular posteriors. This posterior context is processed by the second MLP having 405 (9x45) input, 2000 hidden, and 45 output nodes. The size of the

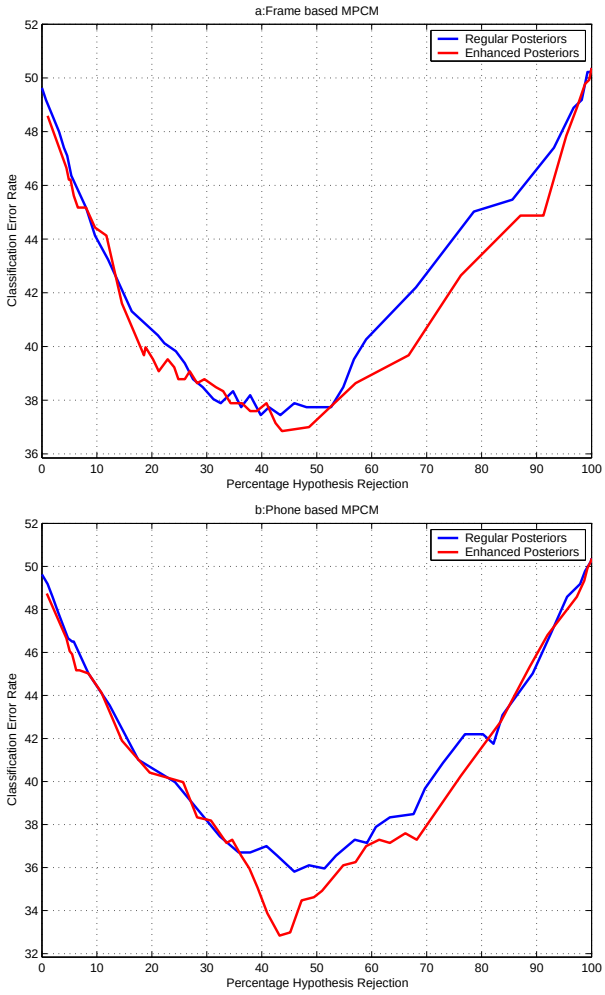


Figure 5.5: CER curves for MPCM word hypothesis confidence measures. (a) The error curves for $frame - basedMPCM$ measure, and (b) the curves for $phone - basedMPCM$ measure. The y axis is CER percentage and the x axis is word hypothesis rejection percentage. The blue curves are obtained using regular posteriors and the red curves are obtained using HMM-based enhanced posteriors.

temporal posterior context and the hidden layer are obtained empirically. The phone and word confidence measures are estimated using both regular MLP posteriors (first MLP output), and MLP-based enhanced posteriors (second MLP output).

Phone Confidence Measures: Figures 5.6 and 5.7 are showing performance curves for phone hypothesis confidence measures (NPCM and MPCM) obtained using regular and

enhanced posteriors. Regular posterior results are plotted in blue and enhanced posterior results are plotted in red. The area under the curves corresponding to the enhanced posteriors (second MLP output) is smaller (i.e. better trade-offs) compared to the ones corresponding to the regular posteriors (first MLP output).

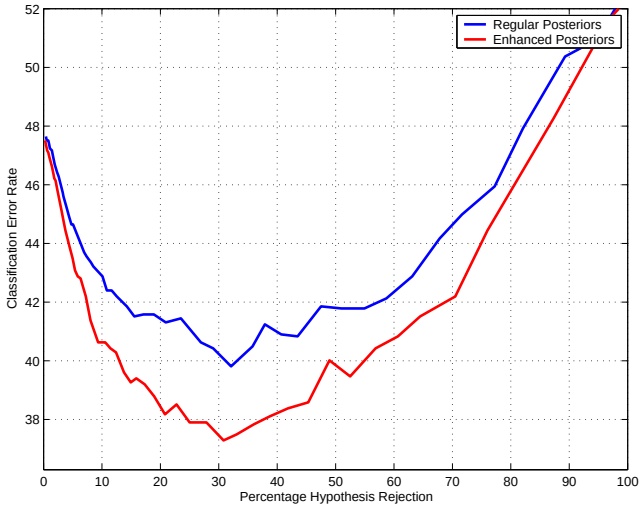


Figure 5.6: CER curves for NPCM phone hypothesis confidence measure. The y axis is showing CER percentage and the x axis is showing phone hypothesis rejection percentage. The blue curve is obtained using regular posteriors and the red curve is obtained using MLP-based enhanced posteriors.

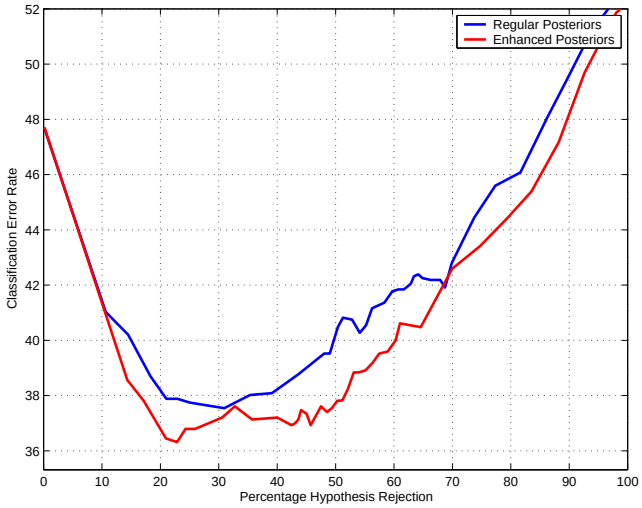


Figure 5.7: CER curves for MPCM phone hypothesis confidence measure. The conditions are the same as Fig. 5.6.

Word Confidence Measures: Figures 5.8 and 5.9 are showing the trade-offs obtained for NPCM and MPCM word confidence measures, using both regular and enhanced posteriors. Results corresponding to the regular posteriors are plotted in blue and results corresponding to enhanced posteriors are plotted in red. Again, the enhanced posteriors are consistently performing better than the regular posteriors for confidence measurement. This is consistent for both NPCM and MPCM measures.

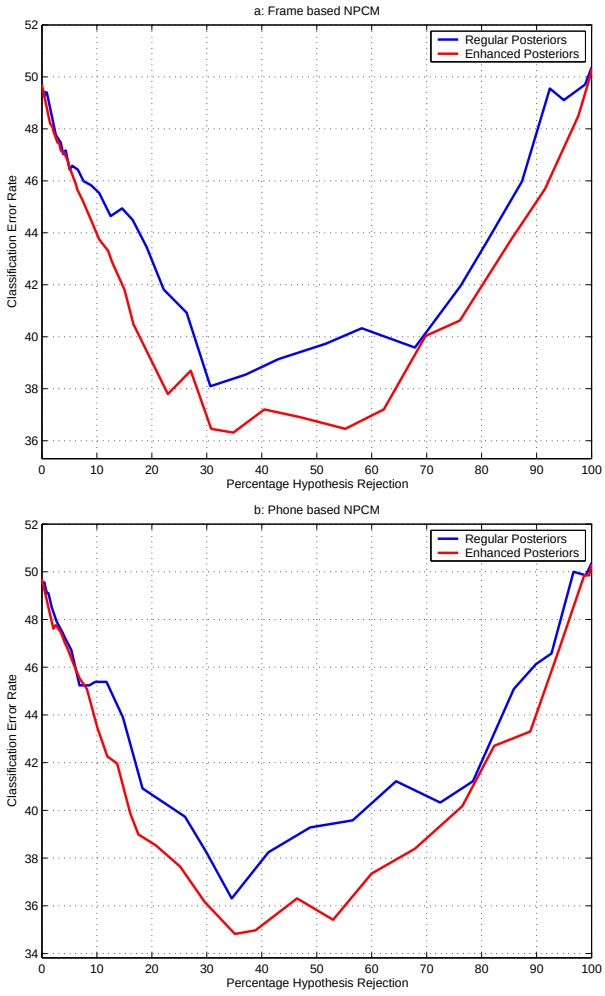


Figure 5.8: CER curves for NPCM word hypothesis confidence measures. (a) The error curves for $frame-basedNPCM$ measure, and (b) the curves for $phone-basedNPCM$ measure. The y axis is CER percentage and the x axis is word hypothesis rejection percentage. The blue curves are obtained using regular posteriors and the red curves are obtained using MLP-based enhanced posteriors.

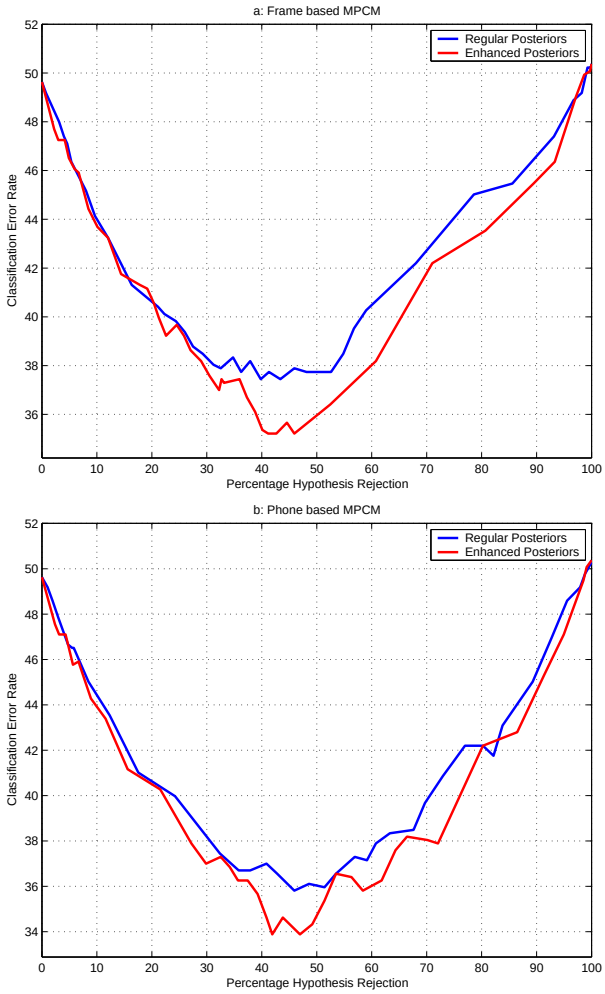


Figure 5.9: CER curves for MPCM word hypothesis confidence measures. (a) The error curves for $frame - basedMPCM$ measure, and (b) the curves for $phone - basedMPCM$ measure. The y axis is CER percentage and the x axis is word hypothesis rejection percentage. The blue curves are obtained using regular posteriors and the red curves are obtained using MLP-based enhanced posteriors.

5.3 Summary and Conclusions

In this chapter, we studied the use of the enhanced posteriors as local scores (measures) in ASR. As the first part of the study, we investigated the use of these posteriors as state emission probabilities for decoding in hybrid HMM/ANN configuration. We showed that MLP-based enhanced posteriors noticeably outperform the regular posteriors for phone

and word recognition over different small and large vocabulary databases. The HMM-based enhanced posteriors perform comparable to the regular posteriors, however the resulting system is much less sensitive to tuning ad-hoc insertion/deletion penalties during decoding.

As the second part of the study, we investigated the use of enhanced posteriors in confidence measurement. We have presented the conventional confidence measures defined for hybrid HMM/ANN ASR. We proposed to use the enhanced phone posteriors instead of the regular posteriors in the confidence level estimation. The experiments showed that using enhanced posteriors, the confidence levels are consistently performing better for predicting whether a hypothesis is correct or incorrect, as compared to regular phone posteriors.

Chapter 6

Multi-stream Enhanced Posterior Estimation

In the previous chapters, we have discussed about increasing attention to the use of local posterior probabilities in ASR. Multi-stream processing [93, 94, 95, 96] is another successful and recently explored filed of research in ASR. Multi-stream systems take the advantage of obtaining information from multiple complementary sources of information to arrive at a decision. Multi-stream processing can result in improving the robustness and performance of a speech recognition system. The redundancy which exists in multi-stream systems makes them more robust against failures. These systems can give reasonable performance in the case of failure of some streams in the system. Moreover, they may result in improved performance when all the classifiers trained on different feature streams work reliably and their outputs are optimally combined.

Considering the success and increased attention to the fields of posterior based ASR and multi-stream processing, in this chapter we study an extension of HMM-based enhanced posterior estimation for the case of multi-stream features. Previously, we have seen how the HMM-based enhanced posteriors can be estimated through a conventional HMM con-

figuration. In this chapter, the extension of this approach to the case of multi-stream HMMs is studied. Building upon the idea of multi-stream HMMs [93, 95], we present investigations for enhancing posterior estimation in ASR, by estimating (phone) posteriors through a multi-stream HMM configuration. We refer to the new posteriors as "multi-stream enhanced posteriors". This provides the possibility of taking into account multiple streams of features for posterior estimation, as well as accommodating context, and prior phonetic and lexical knowledge encoded in the topology of the HMM. The multiple evidences can be provided from different input channels, presenting different complementary aspects of the speech signal. One can even think of using different features from other modalities (e.g. visual features). The estimation of these multi-stream posteriors is based on multi-stream forward-backward HMM recursions developed in this thesis. In these recursions, multiple evidences obtained from multiple streams of features are combined, and used to estimate a single stream of enhanced phone posteriors. This approach provides a new theoretical framework for estimating posteriors taking into account different streams of features, as well as context and prior knowledge. As a practical case, we have used these enhanced posteriors as features for training and inference in a standard HMM/GMM system (a configuration similar to Tandem [2]). The input streams of features are PLP cepstral [22] and MRASTA temporal [71] features, which are known to have complementary information. We have used OGI digits [78, 79] and conversational telephone speech (CTS) [80] databases for the experiments. We show that this method gives consistent performance improvement over baseline PLP-Tandem [2] and MRASTA-Tandem [71] techniques, and also an entropy based combination method [97].

The chapter is organized as follows: Section 6.1 briefly reviews principles of state posterior estimation and forward-backward recursions through a single stream (conventional) HMM. The extension of this method to multi-stream case is studied in Section 6.2. Section 6.3 describes a practical system for enhancing phone posterior estimates by taking into account multiple streams of features, as well as prior knowledge. Section 6.4 presents the experiments and results. Conclusions and future work directions are discussed in Section 6.5.

6.1 Single Stream HMM-based Posterior Estimation

In Section 3.1, the estimation of phone posteriors through a HMM was described. It was shown that phone (state) posteriors can be estimated using HMM forward-backward recursions. In these recursions, MLPs are used to estimate HMM state emission probabilities. Each state of the HMM is associated with one of the MLP outputs (phone posteriors) in the HMM configuration. The MLP outputs are divided by the corresponding phone priors to obtain scaled likelihoods. The scaled likelihoods are then used as emission probabilities in the HMM:

$$\frac{p(x_t|s_t^k)}{p(x_t)} = \frac{p(s_t^k|x_t)}{p(s_t^k)} \tag{6.1}$$

$$\alpha(k,t) = \frac{p(s_t^k|x_t)}{p(s_t^k)} \sum_j p(s_t^k|s_{t-1}^j)\alpha(j,t-1) \tag{6.2}$$

$$\beta(k,t) = \sum_j \frac{p(s_{t+1}^j|x_{t+1})}{p(s_{t+1}^j)} p(s_{t+1}^j|s_t^k)\beta(j,t+1) \tag{6.3}$$

where x_t is acoustic feature at time t, and s_t^k is the event of having state k at time t. The outcome of the forward-backward recursions is the HMM state posterior probability $p(s_t^k|x_{1:T}, M)$. The state posteriors are then turned into enhanced phone posterior estimates $p(q_t^i|x_{1:T}, M)$. In the following section, we study the extension of the state posterior estimation to the multi-stream case, where the state posteriors are estimated for a multi-stream HMM.

