Sparse and Low-rank Modeling for Automatic Speech Recognition

Thèse N° 9035

Présentée le 8 mars 2019

à la Faculté des sciences et techniques de l'ingénieur
Laboratoire de l'IDIAP
Programme doctoral en génie électrique

pour l'obtention du grade de Docteur ès Sciences

par

Pranay DIGHE

Acceptée sur proposition du jury

Prof. J.-Ph. Thiran, président du jury
Prof. H. Bourlard, directeur de thèse
Prof. F. Metze, rapporteur
Dr R. Schlüter, rapporteur
Prof. J.-M. Vesin, rapporteur

2019

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Abstract

This thesis deals with exploiting the low-dimensional multi-subspace structure of speech towards the goal of improving acoustic modeling for automatic speech recognition (ASR). Leveraging the parsimonious hierarchical nature of speech, we hypothesize that whenever a speech signal is measured in a high-dimensional feature space, the true class information is embedded in low-dimensional subspaces whereas noise is scattered as random high-dimensional erroneous estimations in the features. In this context, the contribution of this thesis is twofold: (i) identify sparse and low-rank modeling approaches as excellent tools for extracting the class-specific low-dimensional subspaces in speech features, and (ii) employ these tools under novel ASR frameworks to enrich the acoustic information present in the speech features towards the goal of improving ASR. Techniques developed in this thesis focus on deep neural network (DNN) based posterior features which, under the sparse and low-rank modeling approaches, unveil the underlying class-specific low-dimensional subspaces very elegantly.

In this thesis, we tackle ASR tasks of varying difficulty, ranging from isolated word recognition (IWR) and connected digit recognition (CDR) to large-vocabulary continuous speech recognition (LVCSR). For IWR and CDR, we propose a novel *Compressive Sensing* (CS) perspective towards ASR. Here exemplar-based speech recognition is posed as a problem of recovering sparse high-dimensional word representations from compressed low-dimensional phonetic representations. In the context of LVCSR, this thesis argues that albeit their power in representation learning, DNN based acoustic models still have room for improvement in exploiting the *union of low-dimensional subspaces* structure of speech data. Therefore, this thesis proposes to enhance DNN posteriors by projecting them onto the manifolds of the underlying classes using principal component analysis (PCA) or compressive sensing based dictionaries. Projected posteriors are shown to be more accurate training targets for learning better acoustic models, resulting in improved ASR performance. The proposed approach is evaluated on both close-talk and far-field conditions, confirming the importance of sparse and low-rank modeling of speech in building a robust ASR framework. Finally, the conclusions of this thesis are further consolidated by an information theoretic analysis approach which explicitly quantifies the contribution of proposed techniques in improving ASR.

Keywords: automatic speech recognition, deep neural network, sparsity, dictionary learning, low-rank, principal component analysis.

Résumé

Cette thèse traite de l'exploitation de la structure multi-sous-espaces de la parole en dimension réduite, où le but est d'améliorer la modélisation acoustique pour la reconnaissance automatique de la parole (RAP). En tirant parti de la nature hiérarchique parcimonieuse de la parole, nous émettons l'hypothèse que chaque fois qu'un signal de parole est mesuré dans un espace de grande dimension, les informations de classe réelles sont incorporées dans des sous-espaces de faible dimension, tandis que le bruit est éparpillé de façon aléatoire dans des estimations erronées de paramètres dans cet espace de grande dimension. Dans ce contexte, la contribution de cette thèse est double: (i) l'identification d'approches de modélisation clairsemées et de bas rang, qui s'avèrent être d'excellents outils pour extraire les sous-espaces de faible dimension spécifiques à la classe dans les fonctionnalités vocales, et (ii) l'utilisation de ces outils dans de nouveaux cadres de RAP pour enrichir l'information acoustique présente dans les caractéristiques vocales dans le but d'améliorer les performances de la RAP. Les techniques développées dans cette thèse se concentrent sur les caractéristiques postérieures basées sur les réseaux neuronaux profonds (DNN, pour deep neural network) qui, sous les approches de modélisation rares et de bas rang, dévoilent très élégamment les sous-espaces de basse dimension spécifiques aux classes.