6.2 Estimating Posteriors Through a Multi-stream HMM

In multi-stream HMM configuration, the definition of the state posterior is extended to the probability of being in specific state k at specific time t, s_t^k, given the whole observation

sequences for *multiple streams*, and model M encoding specific prior knowledge[1]:

$$\gamma(k,t) \triangleq p(s_t^k|x_{1:T}^1,...,x_{1:T}^n,...,x_{1:T}^N,M) \tag{6.4}$$

where superscript n indicates the stream number, and N is the total number of streams. We refer to the state posterior estimated using multiple streams of features as "multi-stream state posterior". In the remainder of this chapter, we often drop the M keeping in mind that all the recursions are processed through a Markov model M. As we show in this section, multi-stream state posteriors can be estimated using multi-stream forward α and backward β recursions developed in this thesis. The multi-stream α and β recursions can be written based on individual stream α^n and β^n recursions. In this work, we focus on the posterior based systems, therefore all the recursions are written using scaled likelihoods. The same multi-stream recursions but for likelihood based systems have been described in [98].

We start with individual stream forward α^n recursion (superscript n indicates the stream number):

$$\begin{aligned}\alpha^n(k,t) &\triangleq \frac{p(x_{1:t}^n,s_t^k)}{\prod_{\tau=1}^t p(x_\tau^n)} \\ &= \frac{p(s_t^k|x_t^n)}{p(s_t^k)}\sum_j p(s_t^k|s_{t-1}^j)\alpha^n(j,t-1)\end{aligned} \tag{6.5}$$

[1]As in Section 3.1, we start with state posterior estimation, and we show that state posteriors can be integrated into phone posteriors.

and backward β^n recursion:

$$\begin{aligned}\beta^n(k,t) &\triangleq \frac{p(x^n_{t+1:T}|s^k_t)}{\prod_{\tau=t+1}^{T} p(x^n_\tau)} \\ &= \sum_j \frac{p(s^j_{t+1}|x^n_{t+1})}{p(s^j_{t+1})} p(s^j_{t+1}|s^k_t)\beta^n(k,t+1)\end{aligned} \tag{6.6}$$

where $\alpha^n(k,t)$ and $\beta^n(k,t)$ show the forward and backward recursions for stream n.

Using individual stream forward recursions α^n and applying the usual HMM assumptions, we can write multi-stream forward α recursion as follows:

$$\begin{aligned}\alpha(k,t) &\triangleq \frac{p(x^1_{1:t}, x^2_{1:t}, ..., x^N_{1:t}, s^k_t)}{\prod_{\tau=1}^{t} p(x^1_\tau) \prod_{\tau=1}^{t} p(x^2_\tau) ... \prod_{\tau=1}^{t} p(x^N_\tau)} \\ &= \frac{\alpha^1(i,t)}{p(s^k_t)} \frac{\alpha^2(i,t)}{p(s^k_t)} ... \frac{\alpha^N(i,t)}{p(s^k_t)} p(s^k_t) \\ &= \frac{\prod_{n=1}^{N} \alpha^n(i,t)}{p(s^k_t)^{N-1}}\end{aligned} \tag{6.7}$$

See Appendix (Section 6.6.1) for detailed equations. For resolving above equations, we add the following independence assumption:

$$p(x^1_{1:t}, x^2_{1:t}, ..., x^N_{1:t}|s^k_t) = p(x^1_{1:t}|s^k_t)p(x^2_{1:t}|s^k_t)...p(x^N_{1:t}|s^k_t) \tag{6.8}$$

This assumption implies that knowing the state, the past observations in different streams are independent. This allows modeling the observations in different streams independently.

The multi-stream β recursion can also be written using individual stream β^n recursions:

$$\begin{aligned}\beta(i,t) &\triangleq \frac{p(x^1_{t+1:T}, x^2_{t+1:T}, ..., x^N_{t+1:T}|s^k_t)}{\prod_{\tau=t+1}^{T} p(x^1_\tau) \prod_{\tau=t+1}^{T} p(x^2_\tau) ... \prod_{\tau=t+1}^{T} p(x^N_\tau)} \\ &= \beta^1(i,t)\beta^2(i,t)...\beta^N(i,t) \\ &= \prod_{n=1}^{N} \beta^n(i,t) \end{aligned} \tag{6.9}$$

See Appendix (Section 6.6.2) for detailed equations. For resolving the equations, we add the following independence assumption:

$$p(x^1_{t+1:T}, x^2_{t+1:T}, ..., x^N_{t+1:T}|s^k_t) = p(x^1_{t+1:T}|s^k_t)p(x^2_{t+1:T}|s^k_t)...p(x^N_{t+1:T}|s^k_t) \tag{6.10}$$

This assumption implies that knowing the state, the future observations in different streams are independent. This allows modeling the observations in different streams independently. The multi-stream state posterior $\gamma(i,t)$ can then be obtained using multi-stream α and β recursions:

$$\gamma(k,t) \triangleq p(s^k_t|x^1_{1:T}, x^2_{1:T}, ..., x^N_{1:T}) = \frac{\alpha(k,t)\beta(k,t)}{\sum_j \alpha(j,t)\beta(j,t)} \tag{6.11}$$

See Appendix (Section 6.6.3) for detailed proof of the equation. We remind that all multi-stream recursions are processed through a (Markov) model M.

In most of the practical cases in ASR, we are interested in phone posteriors rather than state posteriors. Phone posteriors can be expressed in terms of state posteriors $\gamma(k,t)$ as

follows:

$$\begin{aligned} p(q_t^i|x_{1:T}^1, x_{1:T}^2, ..., x_{1:T}^N) &= \sum_{j=1}^{N_s} p(q_t^i, s_t^j|x_{1:T}^1, x_{1:T}^2, ..., x_{1:T}^N) \\ &= \sum_{j=1}^{N_s} p(q_t^i|s_t^j, x_{1:T}^1, x_{1:T}^2, ..., x_{1:T}^N) p(s_t^j|x_{1:T}^1, x_{1:T}^2, ..., x_{1:T}^N) \\ &= \sum_{j=1}^{N_s} p(q_t^i|s_t^j, x_{1:T}^1, x_{1:T}^2, ..., x_{1:T}^N) \gamma(j,t) \end{aligned} \tag{6.12}$$

where N_s is the total number of HMM states, q_t is a phone at time t, and q_t^i is the event of having phone i at time t. Probability $p(q_t^i|s_t^j, x_{1:T}^1, x_{1:T}^2, ..., x_{1:T}^N)$ represents the probability of being in a given phone i at time t knowing to be in the state j at time t. If there is no parameter sharing between phones, this is deterministic and equal to 1 or 0. Otherwise, this can be estimated from the training data.

In this way, we have ended up with a new phone posterior $p(q_t^i|x_{1:T}^1, x_{1:T}^2, ..., x_{1:T}^N, M)$, which is estimated out of multiple (complementary) streams of features, and additionally integrates the prior knowledge M encoded in the HMM topology, as well as the long context $x_{1:T}$ in different streams. Considering multi-stream feature combination and integration of additional prior knowledge, the new phone posterior is more informative than the regular single stream MLP posteriors.

Figure 6.1 shows a diagram of the multi-stream posterior estimation method. The system has two modules: The first module gets two streams of raw acoustic features (e.g. PLP and MRASTA) extracted from speech signal, and estimates two streams of regular phone posteriors using MLPs. This is called “single stream regular posterior estimation”. PLP and MRASTA features are known to have complementary information. These single streams of posteriors are used (after turning to scaled likelihoods) in the second module, which is a multi-stream posterior based HMM, to estimate multi-stream enhanced phone posteriors. In the next section, we study a practical case for the use of these posteriors as features, in

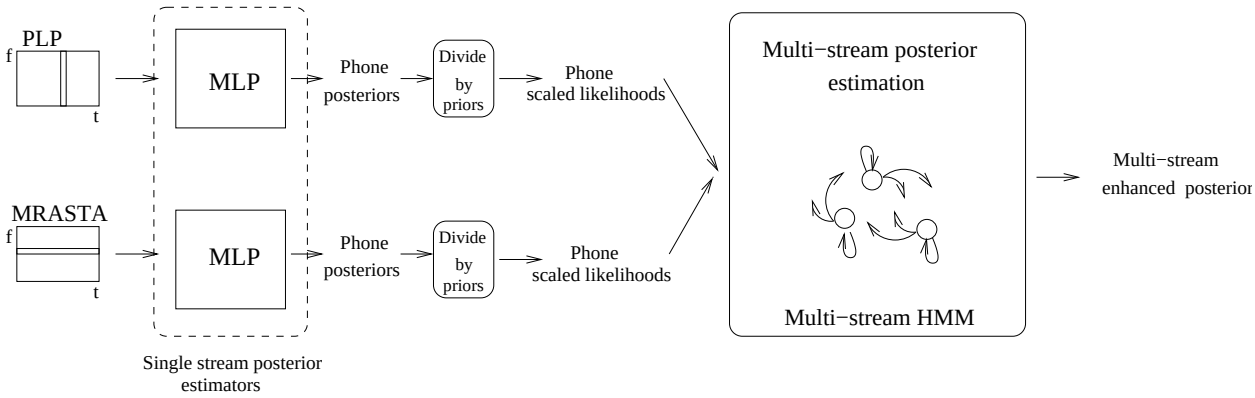


Figure 6.1: Multi-stream posterior estimation: Two streams of posteriors are estimated from PLP and MRASTA features using MLPs. These posteriors are then turned into scaled likelihoods by dividing by the priors. The resulting two streams of scaled likelihoods are fed to the multi-stream HMM. The multi-stream phone posteriors are estimated using multi-stream forward-backward recursions as described in Section 6.2.

a configuration similar to Tandem system.

6.3 Using Multi-stream Posteriors in ASR Systems

The multi-stream phone posteriors can be used as more informative posteriors in different speech processing systems. In this section, we investigate the use of these posteriors as features in a configuration similar to Tandem. Considering the block diagram of Figure 6.1, in the following we study different components of the multi-stream posterior estimation system in a more detailed and practical manner.

6.3.1 Input Streams of Features

The first step in developing the system is to choose two sets of features having complementary information. Spectral (cepstral) features and features having long temporal information are suitable candidates. We used PLP cepstral features [22] and MRASTA temporal

features [71] as input feature streams for the system. MRASTA features represent temporal energy pattern for different bands over a long context, while PLPs represent full short-term spectrum.

6.3.2 Single Stream Regular Posterior Estimation

In the first module in the diagram of Figure 6.1, the two input feature streams (PLP and MRASTA) are processed by MLPs to estimate posterior probabilities of context-independent phones. For PLP cepstral features, 9 frames of PLP coefficients and their first and second order derivatives are concatenated as the input for a trained MLP to estimate posterior probabilities of context-independent phones [2]. The phonetic class is defined with respect to the center of 9 frames. For the case of MRASTA, different bands temporal energy pattern over 1 second MRASTA temporal vector are first processed by band Gaussian filters. The outputs of these band filters are fed as inputs for a merger MLP [71]. The Merger MLP gives the posterior estimate for context-independent phones. Again, phonetic class is defined with respect to the center of 1 second temporal vector. In the remainder of the chapter, we call these single stream posterior estimates as PLP and MRASTA posteriors, respectively.

6.3.3 Multi-stream Enhanced Posterior Estimation

Having two stream of posteriors estimated from PLP and MRASTA features using MLPs, the next step is estimating state posteriors through the multi-stream HMM configuration. Posteriors are first divided by priors to obtain scaled likelihoods. These scaled likelihoods are then used in multi-stream forward-backward recursions according to (6.7, 6.9) to obtain estimates of state posteriors.

6.4 Experiments and Results

Results are presented on OGI digits [78, 79] and the reduced vocabulary version of the DARPA conversational telephone speech-to-text (CTS) task (1'000 words) databases [80]. We used PLP and MRASTA features as input streams to our system.

The PLP cepstral coefficients are extracted using 25 ms window with 10 ms shifts. At each frame, 13 PLP coefficients, their first-order and second-order derivatives are extracted and concatenated to make one feature vector.

For extracting MRASTA features, the short-term critical band spectrum is computed in 25 ms windows with 10 ms shifts and the logarithm of the estimated critical band spectral densities are taken. There are 15 bands. For each band, 50 frames before and after the center of analysis is taken resulting in 101 points long temporal MRASTA vector [71].

In this work, each phone is modeled by one state in the multi-stream HMM and we assume ergodic uniform transition probabilities between phones.

6.4.1 OGI Digits

The task is recognition of eleven words (American English Digits). The test set was derived from the subset of CSLU Speech Corpus [78, 79], containing utterances of connected digits. There are 2169 utterances (about 1.7 hours) in the test set. Training set contains 2547 utterances (about 1.2 hours). This set is also derived from CSLU Speech Corpus and utterances containing only connected digits are used. The standard HMM/GMM training/inference back-end system is based on HTK. There are 29 context-independent phonetic classes. The subset of OGI stories [79] plus a subset of OGI numbers [78] was used for training MLPs for single stream posterior estimation. This set has in total 3798 utterances with total length about 4.5 hours.

Two streams of posteriors (one from PLP features and the other one from MRASTA fea-

tures) are estimated as explained in Section 6.3.2 for the test and training set. They are then turned into scaled likelihoods and used in the multi-stream HMM module to get the estimates of enhanced state (phone) posteriors. These enhanced phone posteriors are gaussianized and decorrelated through Log and Karhunen-Loeve (KL) transforms, and fed as features to the standard HMM/GMM back-end. The standard HMM/GMM system is based on HTK [68]. For comparison purposes, we also run the standard HMM/GMM system using single stream regular posterior estimates as features (after Log and KL transforms) in order to obtain the baseline performance of single stream PLP and MRASTA posteriors before the combination. This corresponds to normal PLP-Tandem and MRASTA-Tandem systems. Moreover, we use an inverse entropy based combination method [97] (reported to perform well) to combine PLP and MRASTA posteriors, and compare the combination performance with our method. Table 6.1 shows the result of recognition studies. The first column shows the features which are fed to the standard HMM/GMM module. The second column shows the corresponding word error rates (WER). The first row shows the baseline performance of posteriors estimated using PLP features (the first stream). The second row shows the baseline performance of posteriors estimated using MRASTA features (the second stream). The third row shows the performance of features obtained by inverse entropy combination of PLP and MRASTA posteriors. The fourth row shows the performance of our system which uses multi-stream phone posteriors obtained by combining the mentioned streams of PLP and MRASTA posteriors through the multi-stream HMM. The system using multi-stream enhanced posteriors as features performs better than the systems using regular single stream posteriors (before the combination) and also inverse entropy based combination.

6.4.2 DARPA CTS Task

The use of multi-stream posterior estimation method was further evaluated on conversational telephone speech (CTS) recognition task [80]. The CTS database specifications and MLP posterior estimation details can be found in Section 4.1.

Features	WER
PLP posteriors	3.6%
MRASTA posteriors	4.8%
Inverse entropy combination	3.5%
Multi-stream enhanced posteriors	**2.9%**

Table 6.1: Word error rates (WER) on OGI Digits task. Results are shown for regular single stream PLP and MRASTA posteriors, their combination using inverse entropy, and finally multi-stream enhanced posteriors.

Features	WER
PLP posteriors	44.2%
MRASTA posteriors	51.0%
Inverse entropy combination	43.8%
Multi-stream enhanced posteriors	**41.9%**

Table 6.2: Word error rates (WER) on CTS task. Results are presented for regular single stream PLP and MRASTA posteriors, their combination using inverse entropy, and finally multi-stream enhanced posteriors.

Similar experiments as the case of OGI Digits database was repeated. Table 6.2 shows the recognition results. Again, multi-stream posterior combination gives noticeable improvement over PLP and MRASTA posteriors before the combination, and also inverse entropy combination.

6.5 Summary and Conclusions

In this chapter, we described the extension of the HMM-based enhanced posterior estimation to the case of multi-stream features. We explained how the posterior estimation can be enhanced by taking into account (combining) different streams of features, as well as possible prior information. We used these multi-stream enhanced posteriors as fea-

tures for a standard HMM/GMM system. We showed our system performs better than PLP-Tandem and MRASTA-Tandem baseline systems, and also inverse entropy combination method. The proposed theoretical framework provides a principled way for combining different streams of features by hierarchical posterior estimation, as well as introducing context and prior knowledge to get better (more informative) evidences in the form of posteriors.

Although not investigated here, the multi-stream state posteriors can be also used for re-estimating MLP parameters in the single stream posterior estimation modules. In this case, the MLPs used for estimating single stream posteriors are retrained with multi-stream enhanced posteriors as new targets.

6.6 Appendix

In this section, detailed expansion for some of the equations of the chapter is presented.

6.6.1 Multi-stream Forward Recursion

Using individual stream forward recursions α^n and applying the usual HMM assumptions, we can write multi-stream forward α recursion as follows:

$$\alpha(k,t) \triangleq \frac{p(x^1_{1:t}, x^2_{1:t}, ..., x^N_{1:t}, s^k_t)}{\prod_{\tau=1}^{t} p(x^1_\tau) \prod_{\tau=1}^{t} p(x^2_\tau) ... \prod_{\tau=1}^{t} p(x^N_\tau)} \tag{6.13}$$

$$= \frac{p(x^1_{1:t}, x^2_{1:t}, ..., x^N_{1:t} | s^k_t) p(s^k_t)}{\prod_{\tau=1}^{t} p(x^1_\tau) \prod_{\tau=1}^{t} p(x^2_\tau) ... \prod_{\tau=1}^{t} p(x^N_\tau)} \tag{6.14}$$

Adding the following independence assumption:

$$p(x^1_{1:t}, x^2_{1:t}, ..., x^N_{1:t} | s^k_t) = p(x^1_{1:t} | s^k_t) p(x^2_{1:t} | s^k_t) ... p(x^N_{1:t} | s^k_t)$$

We have:

$$\alpha(k,t) = \frac{p(x^1_{1:t}|s^k_t)p(x^2_{1:t}|s^k_t)...p(x^N_{1:t}|s^k_t)p(s^k_t)}{\prod_{\tau=1}^{t} p(x^1_\tau) \prod_{\tau=1}^{t} p(x^2_\tau)... \prod_{\tau=1}^{t} p(x^N_\tau)} \tag{6.15}$$

Applying Bayes rule $p(x^n_{1:t}|s^k_t) = \frac{p(x^n_{1:t},s^k_t)}{p(s^k_t)}$, and some algebraic manipulations give:

$$\alpha(k,t) = \frac{\frac{p(x^1_{1:t},s^k_t)}{\prod_{\tau=1}^{t} p(x^1_\tau)}}{p(s^k_t)} \frac{\frac{p(x^2_{1:t},s^k_t)}{\prod_{\tau=1}^{t} p(x^2_\tau)}}{p(s^k_t)} ... \frac{\frac{p(x^N_{1:t},s^k_t)}{\prod_{\tau=1}^{t} p(x^N_\tau)}}{p(s^k_t)} p(s^k_t) \tag{6.16}$$

Considering the definition of individual stream forward α^n recursion in (6.5) we have:

$$\begin{aligned} \alpha(k,t) &= \frac{\alpha^1(k,t)}{p(s^k_t)} \frac{\alpha^2(k,t)}{p(s^k_t)} ... \frac{\alpha^N(k,t)}{p(s^k_t)} p(s^k_t) && (6.17) \\ &= \frac{\prod_{n=1}^{N} \alpha^n(k,t)}{p(s^k_t)^{N-1}} && (6.18) \end{aligned}$$

6.6.2 Multi-stream Backward Recursion

The multi-stream β recursion can also be written using individual stream β^n recursions:

$$\beta(k,t) \triangleq \frac{p(x^1_{t+1:T}, x^2_{t+1:T}, ..., x^N_{t+1:T}|s^k_t)}{\prod_{\tau=t+1}^{T} p(x^1_\tau) \prod_{\tau=t+1}^{T} p(x^2_\tau)... \prod_{\tau=t+1}^{T} p(x^N_\tau)} \tag{6.19}$$

Adding the following independence assumption:

$$p(x^1_{t+1:T}, x^2_{t+1:T}, ..., x^N_{t+1:T}|s^k_t) = p(x^1_{t+1:T}|s^k_t)p(x^2_{t+1:T}|s^k_t)...p(x^N_{t+1:T}|s^k_t)$$

We have:

$$\beta(k,t) = \frac{p(x^1_{t+1:T}|s^k_t)p(x^2_{t+1:T}|s^k_t)...p(x^N_{t+1:T}|s^k_t)}{\prod_{\tau=t+1}^{T} p(x^1_\tau)\prod_{\tau=t+1}^{T} p(x^2_\tau)...\prod_{\tau=t+1}^{T} p(x^N_\tau)} \tag{6.20}$$

$$= \frac{p(x^1_{t+1:T}|s^k_t)}{\prod_{\tau=t+1}^{T} p(x^1_\tau)}\frac{p(x^2_{t+1:T}|s^k_t)}{\prod_{\tau=t+1}^{T} p(x^2_\tau)}...\frac{p(x^N_{t+1:T}|s^k_t)}{\prod_{\tau=t+1}^{T} p(x^N_\tau)} \tag{6.21}$$

Considering the definition of individual stream backward β^n recursion in (6.6) we have:

$$\beta(k,t) = \beta^1(k,t)\beta^2(k,t)...\beta^N(k,t) \tag{6.22}$$

$$= \prod_{n=1}^{N}\beta^n(k,t) \tag{6.23}$$

6.6.3 Multi-stream State Posterior Estimation

The multi-stream state posterior $\gamma(i,t)$ can then be obtained using multi-stream α and β recursions:

$$\gamma(k,t) \triangleq p(s^k_t|x^1_{1:T}, x^2_{1:T}, ..., x^N_{1:T}) \tag{6.24}$$

Applying Bayes rule we have:

$$\gamma(k,t) = \frac{p(x^1_{1:T}, x^2_{1:T}, ..., x^N_{1:T}, s^k_t)}{p(x^1_{1:T}, x^2_{1:T}, ..., x^N_{1:T})} \tag{6.25}$$

We also have:

$$p(x^1_{1:T}, x^2_{1:T}, ..., x^N_{1:T}) = \sum_j p(x^1_{1:T}, x^2_{1:T}, ..., x^N_{1:T}, s^j_t) \tag{6.26}$$

where the sum is over all the states of the HMM. Therefore,

$$\gamma(k,t) = \frac{p(x^1_{1:T}, x^2_{1:T}, ..., x^N_{1:T}, s^k_t)}{\sum_j p(x^1_{1:T}, x^2_{1:T}, ..., x^N_{1:T}, s^j_t)} \tag{6.27}$$

$$= \frac{p(x^1_{1:t}, x^1_{t+1:T}, x^2_{1:t}, x^2_{t+1:T}, ..., x^N_{1:t}, x^N_{t+1:T}, s^k_t)}{\sum_j p(x^1_{1:t}, x^1_{t+1:T}, x^2_{1:t}, x^2_{t+1:T}, ..., x^N_{1:t}, x^N_{t+1:T}, s^j_t)} \tag{6.28}$$

Applying product rule gives:

$$\gamma(k,t) = \frac{p(x^1_{1:t}, x^2_{1:t}, ..., x^N_{1:t}, s^k_t) p(x^1_{t+1:T}, x^2_{t+1:T}, ..., x^N_{t+1:T} | s^k_t)}{\sum_j p(x^1_{1:t}, x^2_{1:t}, ..., x^N_{1:t}, s^j_t) p(x^1_{t+1:T}, x^2_{t+1:T}, ..., x^N_{t+1:T} | s^j_t)} \tag{6.29}$$

Dividing the nominator and denominator by $\prod_{n=1}^{N} \prod_{\tau=1}^{t} p(x^n_\tau) \prod_{n=1}^{N} \prod_{\tau=t+1}^{T} p(x^n_\tau)$ gives:

$$\gamma(k,t) = \frac{\frac{p(x^1_{1:t}, x^2_{1:t}, ..., x^N_{1:t}, s^k_t) p(x^1_{t+1:T}, x^2_{t+1:T}, ..., x^N_{t+1:T} | s^k_t)}{\prod_{n=1}^{N} \prod_{\tau=1}^{t} p(x^n_\tau) \prod_{n=1}^{N} \prod_{\tau=t+1}^{T} p(x^n_\tau)}}{\frac{\sum_j p(x^1_{1:t}, x^2_{1:t}, ..., x^N_{1:t}, s^j_t) p(x^1_{t+1:T}, x^2_{t+1:T}, ..., x^N_{t+1:T} | s^j_t)}{\prod_{n=1}^{N} \prod_{\tau=1}^{t} p(x^n_\tau) \prod_{n=1}^{N} \prod_{\tau=t+1}^{T} p(x^n_\tau)}} \tag{6.30}$$

Considering the definitions of multi-stream forward and backward recursions in (6.7, 6.9) we have:

$$\gamma(k,t) = \frac{\alpha(k,t)\beta(k,t)}{\sum_j \alpha(j,t)\beta(j,t)} \tag{6.31}$$

Chapter 7

Higher Level Posteriors

In the previous chapters, our studies on the estimation of local posteriors have been mainly concentrated on phone posteriors. The framework of HMM-based posterior enhancement can be extended to the higher level case of local sub-word[1] or word posterior estimation. The HMM-based posterior estimation provides a general framework for estimating enhanced posteriors from the state up to the phone and word units. Local word posteriors can be obtained by integrating posteriors of states belonging to a word in the HMM model:

$$\begin{aligned} p(w_t^i|x_{1:T}, M) &= \sum_{j=1}^{N_s} p(w_t^i, s_t^j|x_{1:T}, M) \\ &= \sum_{j=1}^{N_s} p(w_t^i|s_t^j, x_{1:T}, M)p(s_t^j|x_{1:T}, M) \\ &= \sum_{j=1}^{N_s} p(w_t^i|s_t^j, x_{1:T}, M)\gamma(j,t) \end{aligned} \tag{7.1}$$

[1] In this case, the term 'sub-word' refers to sub-word units larger than phones, such as context-dependent phones, also called as triphones [45].

where w_t is a word at time t, w_t^i represents the event "$w_t = i$", and s_t^j is the event of being in state j at time t. $\gamma(j,t) = p(s_t^j|x_{1:T}, M)$ is the HMM state posterior estimated as described in Section 3.1. $p(w_t^i|s_t^j, x_{1:T}, M)$ represents the probability of being in a given word i at time t knowing to be in the state j at time t. Assuming that there is no parameter sharing between words, it is deterministic and equal to 1 or 0.

In this way, a word posterior at every frame $p(w_t^i|x_{1:T}, M)$ encoding phonetic and lexical knowledge can be obtained. We remind again that this is a 'local' word posterior, i.e. the posterior is estimated at every frame, although long contextual and prior information is taken into account for the estimation. In the following, we present initial investigations on the estimation and use of these higher level (word) posteriors through the practical case of keyword spotting. The basic idea behind our keyword spotting approach is estimating a keyword and a garbage posterior at every *frame*. These posteriors are then used to make a decision (vote) on detection of the keyword at each frame. The frame level decisions (votes) are then accumulated (in this case by counting) to make a global decision on having the keyword in the utterance. In this way, the contribution of possible outliers are minimized, as opposed to the conventional Viterbi decoding approach. In case of Viterbi decoding, the likelihood values with unlimited dynamic range are accumulated, while in our approach the frame level decisions (votes) are accumulated. Although here we focus on the case of keyword spotting, this approach can initiate a new family of decoders using local word posteriors (or word classes) for decoding in ASR.

7.1 Local Word Posteriors for Keyword Spotting

Word spotting is the detection of occurrences of selected words or phrases in speech. Hidden Markov model (HMM) based approaches have been extensively used for this task [99, 100, 101, 102, 103]. The conventional way of spotting keywords using the HMM configuration is Viterbi decoding. Each path in the HMM contains a sequence of keyword and non keyword units. Non keyword units are modeled by the so called 'garbage' model. The

decoder estimates likelihood scores for all possible paths, and the path with the highest score is selected as the output. This score is a global score accumulated over all likelihoods and transitions probabilities in the whole utterance. Therefore, strong outliers can highly contribute in the final global score, thus final decision made based on this score. Moreover, the score is not normalized with respect to the probability of the acoustic observation, thus it is relative to the particular acoustic observation [51]. It means that some factors such as the length of the utterance, the length of keyword and garbage elements, and the numerical range of likelihood values can affect the score. The values of these scores are penalized by changing keyword and garbage entrance penalties, which are acting as spotting thresholds. The optimal choice of these thresholds are obtained by empirically adjusting the operating point (trade-off between true and false alarms) to maximize the performance criteria on a development set.

Based on HMM-based posterior estimation framework presented in Section 3.1, we propose a new posterior based scoring approach for keyword and garbage units. In this scoring approach, a keyword and a garbage unit posterior is estimated at every frame. This posterior can be estimated through the same HMM configuration which is used in Viterbi decoding. The estimation of this posterior is based on HMM state posterior probability definition, taking into account prior knowledge (keyword model topology) and long contextual information. HMM state posteriors are estimated using forward-backward recursions. The state posterior probabilities are then integrated to keyword and garbage unit posteriors for each frame. This is a frame level (local) score for a keyword or garbage element and not a global score for the whole utterance. The estimation of these posteriors involves normalization with respect to the probability of acoustic observation, therefore the posteriors are irrelative to a particular acoustic observation space. These frame level posteriors are then used to make a frame level decision (vote) for the detection of the keyword. The frame level decisions are then accumulated (in this case by counting) to have a global decision about the detection of the keyword in the utterance. Therefore, the main difference between our approach and the Viterbi decoding approach is that in Viterbi decoding, the likelihood values with unlimited dynamic range are accumulated, while in our approach the frame level

decisions (votes) are accumulated. This leads to decreasing the contribution of possible outliers.

We show that the new posterior based scoring approach results in a better trade-off between true and false alarms, compared to the Viterbi based approach. Moreover, it provides the possibility to precalculate keyword specific spotting thresholds based only on the keywords length. Keyword length can be known a priori, or computed from the minimum length and number of phones composing the keyword. This property can be specially important in spotting new keywords without the need for huge development samples. In contrast, in the Viterbi based approach, there is no meaningful interpretation of thresholds (entrance penalties) in terms of priori known keyword characteristics. They should be adjusted empirically using a huge development set.

7.1.1 Modeling Garbage and Keyword Units

The HMM configuration used for keyword spotting is composed of a parallel connection of keyword and garbage unit models (Figure 7.1). Keyword models are created by connecting phone models based on the keyword lexicon. For garbage unit modeling, phone models are connected with uniform transition probabilities [65, 103]. Therefore, the whole HMM configuration is a parallel network of keyword models (left-to-right connection of phone models), and separate phone models (garbage unit model).

7.1.2 Keyword and Garbage Scoring

In this section, we study the traditional Viterbi scoring approach and its potential drawbacks for keyword spotting. We then propose our keyword and garbage scoring approach which is based on local word posterior estimation.

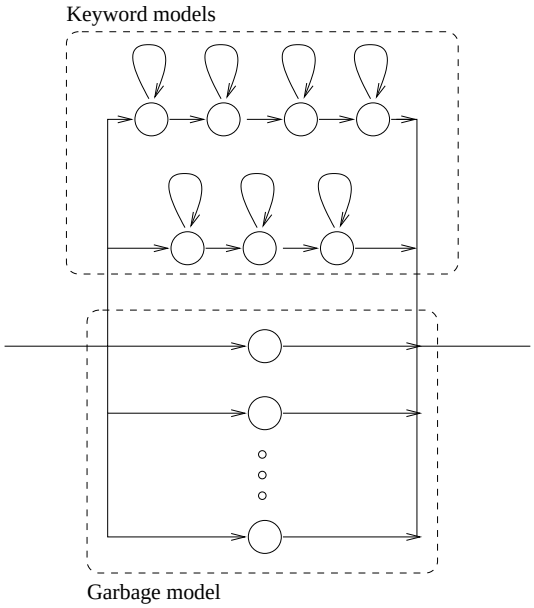


Figure 7.1: HMM configuration for keyword spotting. Keyword models are created by left-to-right connection of phone models. Garbage model is created by uniform connection of phone models.

Viterbi Based Scoring

The conventional scoring approach used for detecting keywords is Viterbi decoding through the HMM configuration [65, 99, 101, 103]. Each path in the decoder is a sequence of keyword and garbage elements. The decoder estimates the scores for all possible paths, and the one with the highest score is selected as the output. This score is related to the joint probability of the path and the feature vectors (evidences). The Viterbi scoring approach has the following drawbacks concerning the keyword spotting task:

- The score is a global score estimated by accumulating all likelihoods for the whole utterance, and it is not specific for a keyword or garbage unit. Therefore, the temporal outliers can strongly affect the final global score, and result in a wrong spotting case.