Dans cette thèse, nous abordons des tâches de RAP de difficulté variable, allant de la reconnaissance de mots isolés (RMI) et de la reconnaissance de chiffres connectés (RCC) à la reconnaissance vocale continue à grand vocabulaire (RVCGV). Pour la RMI et la RCC, nous proposons une nouvelle perspective d'*aquisition comprimée* avec la RAP en vue. Dans ce cas, la reconnaissance de la parole basée sur l'exemple est posée comme un problème de récupération de représentations de mots de grande dimension éparses à partir de représentations phonétiques compressées de faible dimension. Dans le contexte de la RVCGV, cette thèse soutient que, malgré leurs capacités dans l'apprentissage par représentation, les modèles acoustiques basés sur les DNNs ont encore une certaine marge de manœuvre pour exploiter la structure de l'*union des sous-espaces de faible dimension* des données de parole. Par conséquent, cette thèse propose d'améliorer les postérieurs des DNNs en les projetant sur les variétés des classes sous-jacentes en utilisant des dictionnaires basés sur l'analyse en composantes principales (ACP) ou la détection par compression. Les postérieurs projetés se révèlent être des cibles d'entraînement plus précises pour apprendre de meilleurs modèles acoustiques, ce qui se traduit par une amélioration de la performance de la RAP. L'approche proposée est évaluée à la fois dans des conditions de conversation rapprochée et dans des conditions distantes, confirmant l'importance de la modélisation clairsemée et de faible rang de la parole dans la construction d'un système de RAP robuste. Enfin, les conclusions de cette thèse sont encore consolidées par une approche d'analyse théorique de l'information qui quantifie explicitement la contribution des techniques proposées dans l'amélioration de la RAP.

Mots clefs: reconnaissance automatique de la parole, réseau de neurones profond, rareté, apprentissage de dictionnaire, bas rang, analyse en composantes principales.

Acknowledgements

I express my foremost gratitude to my thesis supervisor Prof. Hervé Bourlard for providing me an internship opportunity 5 years ago at Idiap, which later turned into an exciting and rewarding research experience in form of a Ph.D. I thank Hervé for his guidance, insightful discussions, and ever motivating comments during the course of my Ph.D. He always promoted freedom to explore ideas, spend enough time to experiment on them and liberty to fail in this process. His remark that *"whenever something doesn't work, we should consider ourselves lucky as it's a new learning opportunity"* has been inspirational to me for my research as well as otherwise. A special thanks also goes to Dr. Afsaneh Asaei, who mentored me in the initial years of my Ph.D. Without her guidance, this thesis would not have been possible. I cherish our daily discussions on the white boards of Idiap.

I would also like to thank my thesis jury members: Prof. Florian Metze, Dr. Ralf Schlüter, Prof. Jean-Marc Vesin, and Prof. Jean-Phillipe Thiran for their insightful comments and encouragement.

I sincerely thank Swiss National Science Foundation (SNSF) for supporting my research through the grants, 200021-153507 and 200020-169398, for PHASER project. My gratitude also goes to Siri speech team at Apple Inc. for giving me an internship opportunity which was a great learning experience and broadened my research perspective. A special mention goes to Dr. Srikanth Vishnubhotla who mentored me during the internship and taught me how to be vigorous in research. I would also like to thank Prof. Harish Karnick and Prof. Bhiksha Raj who provided me wonderful research oppotunities before joining EPFL. I am thankful to have their continued support and guidance.

At Idiap, I would like to thank the secretariat and the system administration group for providing a very professional workspace and excellent human and technical resources. The most important part of work and life in Martigny have been the wonderful friends I made over the last 5 years with whom I shared countless beers, coffees, ski trips, vacations, sports, hikes, chai bullas, movies, dinners, etc. I thank Marc, PE, Cijo, GCC, James, Rui, Subhadeep, Dhananjay, Nicholas, Wudi, Tatjana, Angel, Skanda, Srikanth, Mathew, Raphael, Vinayak, Bastien, Sophie, Suraj, Banri, Apoorv, Lesly,Weiping, Sibo, Phil, Tiago and many many more friends whose names I am forgetting- for making Martigny and Switzerland a second home for me. A very special gratitude goes to the french teacher Michel for helping me integrate in the local culture of Valais. A special thanks to Dorothy whose constant support and motivation made everything look easier. A special mention goes to the SAM group and Martigny Cricket Club which made the last year at Idiap memorable. Idiap will be dearly missed.