- The score is not normalized with respect to the probability of the acoustic observation, thus it is relative to the particular acoustic observation space [51]. For example, it

can be related to the length of the utterance, the length and number of keywords and garbage units, the numerical range of features, etc.

- The values of these scores are penalized by changing keyword and garbage entrance penalties, which are effectively spotting thresholds in this approach. However, there is no meaningful interpretation for the entrance penalty values, and they should be adjusted empirically to optimize the performance criteria. It implies the need to a sufficiently large development or training set for each keyword. Considering an application in which keyword set is not fixed or keyword samples are rare, having a huge development set is not always feasible. It would be ideal if we could find a reasonable threshold based on keyword characteristics such as length which can be known a priori or easily measured.

Posterior Based Scoring

Considering the framework of HMM-based posterior estimation presented in Section 3.1, we propose a frame level (local) posterior probability score for keyword and garbage units. This posterior probability can be estimated through the same HMM configuration which is used for the Viterbi decoding (as described in Section 7.1.1). The estimation of these posteriors is based on using forward-backward recursions for HMM state posterior estimation (described in Section 3.1). Assuming keyword and garbage models as word units, the state posteriors are then integrated to keyword and garbage posteriors at every frame. Having state posteriors $p(s_t^j|x_{1:T}, M)$ estimated through the HMM model M, the local posterior for keyword i can be obtained as follows:

$$\begin{aligned} p(K_t^i|x_{1:T}, M) &= \sum_{j=1}^{N_s} p(K_t^i, s_t^j|x_{1:T}, M) \\ &= \sum_{j=1}^{N_s} p(K_t^i|s_t^j, x_{1:T}, M)p(s_t^j|x_{1:T}, M) \end{aligned} \tag{7.2}$$

where N_s is the total number of HMM states, K_t is a keyword at time t, and K_t^i represents the event of having keyword i at frame t. $p(K_t^i|s_t^j, x_{1:T}, M)$ represents the probability of being in a given keyword i at time t knowing to be in the state j at time t. Assuming that there is no parameter sharing between words, it is deterministic and equal to 1 or 0.

Considering garbage model as a word unit, the same approach can be used to estimate a frame level posterior score for the garbage unit:

$$\begin{aligned} p(G_t|x_{1:T}, M) &= \sum_{j=1}^{N_s} p(G_t, s_t^j|x_{1:T}, M) \\ &= \sum_{j=1}^{N_s} p(G_t|s_t^j, x_{1:T}, M)p(s_t^j|x_{1:T}, M) \end{aligned} \tag{7.3}$$

where G_t is the garbage unit at time t, and $p(G_t|s_t^j, x_{1:T}, M)$ represents the probability of being in a state j (which is a part of garbage model) at time t. Again, this probability is deterministic and equal to 1 or 0.

In this way, keyword posterior score at time t, $p(K_t^i|x_{1:T}, M)$, and a garbage posterior score at time t, $p(G_t|x_{1:T}, M)$ is obtained. Comparing with the Viterbi decoding approach, the new scoring approach provides the following advantages:

- It provides a frame level keyword or garbage specific score, instead of a global score for the whole utterance. A local keyword posterior can not be high without having a high emission probability (local evidence) for the keyword, while the score in the decoder based approach is global and can be affected by several different factors.
- This score is normalized with respect to the probability of acoustic observation (see Equation 3.4), and thus irrelative to the particular observation sequence.
- Having frame level normalized scores provides the possibility of relating the spotting thresholds to length of keywords (explained in more details in the next section).

Next section describes how these local posteriors are used to decide about detection of a keyword in the utterance.

7.1.3 Keyword Detection and Threshold Precalculation

Having the frame level keyword or garbage posteriors, the next step is to decide about existence of the keyword in the utterance. Figure 7.2 shows a block diagram of the approach. The frame level keyword and garbage posteriors $p(K_t|x_{1:T}, M)$ and $p(G_t|x_{1:T}, M)$ are compared to make a frame level decision (vote) about the detection of the keyword[2]. The frame level decisions are then accumulated, in this case by counting continuous frame level keyword detections. The outcome is the detected length of the keyword in the utterance. The main difference between our approach and the Viterbi decoding approach is accumulating frame level decisions (votes) instead of frame level likelihoods (with unlimited dynamic range). This leads to decreasing the contribution of possible outliers compared to Viterbi decoding.

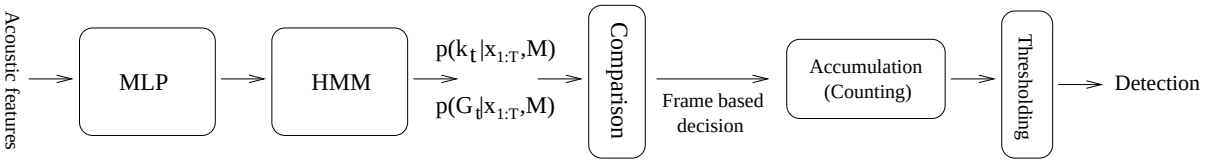


Figure 7.2: Block diagram of posterior based keyword spotting approach. The frame level keyword and garbage posteriors $p(K_t|x_{1:T}, M)$ and $p(G_t|x_{1:T}, M)$ are estimated through the HMM model. These posteriors are compared, yielding a frame level decision (vote) on detection of the keyword. The frame level decisions are then accumulated (by counting). The resulting score is compared with a length-based threshold to decide about detection of the keyword in the utterance.

As mentioned, the above process provides a score showing the detected length of the keyword in the utterance. Therefore, the spotting threshold can be precalculated based on the

[2]Assuming only one keyword, we can drop the superscript i in $p(K_t^i|x_{1:T}, M)$ and write it as $p(K_t|x_{1:T}, M)$.

length of the keyword. The length of a keyword can be known a priori (using a few samples of the keyword) or computed using minimum duration of phones in the keyword. This threshold can be further adjusted having in mind that it is related to the keyword length, in order to achieve different desired operating points. This can be important for a practical keyword spotting system specially if the keyword set is not fixed, or if the task is spotting names or words which are not appearing frequently in the database. In these cases, we cannot have a huge development set for each new keyword and new condition to properly adjust the spotting thresholds. Therefore, precalculating keyword specific thresholds based on priori known characteristics of the keywords (e.g. length) can be useful.

7.1.4 Experiments and Results

In this section, we present preliminary experiments to evaluate the idea of using word posteriors for keyword spotting. For the experiments, we create garbage and keyword unit models based on phone models (as described in Section 7.1.1). We compare the Viterbi scoring approach with the new posterior based scoring approach for spotting keywords.

We used conversational telephone speech (CTS) [80] and Numbers'95 [78] databases for the experiments. The specifications of the databases can be found in Chapter 4. The acoustic feature vectors are PLP cepstral coefficients and their first and second order derivatives. The HMM emission probabilities are phone posteriors estimated by a MLP (hybrid HMM/MLP configuration). Detailed information on phone posterior estimation can be found in Chapter 4.

We have used 9 keywords from the CTS database and 6 keywords from Numbers'95 database. These keywords are 'you', 'yeah', 'like', 'think', 'something', 'because', 'people', 'play', 'night', 'one', 'five', 'four', 'fifteen', 'seven', and 'zero'. Their selection is based on having a large variability in terms of frequency, number of phones and length.

In the first set of experiments, the performance of our posterior based scoring system is

compared with the Viterbi decoder based system in terms of trade-off between true and false alarms. The HMM configuration is the same for the two methods. We use receiver operating characteristic (ROC) curves (as used conventionally for keyword spotting evaluation) in order to measure and compare the performance of the two systems. Figure 7.3 shows ROC curves obtained by the two methods for different keywords[3]. In most of the cases, the area under the curve is larger for the posterior based approach, indicating better trade-off between true and false alarms. In the Viterbi based approach, the score which is used to decide about detecting a keyword is a global score obtained for the whole utterance, and accumulated over all the evidences for garbage and keywords, transition probabilities, etc. Therefore, even when there is no keyword in the utterance, a 'fake' existence of a keyword can be possibly made by a strong temporal outlier (having very large or very small likelihood). In contrast, a temporal outlier in the posterior based approach can only affect few frame level decisions (votes), thus less probable to lead in a wrong spotting case.

In the second group of experiments, we study the relation between the spotting rates and the thresholds for the two approaches, and the possibility of precalculating keyword specific thresholds in the posterior based system. Figure 7.4 shows this relation for some of the keywords with different lengths. The threshold for the posterior based system is the period of continuous frame level keyword detection (in frames), while the threshold for the decoder based approach is the entrance penalty value. As can be seen, the threshold for the posterior based system is a meaningful value related to the length of the keyword, as long words (e.g. fifteen) need higher threshold while shorter words need less. In contrast, it is not easy to find a meaningful interpretation of thresholds for the decoder based system. The first row of Figure 7.4 shows a practically very useful behaviour about the relation between the spotting threshold and performance in our approach. As can be seen, in all the plots there is a turning (inflection) point indicated by 'x'. Figure. 7.5 is showing the case for the keyword 'zero' as a sample to describe the behaviour better. The true alarm rate before this turning point is almost fixed and equal to the maximum achievable true

[3]In order to have a rough idea about the difficulty of these tasks (CTS and Numbers'95), it is useful to mention that the baseline speech recognition performance for CTS and Numbers'95 databases are about 55 and 95 percent recognition rate, respectively.

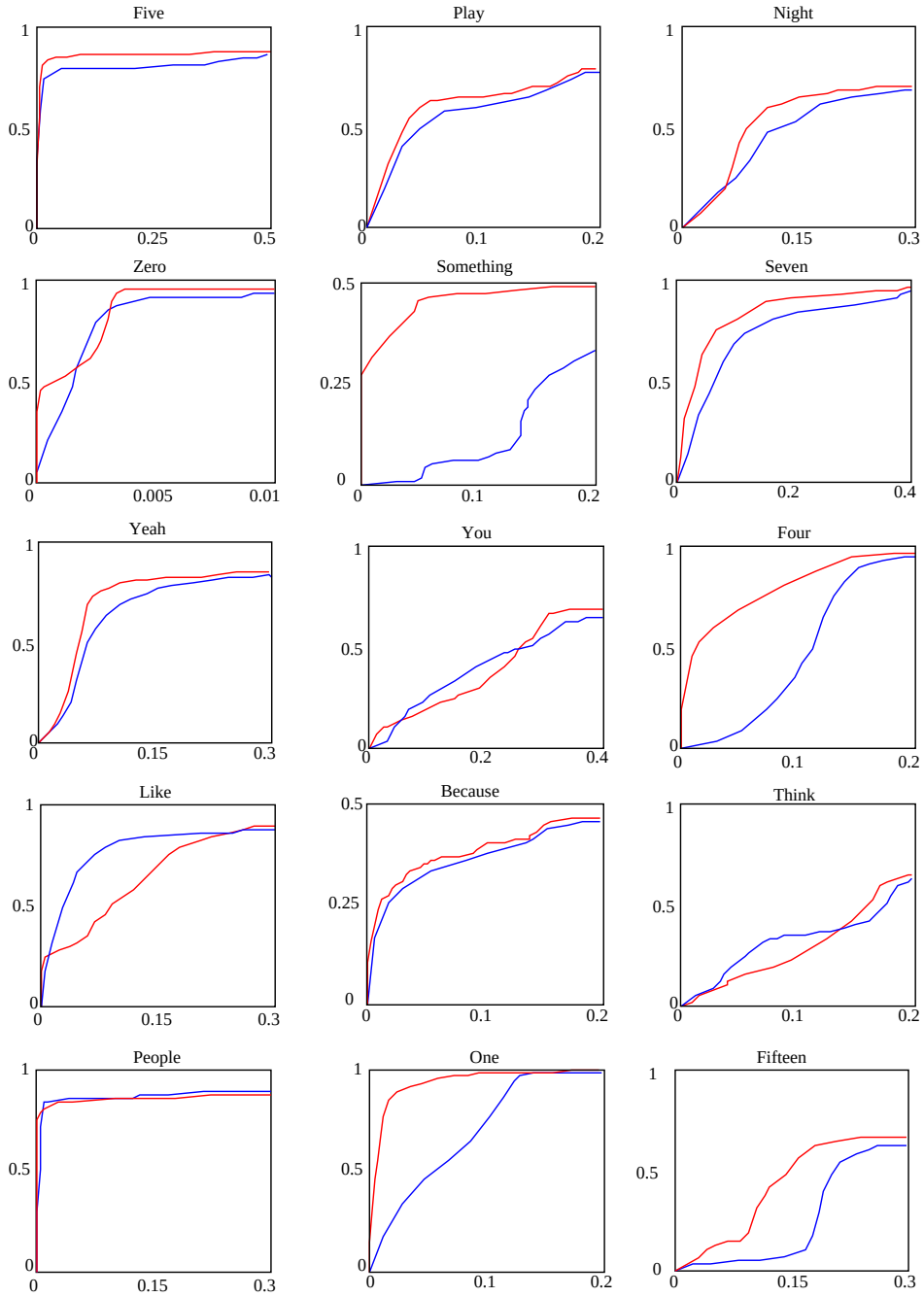


Figure 7.3: ROC curves for different keywords. The blue curves show Viterbi based results and red curves show posterior based approach results. The y axis is the number of true alarms normalized by the number of keyword samples in the database, and the x axis is the number of false alarms normalized by the number of word samples. In all the plots, the region that the behaviour of the curves changes is shown. For larger values of false alarms, the behaviour of the red and blue curves is similar.

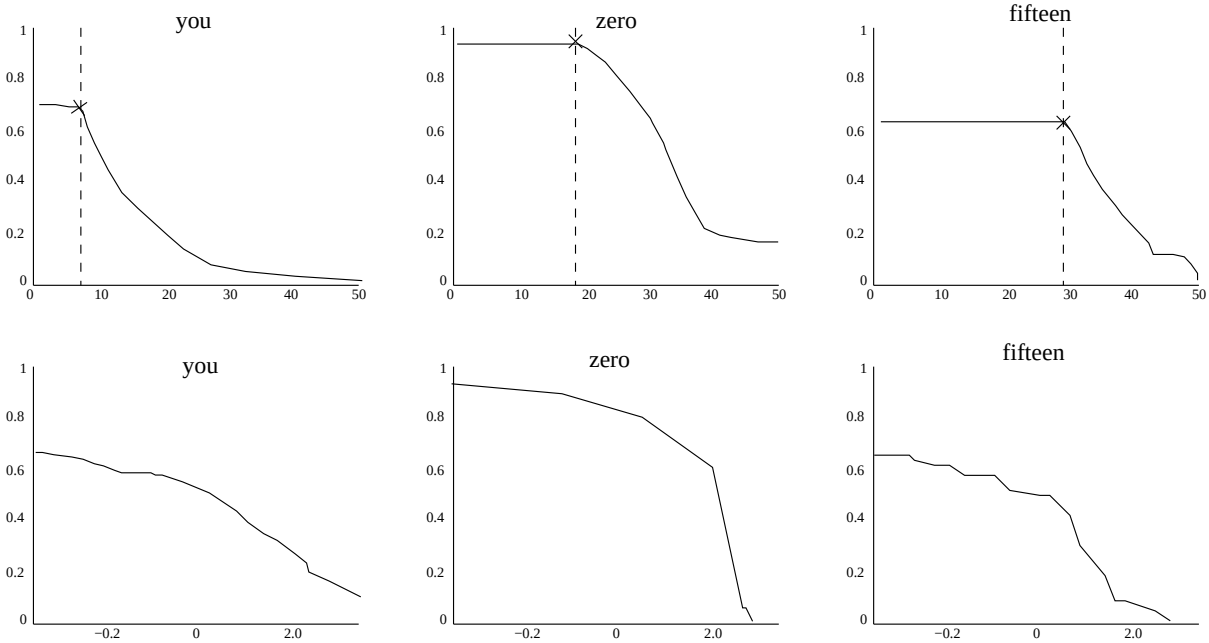


Figure 7.4: Relation between spotting rates and thresholds for the two methods. The first row is showing posterior based approach and the second row shows Viterbi based approach. The y axis shows the spotting rates and the x axis shows the thresholds.

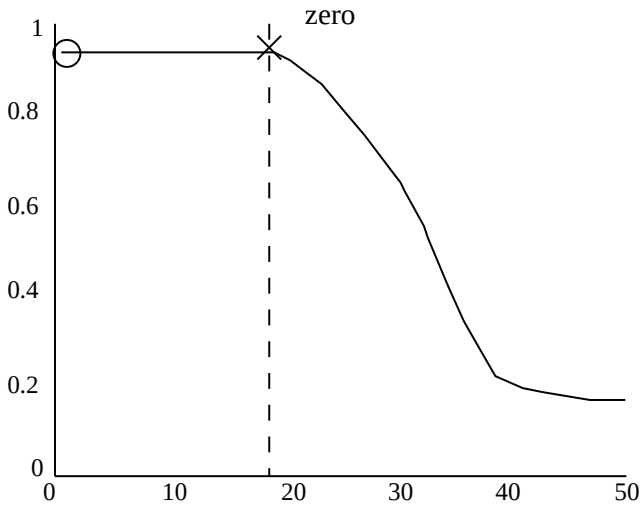


Figure 7.5: The relation between TA rate and threshold for keyword ‘zero’. Vertical axis shows TA rate and horizontal axis shows thresholds.

alarm rate[4], while it drops rapidly after the turning point. Similar behaviour is observed for most of the keywords, although not plotted here. At the point indicated by 'o', the false alarm is maximum. Moving from the point 'o' towards the turning point, the false alarm rate reduces, while the true alarm rate is unchanged (maximum). At the turning point, the true alarm rate is still close to maximum, while the false alarm rate is minimum for this condition. This can be the best operating point concerning a practical application. Our studies have shown that the threshold at the turning point is very close to the estimated average length of the keyword, and higher than the minimum estimated length. This is the key point to precalculate a practically useful threshold based on the length of each keyword. The minimum and average keyword length can be measured using few samples of the keyword, or estimated based on minimum and average length of the phones composing the keyword. Table 7.1 is clarifying this issue. It shows the performance of the posterior based system obtained with precalculated thresholds for different words. The last column in the table shows the maximum achievable spotting rate with the posterior based approach. The second column shows the true and false alarm rates when the thresholds are set to the minimum length of the keywords. The minimum length of a keyword is assumed to be equal to sum of the minimum length of its phones. The third column shows the true and and false alarm rates when the threshold is set to the average length of each keyword. The average keyword length is estimated based on the the average length of the phones composing the keyword. As can be seen, the true alarm rate is similar (close) for the thresholds equal to minimum or average keyword length, while the false alarm is minimized when changing the threshold from minimum to the average length (conditioned on keeping the true alarm maximum). This behaviour can be practical for keyword spotting systems, as the threshold can be precalculated based on the average or minimum length of the keyword. The precalculated thresholds can be adjusted further based on the desired trade-offs, taking into account that they are related to the length of keywords. In contrast, since the score in the decoder based approach is related to several factors (as mentioned in

[4]Obviously, the maximum achievable true alarm rate is 1.0 when the length based threshold is set to 0, and the system starts spotting all the words as keyword. However, what is meant here by the maximum achievable rate is maximum excluding this case, i.e. maximum true alarm rate with a threshold higher than 0.

Section 7.1.2), the spotting threshold is also a complex function of those factors. Therefore, the threshold precalculation cannot be applied in this case, and it is necessary to have a huge development set for any new keyword to adjust the thresholds.

Keyword	TA and FA rate for min length threshold	TA and FA rate for average length threshold	Max TA rate
one	(9) 0.98 - 0.10	(12) 0.94 - 0.07	0.99
four	(9) 0.95 - 0.22	(13) 0.94 - 0.17	0.96
five	(9) 0.82 - 0.20	(13) 0.80 - 0.10	0.85
zero	(12) 0.95 - 0.02	(18) 0.95 - 0.01	0.96
fifteen	(21) 0.64 - 0.33	(27) 0.63 - 0.25	0.64
you	(5) 0.69 - 0.40	(7) 0.67 - 0.30	0.71
yeah	(6) 0.81 - 0.25	(9) 0.79 - 0.18	0.85
like	(9) 0.84 - 0.32	(13) 0.83 - 0.28	0.87
think	(12) 0.65 - 0.25	(15) 0.62 - 0.20	0.69
people	(15) 0.80 - 0.05	(19) 0.76 - 0.01	0.86
because	(15) 0.49 - 0.28	(20) 0.47 - 0.20	0.53
something	(18) 0.61 - 0.96	(25) 0.58 - 0.40	0.66

Table 7.1: True alarm (TA) and false alarm (FA) rates for different keywords and different length based thresholds. The spotting thresholds are set to the minimum keyword length in column 2, and the average keyword length in column 3. The first number (inside bracket) in columns 2 and 3 is showing the value of threshold (length values are in frames), and the two other numbers are TA and FA rates respectively. The last column shows the maximum achievable TA rate for each keyword (the threshold is higher than 0).

7.2 Summary, Conclusions and Future Work

In this chapter, we described how the framework of HMM-based posterior estimation can be extended to the case of local word posterior estimation. We presented initial investigations on this issue through the practical case of keyword spotting. We proposed estimating frame level (local) posterior based scores for keyword and garbage units. The frame level keyword and garbage posteriors are then used to make a frame level decision about detecting the keyword. These frame level detections are accumulated (by counting) to make a global decision for having the keyword in the utterance. Comparing with the Viterbi

decoding approach which makes a global decision by accumulating likelihoods, here we make a global decision based on frame level decisions. In our approach, outliers can only affect few frame level decisions, while in the Viterbi based approach, they can noticeably affect the whole global score. We showed that the new posterior based scoring approach results in a better trade-off between true and false alarms. In addition, we also studied the relation between spotting rates and the thresholds for the posterior based and Viterbi based approaches. We showed that the posterior based approach provides the possibility to precalculate keyword specific spotting thresholds based on the length of the keywords. In contrast, thresholds in the Viterbi based approach are ad-hoc meaningless parameters. The baseline systems used in this chapter are adequate for the comparison within the scope of this work to highlight the contributions. However, it would be insightful to apply the proposed keyword spotting method for more competitive keyword spotting systems.

Although in this chapter we studied the case of keyword spotting, the same local word posterior scores can be possibly used for a general case of speech recognition. At every frame, posteriors of all the words in the vocabulary are estimated. This yields a word posterior vector $\bar{w}_t$ at every frame t. Each element of this vector is associated with a word in the vocabulary. These word posteriors can be used in different ways to decode the word sequence in the utterance. The simplest case is to use the word posteriors as local scores in a Viterbi decoder. The decoder combines the word posterior scores with the language model. The advantage of this approach can be minimizing the effect of temporal outliers. In addition, as opposed to likelihood scores in the classical decoding, the posterior based scores are normalized with respect to different sources of variability in the utterance. This new decoding approach can initiate a new alternative family of decoders accumulating local decisions (votes) instead of local acoustic scores (likelihoods).

Chapter 8

Comparing Enhanced and Regular Posteriors

In this chapter, we study the difference (deviation) between the regular and enhanced posteriors. Since enhanced posteriors are obtained by enriching the regular posteriors with prior knowledge and context, the difference can indicate mismatch between the prior knowledge and regular posteriors content. The regular MLP posteriors $p(q_t^i|x_t)$ represent data as a sequence of phone evidences. The HMM-based enhanced posteriors $p(q_t^i|x_{1:T}, M)$ can be considered as the MLP phone posteriors enriched by phonetic and lexical knowledge M, and context $x_{1:T}$. Therefore, comparing the two posteriors, the difference can indicate the cases that the data (represented by MLP posteriors) does not match the assumed prior phonetic and lexical knowledge. Since the two posteriors are estimated at every frame, we can have a frame level measure of deviation, thus a frame level measure of match/mismatch between data and phonetic/lexical knowledge. The deviation can be measured using Kullback-Leibler (KL) divergence. The KL divergence measure between the two posteriors can be used to detect inconsistency between data and model at different levels. We refer to this measure as "deviation" measure in this chapter.

One of the important applications of measuring this inconsistency can be detecting out-of-vocabulary (OOV) words for posterior based ASR systems [104]. In case of an OOV, the lexical knowledge does not match an existing sample of data, resulting in large deviation (KL divergence) between the two posteriors. In this chapter, we present initial investigations on the use of the mentioned deviation measure through the practical case of OOV word detection. We refer to this approach as "deviation based" approach. Confidence measurement is one of the state-of-the-art approaches for detecting OOV words, as an OOV word can result in low confidence level for a hypothesized recognizer output. We compare the mentioned deviation measure with the conventional posterior based confidence measures (PCMs) [50, 51] for detecting OOV words.

8.1 Detecting Out-of-Vocabulary Words

One of the most serious problems of the current ASR systems is their poor ability in dealing with out-of-vocabulary (OOV) words [105, 106]. A word that is not in the dictionary of the recognizer is likely to be replaced in the output of the recognizer by the high prior probability word that is in the dictionary and is emphasized by the language model. This undesirable property could have disastrous consequences on the utility of the recognizer in applications such as speech data mining or information summarization, since the OOV words (and in general low probability words) could have high information value. OOV words are not necessarily rare words. A word can be OOV for a specific small vocabulary task, scenario or conversation situation but can be common in general. OOV word detection can be essential for small vocabulary tasks (specific applications), as well as large vocabulary.

One state-of-the-art approach to address the OOV word problem in the posterior based ASR is to identify potentially misrecognized words from the low confidence of the recognition results [50, 51, 107, 108]. In posterior based ASR, one indicator of confidence is derived by using the recognizer output hypotheses (aligned state sequence), and evaluating a normal-

ized average likelihood or posterior measure inside the detected phone and word segments. The segmentation is obtained by back-tracking alignment of the recognized utterance. We refer to these measures as conventional posterior based confidence measures (PCMs). Relying explicitly on the recognition and phone segmentation results of the recognizer is the main disadvantage of these measures. The effectiveness of these measures is sensitive to correct and precise recognition of phone segment boundaries.

In this chapter, we present an alternative approach that does not require explicit recognition or segmentation (decisions about phone segment boundaries) in the utterance. Instead, two streams of local phone posterior probabilities are compared based on the measure of similarity between their distribution. One stream of probabilities is derived solely from the acoustic evidence by trained MLP, referred as "regular posteriors" . The second stream is derived from acoustic evidence together with higher level prior knowledge (e.g. lexical knowledge as available for the existing recognizer) and long acoustic context, referred as "enhanced posterior". It is estimated as described in Section 3.1.

The comparison of these two local posteriors provides a frame level measure of the match between the acoustic information and prior knowledge. A significant mismatch can indicate an OOV word. Unlike PCMs, the new measure does not use explicit phone and word recognition and segment boundary detection, thus it is not affected by imperfect recognition and segmentation.

As we will show later, PCMs can be considered as special case of our deviation based measure where the enhanced phone posteriors are replaced with binary (0 or 1) values obtained from recognition and phonetic segmentation.

In the following, Section 8.1.1 reviews regular posterior estimation. Section 8.1.2 reviews the integration of lexical knowledge in the posterior estimation. Section 8.1.3 deals with the way the two posterior streams are compared to yield a measure for detecting OOV words. Section 8.1.4 presents initial experiments and results, and compares the performance of the new measure with PCMs. Section 8.1.5 discusses the relation between the

deviation based measure and PCMs.

8.1.1 Regular Phone Posterior Estimation

As discussed in Section 2.2, MLPs provide regular phone posterior probabilities $p(q_t^i|x_t)$ which are driven by acoustic features (data) and independent of the long context or prior knowledge. Regular phone posteriors can be considered as a sensory stream, representing data as a sequence of phone posteriors. These phone posteriors are estimated only from a limited span of acoustic feature frames, without taking into account prior lexical knowledge.

8.1.2 Enhanced Posterior Estimation: Integrating Lexical Knowledge

Enhanced phone posteriors are derived not only from the acoustic input but also by integrating prior lexical knowledge. Subsequently, the acoustic evidence that match the prior and contextual knowledge is emphasized and the evidence that does not support it is suppressed. As studied in Section 3.1, these enhanced posteriors are estimated through a HMM configuration using forward-backward algorithm. It was shown that we can estimate the enhanced phone posterior $p(q_t^i|x_{1:T}, M)$, where q_t^i is the event of having phone i at time t, $x_{1:T}$ is the acoustic context as available in the whole utterance, and M is the HMM encoding specific lexical knowledge. Regular MLP posteriors are used as emission probabilities for the HMM/ANN module which integrates prior lexical and contextual knowledge. The HMM can be considered as a filter suppressing the acoustic evidences which does not match the prior lexical knowledge. As a consequence, when encountering an OOV word, the evidence representing the OOV word is suppressed, because of no match with the prior knowledge. Therefore, the enhanced posteriors stream deviates from the regular posteriors, indicating the OOV word.

8.1.3 Comparing Enhanced and Regular Posteriors

In order to detect OOV words, the difference between the two types of posteriors (regular and enhanced) is measured. This difference then yields an estimate of match/mismatch between data and prior lexical knowledge. In this work, we use Kullback-Leibler (KL) divergence to evaluate the difference between the two types of posteriors. KL divergence is suitable for measuring similarity of two probability distributions:

$$\begin{aligned} KL(\overline{C_t}, \overline{S_t}) &= \sum_i C_t^i log_2 \frac{C_t^i}{S_t^i} \\ &= \sum_i p(q_t^i|M, x_{1:T}) log_2 \frac{p(q_t^i|M, x_{1:T})}{p(q_t^i|x_t)} \\ \overline{S_t} &= p(q_t^i|x_t) \\ \overline{C_t} &= p(q_t^i|M, x_{1:T}) \end{aligned} \tag{8.1}$$

where $\overline{S_t}$ is a vector of regular MLP posteriors at time t, and $\overline{C_t}$ is a vector of enhanced posteriors at time t. S_t^i and C_t^i show the ith element of the posterior vectors at frame t. We refer to the KL divergence between the two posteriors as "deviation" measure.

The frame level deviation measures are then smoothed by a moving average filter to remove the effect of short term mismatches, and emphasize on word-level mismatch between two posterior streams. An OOV word is indicated by the increase in smoothed deviation measures (KL divergence) above a pre-set threshold. Another alternative approach can be accumulating the deviation measures inside the word hypothesis, followed by the normalization with respect to the length of the word. This is the same as setting the length of the moving average filter equal to the length of the word. In this case, we need word segments but still no need to have the phone segment boundaries. Phone segments can be very short

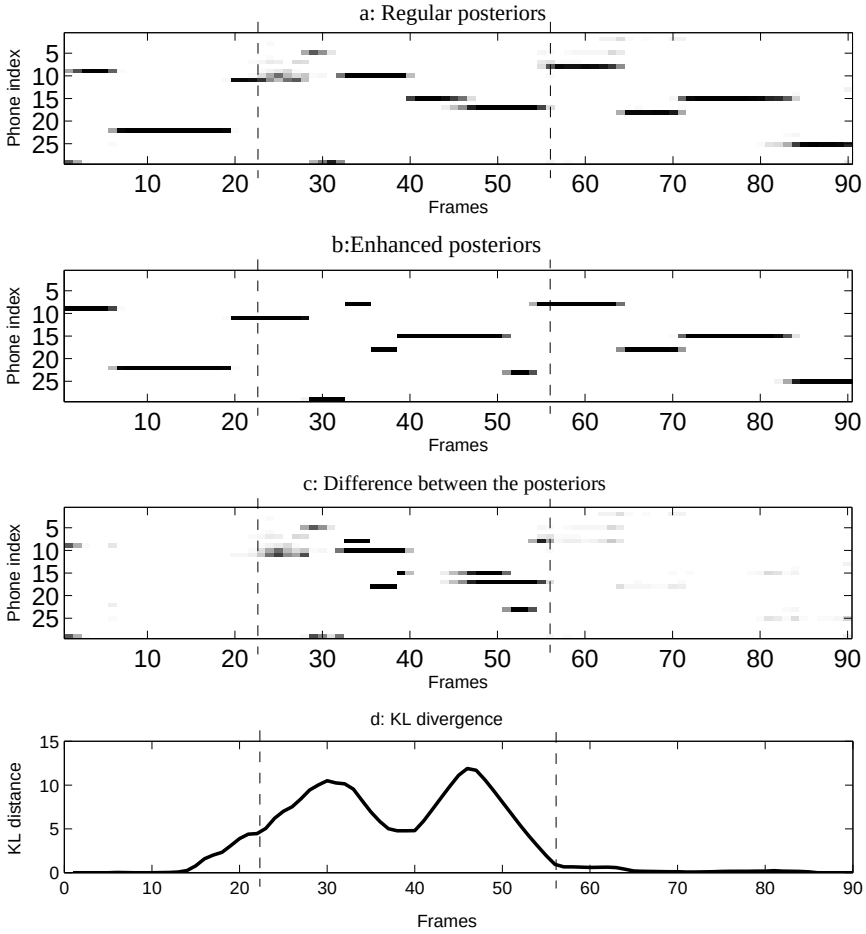


Figure 8.1: (a) Regular posteriors, (b) enhanced posteriors integrating lexical knowledge, (c) difference between regular and enhanced posteriors, and (d) deviation (KL divergence) between regular and enhanced posteriors. The utterance is 'five three zero', where 'three' has been assumed as the OOV word.

and are not precisely detected, thus they can be a major source of low performance in the PCM estimation.

A sample of regular and enhanced posteriors, their difference and their deviation (KL divergence) over time is shown in Figure 8.1. The utterance contains 'five three zero' where the word 'three' represents an OOV word, not present in the vocabulary. Figure 8.1.a shows regular posteriors for this utterance, 8.1.b shows the enhanced posteriors integrating lexical knowledge, and 8.1.c shows the difference between 8.1.a and 8.1.b. As it can be seen,

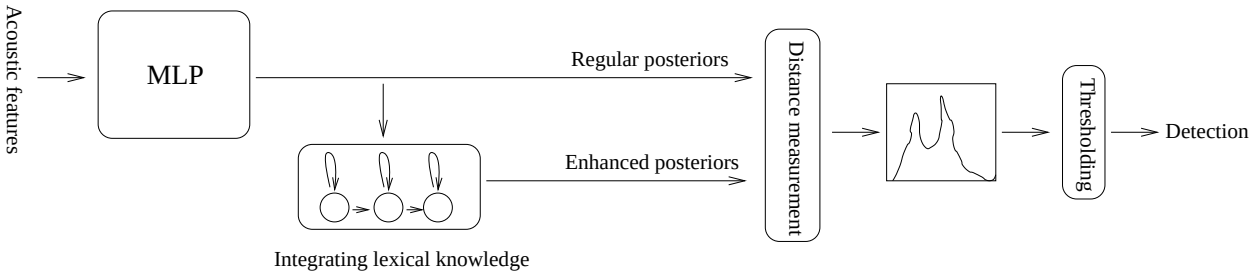


Figure 8.2: The configuration for our deviation based OOV word detection method. Regular posteriors are estimated by an MLP. Enhanced posteriors are estimated using a HMM/ANN module integrating prior lexical knowledge. The two posterior streams are compared by measuring the deviation (KL divergence) between the posterior vectors at each frame. The deviation measures are then compared with a threshold to decide on having OOV word.

there is a region with major difference corresponding to the word 'three' (marked roughly by dashed lines). Figure 8.1.d shows the deviation measure (KL divergence) between the two posteriors. A peak in deviation values corresponding to the word 'three' can be observed.

Figure 8.2 shows a diagram of the whole system: The regular phone posteriors are estimated by an MLP. The enhanced posteriors are estimated using a HMM integrating prior lexical knowledge based on a dictionary. The HMM module uses the regular MLP posteriors as state emission probabilities. The content of the two posterior streams (regular and enhanced posteriors) are compared based on measuring deviation (KL divergence) at each frame. The deviation measure is considered as a frame level measure for the correctness of the recognizer output. The deviation measures are then smoothed and compared with a threshold to detect OOV words.

8.1.4 Experiments and Results

In this section, we report initial results for detecting OOV words using the presented method. We have used OGI digits database [79] for the experiments. Database specifi-

cations can be found in Section 6.4.1. The MLP based MRASTA method [71] was used to estimate regular phone posteriors. MRASTA feature extraction details can be found in Section 6.4.1.

For estimating enhanced posteriors (integrating lexical knowledge), the regular phone posteriors are used as emission probabilities for a HMM/ANN module. The role of this module is to integrate prior lexical knowledge. The topology of this HMM/ANN module contains all the words in the vocabulary except the one that was removed (assumed as an OOV word). All digits take their turns to represent the OOV word. The regular and enhanced posterior vectors are compared frame by frame by measuring the deviation (KL divergence). The deviation measures are then smoothed by a moving average filter with the length of 10 frames. The smoothed deviation measures are compared with a threshold to make a decision on detecting OOV words. The alternative approach is using a moving average filter equal to the length of the word.

We have compared our deviation based approach with a group of conventional posterior based confidence measures (PCMs) presented in the literature [50, 51], and also studied in Section 5.2. These confidence measures are based on recognition and segmentation of the utterance into phones and words (by back-tracking alignment of the recognized utterance), and evaluating a posterior based measure inside the detected segments for the hypothesized word. The most typical ones, normalized posterior based confidence measures (NPCMs), are defined as follows:

$$phone-basedNPCM(w) = \frac{1}{L}\sum_{l=1}^{L}(\frac{1}{e_l-b_l+1}\sum_{t=b_l}^{e_l}\log p(q_t^l|x_t)) \quad (8.2)$$

$$frame-basedNPCM(w) = \frac{1}{\sum_{l=1}^{L}(e_l-b_l+1)}\sum_{l=1}^{L}\sum_{t=b_l}^{e_l}\log p(q_t^l|x_t) \quad (8.3)$$

where L is number of phones in the hypothesized word, q_t^l is phone l at time t, and e_l and b_l are the beginning and the end of each phone hypothesis q^l.

The performance of the individual systems is measured in terms of the trade-off between true and false alarms for detecting OOV words. We have introduced each of the words individually as an OOV word by removing it from the vocabulary. Figure 8.3 shows the receiver operating characteristic (ROC) curves obtained by our method, and conventional posterior based methods (NPCM measures). Our approach shows noticeably larger area under the ROC curve, indicating better trade-off between true and false alarms.

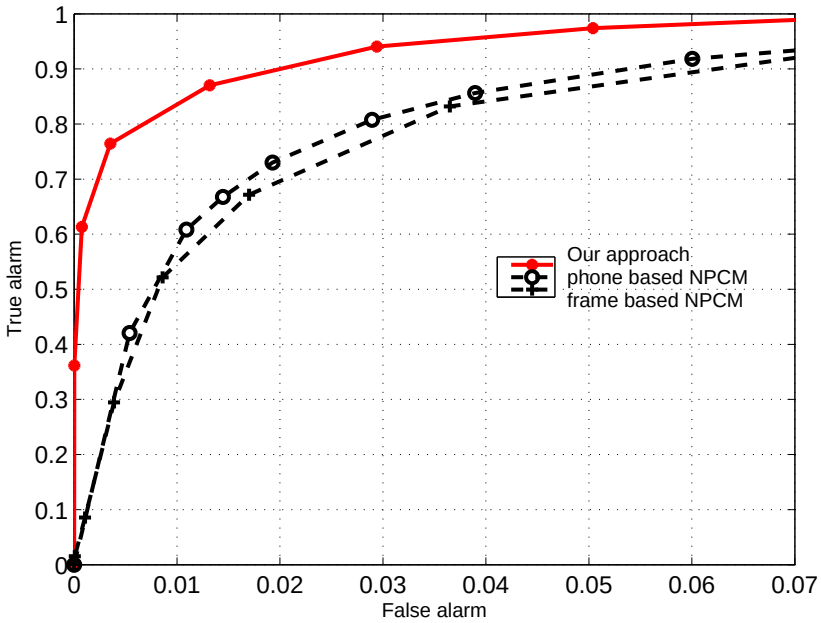


Figure 8.3: ROC curves for our deviation based approach, and conventional confidence measures (phone-based and frame-based NPCM). The y axis is showing true alarms and the x axis is showing false alarms. The number of true alarms is normalized with respect to the number of OOV samples, and the number of false alarms is normalized with respect to total number of words in the test set. Our approach shows better trade-off (larger area under the ROC curve).

8.1.5 Discussion

The deviation based measure can be considered as a generalization of the PCM measures. Here we show that if the enhanced posterior term $p(q_t^i|x_{1:T}, M)$ in the deviation measure (KL divergence) is replaced with the phonetic segmentation/recognition obtained from the recognizer output, there is a close relation between the deviation measure and PCM mea-

sures. Having q^l as a phone hypothesis in the interval $b_l \leq t \leq e_l$, we define a vector $\overline{L_t}$ at time t such that:

$$L_t^i = \begin{cases} 1 & i = l \\ 0 & \text{otherwise} \end{cases} \tag{8.4}$$

where L_t^i is the i^{th} element of vector $\overline{L_t}$. The binary valued $\overline{L_t}$ vector can be obtained from phonetic segmentation/recognition. Replacing $\overline{C_t}$ in (8.1) with $\overline{L_t}$, we have:

$$KL(\overline{C_t}, \overline{S_t}) = \sum_i L_t^i \, log_2 \frac{L_t^i}{p(q_t^i|x_t)} \tag{8.5}$$

$$KL(\overline{C_t}, \overline{S_t}) = \begin{cases} -log_2 \, p(q_t^l|x_t) & i = l \\ 0 & \text{otherwise} \end{cases} \tag{8.6}$$

hence,

$$KL(\overline{C_t}, \overline{S_t}) = -log_2 \, p(q_t^i|x_t) \qquad b_l \leq t \leq e_l \tag{8.7}$$

which is logarithm of the regular posterior for phone l in the interval $b_l \leq t \leq e_l$. Therefore, using binary labels in the deviation measure is similar to picking the posterior value belonging to the current phonetic segmentation. The NPCMs are obtained by accumulating these segment based phone posteriors followed by a normalization. Replacing 8.7 in 8.3, we have:

$$frame - basedNPCM(w) = -\frac{1}{\sum_{l=1}^{L}(e_l - b_l + 1)} \sum_{l=1}^{L} \sum_{t=b_l}^{e_l} KL(\overline{C_t}, \overline{S_t}) \tag{8.8}$$

which is the deviation measure averaged (normalized) by the length of the word hypothesis. Therefore, the NPCM measure is a special case of our deviation based measure where the

enhanced posterior term is replaced by phonetic segmentation obtained from the recognizer output. In the other words, the deviation measure can be viewed as a general case of NPCM measures where the binary phone labels are replaced by more smooth enhanced posteriors.

Conceptually, phone labeling, i.e. assigning a frame to an individual phone is not always correct as the phone evidences are highly correlated with the other phones in the context (specially at the phone segment boundaries). In practice also the phone segments are very short and are not precisely obtained (usually with a shift). The performance of PCMs can be highly affected by the imperfect phone segmentation[1]. In contrast, our approach uses the enhanced phone posteriors instead of phone labels (binary valued vectors). The enhanced posteriors are more smooth and still far away from a binary phone label. They also take into account the influence of neighboring phones thus they are more smooth at phone boundary transitions.

8.2 Summary and Conclusions

In this chapter, the difference (deviation) between regular and enhanced posteriors was studied. The proposed deviation measure can be used to detect inconsistency between the data and the model at different levels. One potential application of this deviation measure is detecting OOV words. The measure is based on comparison of two phone posterior streams derived from the identical acoustic evidence while using two different sets of prior constraints. The first stream is obtained using an MLP. The second stream is obtained by enriching the MLP phone posteriors using prior lexical knowledge. The comparison is based on measuring KL divergence (deviation) between the two posterior distributions. In contrast to the PCM measures used for OOV detection, the new approach does not require recognition results and phonetic segmentation, thus not affected by imperfect segmentation. It was also shown that the PCM measures are special case of the deviation based

[1] A phone segment can be only few frames long (e.g. 3-5), therefore a shift in segmentation even as small as few frames can dramatically affect NPCMs.

measure where the smooth enhanced posterior term is replaced with phone labels obtained from phonetic segmentation.

The experiments presented in this chapter were initial investigations towards the use of deviation based method in OOV detection. Using more competitive OOV detection baseline systems, and investigating the method for large vocabulary databases is necessary. The presented deviation based method was further studied in a summer workshop at Johns Hopkins University (July-August 2008) [109] for OOV word detection in large vocabulary databases. Using a similar method, they could show noticeable improvement in detecting OOV words over Wall Street Journal (WSJ) database, as compared to the use of regular confidence measures [8, 50, 51]. They showed that this approach is also suitable for detection of general recognition errors. For more details please refer to [109].

Chapter 9

Summary and Conclusions

In this thesis, we initially discussed the approaches using local posterior probabilities as local measures or as features in ASR systems. Indeed, several approaches in this direction have recently been shown to have a considerable potential to improve state-of-the-art ASR systems. However, we also believe that further progress in this direction will highly depend on improving these posterior estimates. Considering this issue, in this thesis we have presented a principled framework for enhancing the estimation of local posteriors (from the state up to the phone and word levels) by integrating long temporal context, as well as phonetic and lexical knowledge. We proposed and discussed two approaches for integrating long context and phonetic/lexical knowledge:

- **HMM-based enhanced posterior estimation:** The first approach uses a HMM module to integrate prior phonetic/lexical and contextual knowledge. The prior knowledge is encoded in the topology of the HMM module. The regular posteriors are used in HMM forward-backward recursions to integrate prior and contextual knowledge, yielding enhanced state/phone posterior estimates.
- **MLP-based enhanced posterior estimation:** In the second approach, a secondary MLP is used to post-process a temporal context of regular phone posteriors, and learn

long term dependencies between these posteriors. These long term dependencies are phonetic knowledge. During the inference (forward pass of the MLP), the learned knowledge is integrated in the phone posterior estimation. This results in enhanced phone posteriors at the output of the second MLP, as compared to the regular posteriors at the output of the first MLP.

In the HMM-based enhanced posterior estimation, the phonetic/lexical knowledge is explicitly provided by the prior assumptions about duration of phones and lexical use of phones in the words. In the MLP-based enhanced posterior estimation, the phonetic knowledge is learned from data. This difference leads to some dissimilarities in the way they are used in ASR systems.

Comparison of enhanced and regular posteriors showed that the enhanced posteriors perform better for frame level phone classification. Word recognition comparisons are reviewed in the following. In addition, studying the entropy of posteriors showed that the enhanced posteriors are less noisy as compared to the regular posteriors. All the experiments are performed using different small and large vocabulary databases.

9.1 Enhanced Phone Posteriors in ASR

The enhanced local phone posteriors can be used in a wide range of frame synchronous posterior based ASR application, such as hybrid HMM/ANN and Tandem systems. We proposed to replace or complement the use of local posterior probabilities by the new enhanced estimates of these local posteriors. We have investigated the use of enhanced local phone posteriors in three main directions:

- **Enhanced posteriors as features:** The use of enhanced posteriors as features for a standard HMM/GMM module (similar to Tandem) was investigated, and compared with the regular MLP posteriors:

1. HMM-based enhanced posteriors should be combined with the regular MLP posteriors to improve the word recognition performance in Tandem. In this case, the system using HMM-based enhanced posteriors as complementary features outperforms (in word recognition) the system using regular posteriors.

2. MLP-based enhanced posteriors can be used instead of the regular posteriors as features. The MLP-based enhanced posteriors outperform regular posteriors for word recognition in Tandem system.

- **Enhanced posteriors for decoding:** The enhanced posteriors were used as replacement to the regular posteriors for decoding in a hybrid HMM/ANN ASR system:

  1. HMM-based enhanced posteriors perform the same as regular posteriors in decoding. However, the decoder based on these enhanced posteriors is more robust against ad-hoc tuning parameters such as phone and word insertion penalties.

  2. MLP-based enhanced posteriors perform noticeably better for word and phone recognition, as compared to the regular posteriors.

- **Enhanced posteriors in confidence measurement:** We also investigated the use of enhanced posteriors as a replacement to the regular posteriors in confidence measurement. The confidence measures are based on accumulating local posteriors within a phone or word hypothesis. We have shown that using more informative enhanced posteriors results in more reliable confidence measures, as compared to the use of regular posteriors.

The idea of HMM-based enhanced posterior estimation was also extended to the case of multi-stream HMMs. Based on this extension, the enhanced posteriors are estimated by combining two complementary streams of features, as well as taking into account prior and contextual knowledge. The multi-stream enhanced posteriors were used as features in Tandem configuration. They outperform the single stream features, as well as the inverse entropy combination strategy (in word recognition).

We also studied the difference (deviation) between the regular and enhanced phone posteriors. The deviation measure between the two posterior streams can provide an indication of match/mismatch between data and prior knowledge at different levels. This measure was used for detecting OOV words (lexical mismatch) in posterior based ASR.

9.2 Local Word Posterior Estimation

The framework of HMM-based enhanced posterior estimation was also investigated for estimating local word posteriors. This issue was studied through the keyword spotting problem. We proposed estimating a local (frame level) posterior score for keyword and garbage units. Comparison of local keyword and garbage posteriors provides a frame level vote (decision) about the detection of the keyword. The local votes are accumulated (by counting) to decide about detection of the keyword in the utterance. As opposed to Viterbi decoding approach, a strong outlier can only affect few frame level votes, while the global likelihood score in the Viterbi approach can be highly affected by the outliers. Based on the initial experiments, the new posterior based scoring approach performs better in terms of trade-off between true and false spotting alarms, as compared to the traditional Viterbi decoding approach. The new approach also provides the possibility of precalculating keyword-specific spotting thresholds based on the length of the keyword.

9.3 Future Research Directions

- The HMM-based enhancement approach integrates the prior phonetic/lexical knowledge obtained from prior assumptions, while the MLP-based approach learns the knowledge from data. These two types of knowledge can be complementary in many cases, therefore the combination of the two approaches can take the advantage of complementary information. Since both approaches take local posteriors as input, and output local posteriors, the combination is feasible in a straight forward way. For

instance, the HMM-based approach can be used to integrate prior phone duration information, followed by the MLP-based approach to adapt the integration with the actual samples of an existing database.

- In the MLP-based enhanced posterior estimation, the second MLP has been trained on the same database as the first MLP. An alternative can be using a secondary database for training the second MLP. In this case, the second MLP can act as an adaptation module. The first MLP can be seen as a general purpose local phone posterior estimator, while the second MLP adapts the initial posterior estimates for a specific task or condition.

- In the MLP-based posterior enhancement, the strategies for optimizing the structure of the second ANN should be further studied. This can provide the possibility of processing longer temporal context. Phone posteriors have simpler and possibly linearly separable patterns, as compared to the acoustic features. Therefore, it is potentially possible to use a relatively simpler ANN for post-processing the posteriors.

- In this thesis, the estimation of local word posteriors was studied through the case of keyword spotting. However, local word posterior scores can also be used for a general case of speech recognition. At every frame, local posteriors of all the words in the vocabulary are estimated. These local word posteriors (already integrating lexical knowledge) can then be used in different ways to decode the word sequence. The simplest case is to use the word posteriors in a Viterbi decoder along with the grammatical knowledge. As compared to the traditional decoding approach which accumulates local phone likelihoods, the new configuration can be more robust against temporal outliers.

Bibliography

[1] H. Bourlard and N. Morgan, "Connectionist Speech Recognition – A Hybrid Approach," *Kluwer Academic Publishers*, 1994.

[2] H. Hermansky, D.P.W. Ellis and S. Sharma, "Connectionist Feature Extraction for Conventional HMM Systems," *Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing*, pp. 1635-1638, Istanbul, Turkey, 2000.

[3] D. Povey, G. Saon, L. Mangu, B. Kingsbury, and G. Zweig, "EARS Progress Update: Improved MPE, Inline Lattice Rescoring, Fast decoding, Gaussianization and Fisher Experiments," *EARS STT Workshop*, St. Thomas, US Virgin Islands, Dec. 2003.

[4] H. Soltau, B. Kingsbury, L. Mangu, D. Povey, G. Saon, G. Zweig, "The IBM 2004 Conversational Telephony System for Rich Transcription," *Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing*, vol. 1, pp. 205- 208, Philadelphia, USA, 2005.

[5] J. Bilmes, "Maximal Mutual Information Based Reduction Strategies for Cross-correlation based Joint Distribution Modelling," *Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing*, pp. 469-472, Seattle, 1998.

[6] H. H. Yang, S. Sharma, and H. Hermansky, "Relevance of Time Frequency Features for Phonetic and Spectral/Channel Classification," *Speech Communiations*, vol. 31, no. 1, pp. 35-50, Aug. 2000.

[7] L. R. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257-286, 1989.

[8] F. Wessel, R. Schlueter, K. Macherey, and H. Ney, "Confidence Measures for Large Vocabulary Continuous Speech Recognition," *IEEE Trans. Speech and Audio Processing*, vol. 9, no. 3, pp. 288-298, March 2001.

[9] F. Wessel, K. Macherey, and R. Schlueter, "Using Word Probabilities as Confidence Measures," *Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing*, pp. 225-228, USA, May 1998.

[10] A. O. Hatch, "Word-level Confidence Estimation for Automatic Speech Recognition", *M.S. Thesis*, ICSI Technical Report, Berkeley, April 2002.

[11] A. J. Viterbi, "Error Bounds for Convolutional Codes and an Asymptotically Optimal Decoding Algorithm," *IEEE Trans. Inform. Theory*, vol. 13, no. 2, pp. 260-269, 1967.

[12] F. Jelinek, "Continuous Speech Recognition by Statistical Methods," *Proc. IEEE*, vol. 64, no. 4, pp. 532-556, 1976.

[13] L. R. Bahl, F. Jelinek, and R. L. Mercer, "A Maximum-likelihood Approach to Continuous Speech Recognition," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 5, no. 2, pp. 179-190, 1983.

[14] F. Jelinek, "Statistical Methods for Speech," *Proc. Language, Speech and Communication Series*, MIT Press, Cambridge, MA, 1997.

[15] J. K. Baker, "The DRAGON System-An overview," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 23, no. 1, pp. 24-29, 1975.

[16] S. E. Levinson, "Structural Methods in Automatic Speech Recognition," *Proc. IEEE*, vol. 73, no. 11, pp. 1625-1650, 1985.

[17] P. Clarkson and R. Rosenfeld, "Statistical Language Modelling Using the CMU-Cambridge Toolkit," *In Proceedings of European Conference on Speech Communication and Technology*, pp. 2707-2710, Rhodes, Greece, 1997.

[18] A. Nadas, "Estimation of the Probabilities in the Language Model of the IBM Speech Recognition System," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 32, no. 4, pp. 859-861, 1984.

[19] J. Makhoul, "Linear Prediction: A Tutorial Review," *Proc. IEEE*, vol. 63, no. 4, pp. 561-580, 1975.

[20] A. V. Oppenheim, R. W. Schafer, "Digital Signal Processing," *Prentice Hall*, Englewood Cliffs, New Jersey, 1975.

[21] S. Davis, and P. Mermelstein, "Comparison of Parametric Representations for Monosyllabic Word Recognition in Countinuously Spoken Sentences," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 28, no. 4, pp. 357-366, 1980.

[22] H. Hermansky, "Perceptual Linear Predictive (PLP) Analysis of Speech," *Journal of the Acoustical Society of America*, vol. 87, no. 4, pp. 1738-1752, 1990.

[23] B. C. J. Moore, "An Introduction to the Psychology of Hearing," 4th Edition, *Academic Press*, San Diego, 1997.

[24] B. C. J. Moore, "Hearing," *Academic Press*, San Diego, 1995.

[25] S. S. Stevens, "On the Psychophysical Law," Pyschol. Rev. 64, pp. 153-181, 1957.

[26] J. D. Markel and A. H. Gray, "Linear Prediction of Speech," *Springer-Verlag*, New York, 1976.

[27] H. Hermansky, "TRAP-TANDEM: Data-driven Extraction of Temporal Features from Speech," *Proceedings of the IEEE Automatic Speech Recognition and Understanding (ASRU) Workshop*, pp. 255-260, USA, 2003.

[28] S. Furui, "Cepstral Analysis Technique for Automatic Speaker Verification," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 29, no. 2, pp. 254-272, 1981.

[29] S. Furui, "Speaker Independent Isolated Word Recognition Using Dynamic Features of Speech Spectrum," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 34, no. 1 , pp. 254-272, 1986.

[30] A. Andreou, T. Kamm, and J. Cohen, "Experiments in Vocal Tract Normalization," *In Proceedings of the CAIP Workshop: Frontiers in Speech Recognition II*, 1994.

[31] L. Lee and R. Rose, "Speaker Normalization Using Efficient Frequency Warping Procedures," *In Proceedings of IEEE Int. Conf. Acoust., Speech, and Signal Processing*, pp. 353-356, Atlanta, USA, 1996.

[32] B. S. Atal, "Automatic Recognition of Speakers from Their Voices," *Proc. IEEE*, vol. 64, no. 4, pp. 460-475, 1976.

[33] P. Jain and H. Hermansky, "Improved Mean and Variance Normalization for Robust Speech Recognition," *In Proceedings of IEEE International Conference on Acoustic, Speech, and Signal Processing*, 2001.

[34] S. Molau, F. Hilger, and H. Ney, "Feature Space Normalization in Adverse Acoustic Conditions," *In Proceedings of IEEE International Conference on Acoustic, Speech, and Signal Processing*, pp. 656-659, Hong Kong, 2003.

[35] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum Likelihood from Incomplete Data via the EM Algorithm," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 39, no. 1, pp. 1-38, 1977.

[36] L. E. Baum, T. Petrie, G. Souled, and N. Weiss, "A Maximization Technique Occurring in the Statistical Analysis of Probabilistic Functions of Markov Chains," *Ann. Math. Statist.*, vol. 41, no. 1, pp. 164-171, 1970.

[37] N. Merhav and Y. Ephraim, "Hidden Markov Modeling Using a Dominant State Sequence with Application to Speech Recognition," *Computer, Speech and Language*, vol. 5, no. 4, pp. 327-339, 1991.

[38] B. Gold and N. Morgan, "Speech and Audio Signal Processing: Processing and Perception of Speech and Music," *John Wiley and Sons*, 2000.

[39] G. D. Forney, "The Viterbi Algorithm," *Proceedings of the IEEE*, vol. 61, no. 3, pp. 268-278, 1973.

[40] L. R. Rabiner and H. W. Juang, "Fundamentals of Speech Recognition," *Prentice Hall*, Englewood Cliffs, New Jersey, 1993.

[41] J. A. Bilmes, "A Gentle Tutorial of the EM Algorithm and its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models," *ICSI-TR-97-021*, International Computer Science Institute, University of California at Berkeley, 1998.

[42] R. Schwartz, Y. Chow, O. Kimball, S. Roucos, M. Krasner, and J. Makhoul, "Context-dependent Modeling for Acoustic-phonetic Recognition of Continuous Speech," *In Proceedings of IEEE Int. Conf. Acoust., Speech, and Signal Processing*, pp. 1205-1208, Tampa, USA, 1985.

[43] K. F. Lee, "Context-dependent Phonetic Hidden Markov Models for Speaker-independent Continuous Speech Recognition," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 38, no. 4, pp. 599-609, 1990.

[44] S. J. Young, "The General Use of Tying in Phoneme-based HMM Speech Recognisers," *In Proceedings of IEEE Int. Conf. Acoust., Speech, and Signal Processing*, pp. 569-572, San Francisco, CA, 1992.

[45] A. Ljolje, "High Accuracy Phone Recognition Using Context-clustering and Quasitriphonic Models," *Computer, Speech and Language*, vol. 8, pp. 129-151, 1994.

[46] J. G. Fiscus, "A Post-Processing System to Yield Reduced Word Error Rates: Recogniser Output Voting Error Reduction (ROVER)", *In Proc. IEEE ASRU Workshop*, pp. 347-352, Santa Barbara, 1997.

[47] L. Mangu, E. Brill, and A. Stolcke, "Finding Consensus Among Words: Lattice-Based Word Error Minimization", *In Proc. Eurospeech'99*, pp. 495-498, Budapest.

[48] D. Povey and P. C. Woodland, "Minimum Phone Error and I-Smoothing for Improved Discriminative Training", *Proc. IEEE Intl. Conf. on Acoustic, Speech and Signal Processing*, vol. 1, pp. 105-108, Orlando, USA, 2002.

[49] Y. Abdel-Haleem, "Conditional Random Fields for Continuous Speech Recognition," *PhD Thesis*, University of Sheffield, November 2006.

[50] G. Bernardis and H. Bourlard, "Improving Posterior Confidence Measures in Hybrid HMM/ANN Speech Recognition System," *Proceedings of the Intl. Conference on Spoken Language Processing*, pp. 775-778, Sydney, Australia, 1998.

[51] G. Williams and S. Renals, "Confidence Measures for Hybrid HMM/ANN Speech Recognition," *Proceedings of Eurospeech'97*, pp. 1955-1958, Greece, 1997.

[52] R. Rojas, "Neural Networks - A Systematic Introduction", *Springer-Verlag*, Berlin, New-York, 1996.

[53] H. Cruse, "Neural Networks as Cybernetic Systems," 2nd revised edition, ebook, 1996.

[54] T. Kohonen, "Self-Organizing Maps," *Springer Series in Information Sciences*, vol. 30, 2001.

[55] S. Haykin, "Neural networks - A Comprehensive Foundation", 2nd ed., *Prentice-Hall*, NJ, USA, 1999.

[56] D. E. Rumelhart, G. E. Hinton, and R. T. Williams, "Learning Internal Representations by Error Propagation," In D. E. Rumelhart, J. L. McClelland, and the PDP research group, editors, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, vol. 1, pp. 318-362, MIT press, 1988.

[57] J. A. Snyman, "Practical Mathematical Optimization: An Introduction to Basic Optimization Theory and Classical and New Gradient-Based Algorithms," *Springer Publishing*, 2005.

[58] M. Degroot, "Probability and Statistics", 2nd ed., *Addison-Wesley*, 1980.

[59] M. D. Richard and R. P. Lippmann, "Neural network Classifiers Estimate Bayesian a Posteriori Probabilities," *Neural Computation*, no. 3, pp. 461-483, 1991.

[60] H. Gish, "A Probabilistic Approach to the Understanding and Training of Neural Network Classifiers," *in Proc. IEEE Intl. Conf. on Acoustic, Speech and Signal Processing*, pp. 1361-1364, 1990.

[61] H. Bourlard and N. Morgan, "Hybrid HMM/ANN Systems for Speech Recognition: Overview and New Research Directions", *Lecture Notes in Artificial Intelligence (1387)*, pp. 389-417, Springer Verlag, 1998.

[62] J. Hennebert, C. Ris, H. Bourlard, S. Renals, and N. Morgan, "Estimation of Global Posteriors and Forward-Backward Training of Hybrid HMM/ANN Systems," *Proceedings of EUROSPEECH'97*, pp. 1951-1954, Rhodes, Greece, 1997.

[63] N. Morgan and H. Bourlard, "Continuous Speech Recognition: An Introduction to the Hybrid HMM/connectionist Approach," *IEEE Signal Processing Magazine*, pp. 25-42, 1995.

[64] Z. Rivlin, M. Cohen, V. Abrash, Th. Chung, "A Phone Dependent Confidence Measure for Utterance Rejection," *Proc. IEEE Intl. Conf. on Acoustic, Speech and Signal Processing*, pp. 515-518, Atlanta, USA, 1996.

[65] R. C. Rose and D. B. Paul, "A Hidden Markov Model Based Keyword Recognition System," *Proc. IEEE Intl. Conf. on Acoustic, Speech and Signal Processing*, pp. 129-132, USA, 1990.

[66] G. William and S. Renals, "Confidence Measures from Local Posterior Probability Estimates", *Computer, Speech and Language*, vol. 13, pp. 395-411, 1999.

[67] L. Mangu, E. Brill, and A. Stolcke, "Finding Consensus in Speech recognition: Word Error Minimization and other Applications of Confusion Networks", *Computer, Speech and Language*, vol. 14, pp. 373-400, 2000.

[68] S. J. Young, D. Kershaw, J. J. Odell, D. Ollason, V. Valtchev, and P. C. Woodland, "The HTK Book (for HTK version 2.2)," Entropic Ltd., Cambridge, England, 1999.

[69] H. Hermansky and S. Sharma, "TRAPS Classifiers of Temporal Patterns," *Proceedings of Intl. Conf. on Spoken Language Processing*, Sydney, Australia, 1998.

[70] B. Chen, Q. Zhu, and N. Morgan, "Learning Long-term Temporal Features in LVCSR Using Neural Networks," *Proc. Interspeech'04*, pp. 925-928, Korea, October 2004.

[71] H. Hermansky and F. Fousek, "Multi-Resolution RASTA Filtering for TANDEM-Based ASR," *Proc. Interspeech'05*, pp. 361-364, Lisbon, Portugal, September 2005.

[72] Q. Zhu, B. Chen, N. Morgan, and A. Stolcke, "On Using MLP Features in LVCSR," *Proc. Interspeech'04*, pp. 921-924, Korea, October 2004.

[73] S. Ikbal, H. Misra, S. Sivadas, H. Hermansky, and H. Bourlard, "Entropy Based Combination of Tandem Representations for Robust Speech Recognition," *Proc. Interspeech'04*, Korea, October 2004.

[74] L. R. Rabiner, "A Tutorial on Hidden Markov Models and Selective Applications in Speech Recognition," *Proc. IEEE*, vol. 77, pp. 257-286, 1989.

[75] H. Ketabdar, and H. Bourlard, "Enhanced Phone Posteriors for Improving Speech Recognition Systems," *IDIAP Research Report 08-39*, 2008.

[76] J. Pinto, B. Yegnanarayana, H. Hermansky, and M. Doss, "Exploiting Contextual Information for Improved Phoneme Recognition," *in IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, Las Vegas, USA, 2008.

[77] M. Lehtonen, P. Fousek, and H. Hermansky, "Hierarchical Approach For Spotting Keywords", *IDIAP Research Report 05-41*, 2005.

[78] R. Cole, M. Fanty, M. Noel, and T. Lander, "Telephone Speech Corpus Development at CSLU," *In Proc. of International Conference on Spoken Language Processing*, pp. 1815-1818, Japan, 1994.

[79] R. Cole, M. Noel, T. Lander, and T. Durham, "New Telephone Speech Corpora at CSLU," *In Proc. of Eurospeech'05*, pp. 821-824, Spain, 1995.

[80] J. Godfrey, E. Holliman, and J. McDaniel. "SWITCHBOARD: Telephone Speech Corpus for Research and Development," *In Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, pp. 517-520, San Francisco, USA, 1992.

[81] P. Zhan, A. Waibel, "Vocal Tract Length Normalization for Large Vocabulary Continuous Speech Recognition," *Language Technologies Institute Technical Report: CMU-LTI-97-150*, Carnegie Mellon University, Pittsburgh, USA.

[82] Bisani, M., Ney, H., "Bootstrap Estimate for Confidence Intervals in ASR Performance Evaluation," *In Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, vol. 1, pp. 409-412, 2004.

[83] L. R. Bahl, R. Bakis, F. Jelinek, and R. L. Mercer, "Language-model/acoustic-channel-model Balance Mechanism," *IBM Technical Disclosure Bulletin*, vol. 23, no. 7b, pp. 3464-3465, 1980.

[84] K. Takeda, A. Ogawa, and F. Itakura, "Estimating Entropy of Language from Optimal Word Insertion Penalty," *In Proceedings of Int. Conf. Spoken Language Processing*, Australia, 1998.

[85] S. Renals and M. Hochberg, "Efficient Search Using Posterior Phone Probability Estimates," *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, vol. 1, pp. 596-599, Detroit, USA, 1995.

[86] D. Moore, J. Dines, M. Doss, J. Vepa, O. Cheng, and T. Hain, "Juicer: A Weighted Finite-State Transducer Speech Decoder," *3rd Joint Workshop on Multimodal Interaction and Related Machine Learning Algorithms*, USA, 2006.

[87] W. M. Fisher, G. R. Doddingtion, and K. M. Goudie-Marshall, "The DARPA Speech Recognition Research Database: Specifications and Status," *Proc. of DARPA Workshop on Speech Recognition*, pp. 93-99, Feb. 1986.

[88] R. A. Sukkar and J. G. Wilpon, "A Two Pass Classifier Utterance Rejection in Keyword Spotting," *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, vol. 2, pp. 451-454, 1993.

[89] Sh. R. Young, "Detecting Misrecognition and Out-of-Vocabulary Words," *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, vol. 2, pp. 21-24, Australia, 1994.

[90] M. Weintraub, F. Beaufays, Z. Rivlin, Y. Konig, and A. Stolcke, "Neural Network Based Measures of Confidence for Word Recognition," *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, vol. 2, pp. 887-890, Munich, Germany, 1997.

[91] G. Williams and S. Renals, "Confidence Measures for Evaluating Pronunciation Models," *In ESCA Workshop on Modeling Pronunciation Variation for Automatic Speech Recognition*, pp. 151-155, Kerkrade, Netherlands, 1998.

[92] J. Fransen, D. Pye, T. Robinson, P. Woodland, and S. Young, "WSJCAM0 Corpus and Recording Description," *Technical Report 192*, Cambridge University Engineering Department, 1994.

[93] H. Bourlard and S. Dupont, "Sub-band-based Speech Recognition," *Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing*, pp. 1251-1254, Germany, 1997.

[94] S. Dupont and J. Luettin, "Using the Multi-stream Approach for Continuous Audio-visual Speech Recognition: Experiments on the M2VTS database," *In Proceedings of International Conference on Spoken Language Processing*, pp. 1283-1286, Australia, 1998.

[95] S. Dupont and J. Luettin, "Audio-visual Speech Modeling for Continuous Speech Recognition," *IEEE Transactions on Multimedia*, vol. 2, no. 3, pp. 141-151, 2000.

[96] K. Kirchhoff, "Combining Articulatory and Acoustic Information for Speech Recognition in Noisy and Reverberation Environments," *Proc. Int. Conf. on Spoken Language*, pp. 891-894, 1998.

[97] H. Misra, H. Bourlard, and V. Tyagi, "New Entropy Based Combination Rules in HMM/ANN Multi-stream ASR," *Proc. Intl. Conf. on Acoustics, Speech and Signal Processing*, vol. 2, pp. 741-744, China, 2003.

[98] S. Bengio, "Joint Training of Multi-stream HMMs," *IDIAP-RR 05-22*, 2005.

[99] J. G. Wilpon, L. R. Rabiner, C. H. Lee, and E. R. Goldman, "Automatic Recognition of Keywords in Unconstrained Speech Using Hidden Markov Models," *IEEE Trans. ASSP*, vol. 38, no. 11, pp. 1870-1878, 1990.

[100] G. Wilpon, L. G. Miller, and P. Modi, "Improvements and Applications for Key Word Recognition Using Hidden Markov Modeling Techniques," *Proc. Intl. Conf. on Acoustics, Speech and Signal Processing*, pp. 309-312, Toronto, Canada, 1991.

[101] R. Rohlicek, W. Russell, S. Roukos, and H. Gish, "Continuous Hidden Markov Modeling for Speaker-Independent Word Spotting," *Proc. Intl. Conf. on Acoustics, Speech and Signal Processing*, pp. 627-630, Glasgow, UK, 1989.

[102] L. D. Wilcox and M. A. Bush, "Training and Search Algorithms for an Interactive Word Spotting System," *Proc. Intl. Conf. on Acoustics, Speech and Signal Processing*, vol. 2, pp. 97-100, 1992, San Francisco, USA.

[103] H. Bourlard, B. D'hoore, and J.M. Boite, "Optimizing recognition and rejection performance in word spotting systems," *In Proc. Intl. Conf. on Acoustics, Speech and Signal Processing*, vol. 1, pp. 373-376, 1994.

[104] H. Ketabdar, M. Hannemann, and H. Hermansky, "Detection of Out-of-Vocabulary Words in Posterior Based ASR," *Proc. Interspeech'07*, pp. 1757-1760, Belgium, 2007.

[105] L. Chase, "Error-Responsive Feed Back Mechanisms for Speech Recognizers," *PhD Thesis*, School of Computer Science, Carnegie Mellon University, Pittsburgh, USA, April 1997.

[106] T. Hazen and I. Bazzi, "A Comparison and Combination of Methods for OOV Word Detection and Word Confidence Scoring," *Proc. Intl. Conf. on Acoustics, Speech and Signal Processing*, pp. 397-400, Salte Lake City, Utah, 2001.

[107] T. Hazen, "Recognition Confidence Scoring for Use in Speech Understanding Systems," *Proc. of ISCA ASR2000 Tutorial and Research Workshop*, Paris, 2000.

[108] S. Kamppari, and T. Hazen, "Word and Phone Level Acoustic Confidence Scoring", *Proc. Intl. Conf. on Acoustics, Speech and Signal Processing*, pp. 1799-1802, Istanbul, 2000.

[109] L. Burget, P. Schwarz, P. Matejka, M. Hannemann, A. Rastrow, C. White, S. Khudanpur, H. Hermansky, and J. Cernocky, "Combination of Strongly and Weakly Constrained Recognizers for Reliable Detection of OOVs," *Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing*, pp. 4081-4084, Las Vegas, USA, 2008.

Curriculum Vitae

Hamed Ketabdar

Ch. de Fontenay 7b, 1007, Lausanne, Swtizerland http://people.epfl.ch/hamed.ketabdar
Phone : +41787378091, ketabdar.hamed@gmail.com

Education

Sep. 2004- Oct. 2008 (expected) | *Swiss Federal Institute of Technology (EPFL), Idiap* | *Lausanne, Switzerland*

Doctor of Science

- **Research:** Signal processing and machine learning, with application in Automatic Speech Recognition (ASR)
- **Advisor:** Prof. Herve Bourlard
- **Institute:** Idiap Research Institute (www.idiap.ch)
- **Thesis:** Enhancing Posterior Based Speech Recognition Systems

Oct. 2003- Aug. 2004 | *EPFL, Signal processing Institute* | *Lausanne, Switzerland*

Pre-doctoral School in Computer, Communication and Information Sciences

Project: Feature Selection for Signature Verification **Laboratory:** Signal Processing Institute (its.epfl.ch)
Research: Machine Learning, Signal Processing, Biometrics **Courses:** Image and Speech Processing, Machine Learning

Sep. 1999- Sep. 2003 | *Sharif University of Technology (www.sharif.ir)* | *Tehran, Iran*

Bachelor of Science in Electrical Engineering (Electronics)

- **Thesis:** Automatic Recognition of Handwritten Digits
- **GPA:** 17.06/20, Average university GPA: 12.96/20

Computer skills:

- **Computer languages and packages:** C/C++, Pascal, MATLAB, HTK (Hidden Markov Model toolkit), QuickNet (Neural Networks toolkit), Shell Scripting, Python, Assembly (various platforms)
- **Computer architectures and Electronics**: 2106x SHARC Analog Devices DSPs, MCS-51 Microcontrollers
- **Operating systems:** Windows and Linux.

Languages: English (fluent), French (intermediate), Persian (native)

Professional Experience

Sep 2004- Present | *Idiap Research Institute, EPFL* | *Martigny, Switzerland*

Research assistant

- Pursuing my PhD in signal processing and machine learning (statistical pattern recognition), mainly in Automatic Speech Recognition (ASR).
- Research on using artificial neural networks for acoustic modeling, developing approaches for improving the estimates of phone posterior probabilities in ASR.
- Working on large vocabulary speech processing for the purpose of recognition, keyword spotting, out-of-vocabulary word detection, and confidence measurement.
- Developing different practical software for decoding and training in HMMs and neural networks.
- Supervising student projects, laboratory assistant for a speech recognition course.

Jan. 2004- Aug. 2004 | *Signal Processing Institute, EPFL* | *Lausanne, Switzerland*

Research assistant

- Working on identity verification based on human signature verification.
- Developing algorithms and measures for selecting optimal set of features used in signature verification.
- Student paper award for a paper on signature verification.

Jan. 2003- Aug. 2003 | *KishLeadTech Co.* | *Tehran, Iran*

Research and development associate

- Developing algorithms and a practical software for automatic document processing and handwritten character recognition.
- Consulting on different products selection for telephony applications.
- Working on video phones and video transmission over very noisy channels.

May 2000- March 2002 | *Electronics Research Center, Sharif Univ. of Tech.* | *Tehran, Iran*

Research and development associate

- Working on real time implementation of speech coders on Digital Signal Processors (DSPs).
- Time and memory efficient low level programming of critical DSP routines.
- Study and comparison of different DSPs for speech coding (mainly Analog Devices and Texas Instruments families).

Reviewer for IEEE Signal Processing Letters

Honors and awards

- EPFL doctoral school fellowship, Oct. 2003- July 2004
- Graduate student paper award (2nd position) from the International Society of Motor Control for a paper on signature verification, Salerno, Italy , June 2005
- KishLeadTech company scholarship, Tehran, Iran, Jan. 2003 - Aug. 2003
- Entrance to Sharif University of Technology (the most outstanding technical university in Iran) in competition with about 300 000 high school students, Sep. 1999
- SPIE student travel grant, Sep. 1999
- Third prize of 11th International Kharazmi Festival (among 1862 projects) for designing a test grader machine, January 1998. Kharazmi Festival is the most important scientific festival in Iran.

Publications

Journal:

- Hamed Ketabdar and Herve Bourlard, ``Enhanced Phone Posteriors for Improving Speech Recognition Systems'', Under second revision for IEEE Transactions on Speech and Audio Processing.

- Hamed Ketabdar, Samy Bengio, and Herve Bourlard, ``Multi-stream Posterior Estimation for Posterior Based Speech Recognition Systems'', Manuscript under preparation for *Speech Communication.*

Book Chapter:

- Hamed Katebdar, Herve Bourlard and Samy Bengio, ``Hierarchical Multi-Stream Posterior Based Speech Recognition System'', Machine Learning for Multimodal Interaction (MLMI'05), Lecture Notes in Computer Science, vol. 3869, pp. 294-306, Springler-Verlag.

Selected Conference papers (peer reviewed):

- Hamed Ketabdar and Herve Bourlard, ``Hierarchical Integration of Phonetic and Lexical Knowledge in Phone Posterior Estimation", In *Proceedings of 2008 IEEE International Conference on Acoustics, Speech and Signal Processing* (ICASSP 2008), Las Vegas, USA .

- Hamed Ketabdar and Herve Bourlard, ``In-context Posteriors as Complementary Features for TANDEM ASR", In *Proceedings of the European Conference on Speech Communication and Technology* (Interspeech'07- ICSLP), Belgium, August 2007.

- Hamed Ketabdar and Hynek Hermansky, ``Detection of Out-of-Vocabulary Words in Posterior Based ASR'', In *Proceedings of the European Conference on Speech Communication and Technology* (Interspeech'07- ICSLP), Belgium, August 2007.

- Hamed Ketabdar, Jithendra Vepa, Samy Bengio and Hervé Bourlard, ``Using More Informative Posterior Probabilities for Speech Recognition'', In *Proceedings of 2006 IEEE International Conference on Acoustics, Speech and Signal Processing* (ICASSP 2006), Toulouse, France.

- Hamed Ketabdar, Jithendra Vepa, Samy Bengio, and Herve Bourlard, ``Developing and Enhancing Posterior Based Speech Recognition Systems'', In *Proceedings of the 9th European Conference on Speech Communication and Technology* (Interspeech'2005- Eurospeech), Lisbon, Portugal.

- Hamed Ketabdar, Jonas Richiardi, and Andrzej Drygajlo, ``Global Feature Selection for On-line Signature Verification'', *12th Conference of International Graphonomics Society*, pp. 59-63, Salerno, Italy, June 2005.

- Jonas Richiardi, Hamed Ketabdar, and Andrzej Drygajlo, ``Local and Global Feature Selection for Signature Verification'', In *Proceedings of 8th International Conference on Document Analysis and Recognition* (ICDAR'05), Seoul, Korea, 2005.

- Hamed Ketabdar, ``Handwritten Persian Digit Recognition by a Structural Method'', *Document Recognition and Retrieval VII, SPIE's Electronic Imaging*, 24-28 January 2000, San Jose, CA, USA.

Teaching Experience

Oct. 2006- Feb. 2007, EPFL
Mar. 2006- June 2006
Idiap Research Institute

- Teacher assistant for ``Automatic Speech Processing'' course
- Supervising a Bachelor student project on developing a software for pronunciation evaluation