Finally, last but by no means least, I am most grateful to my whole family, especially my parents for their patience, support, guidance, blessings, and freedom they provided me to pursue whatever interests me; and I am immensely thankful to my brother Prateek and his better-half Namrata for all their encouragement and love.

Thanks also to the cheesemakes of Valais!

Martigny, October 2018 Pranay

Contents

List of Figures

List of Tables

List of Algorithms

List of Acronyms

AM	acoustic model
ASR	automatic speech recognition
CDR	connected digit recognition
CE	cross entropy
CNN	convolutional neural network
CS	compressive sensing
DCT	discrete cosine transform
DNN	deep neural network
DTW	dynamic time warping
EM	expectation-maximization
FBANK	log-Mel filterbank energies
FFT	fast Fourier transform
fMLLR	feature-space maximum likelihood linear regression
GMM	Gaussian mixture model
HMM	hidden Markov model
IHM	individual headset microphone
IWR	isolated word recognition
JSEAM	joint speech enhancment acoustic modeling
KL	Kullback–Leibler
LARS	least angle regression
LASSO	least absolute shrinkage and selection operator
LDA	linear discriminant analysis
LM	language model
LSTM	long short-term memory network
LVCSR	large vocabulary continuous speech recognition
MFCC	Mel-frequency cepstral coefficients
MLLT	maximum likelihood linear transform
OMP	orthogonal matching pursuit
PC	principal component
PCA	principal component analysis
RIP	restricted isometric property
RNN	recurrent neural network
RPCA	robust principal component analysis
SDM	single distant microphone
SE	speech enhancement
sMBR	state minimum Bayes risk
SVD	singular value decomposition
TDNN	time-delay neural network
WER	word error rate

1 Introduction

Automatic speech recognition (ASR) is defined as the task of converting a speech signal into text using a computer. The availability of massive amounts of data and faster computational resources coupled with the advancements in *deep learning* [LeCun et al. (2015)] has recently resulted in the development of a wide range of ASR technologies and applications. High performance gains in large vocabulary continuous speech recognition (LVCSR) have made it possible to use ASR systems in commercial products like smartphones which are used by billions of people on a daily basis. Smart personal assistants (like Siri, Alexa, and Google assistant), speech-controlled smart-home devices, systems for transcribing conversations like meeting recordings, automatic subtitle generation for broadcast news or movies, military, and healthcare industry usages are some of the major applications of ASR.

The recent progress has resulted in an ever-increasing need for improving the existing technology to continually address the open challenges in the field of ASR. For example, the recognition of unconstrained conversational speech in noisy conditions, possibly corrupted with overlapping speech, is still a very challenging task and the performance of current state-of-the-art systems is far from reaching parity with human performance. Developing systems suitable for far-field reverberated speech is another major challenge and is an active area of research. It also remains difficult to recognize speech in unseen noise or mismatched conditions.

State-of-the-art hybrid ASR techniques typically employ deep neural networks (DNN) for estimating the probability distribution over speech data under a hidden Markov model (HMM) based back-end which is used for sequence modeling of speech. Although the hybrid DNN-HMM approach has undoubtedly advanced the field of ASR, there are certain fundamental properties of speech which are still not fully exploited by it. One such important property is the hierarchical parsimony in the structure of speech. Modeling speech by utilizing its inherent internal structure not only holds the key towards a robust and improved ASR framework but also towards a critical understanding of speech modeling for ASR in general. The research presented in the current thesis addresses this very quest by proposing novel applications of sparse and low-rank modeling approaches towards improving ASR. In this pursuit, we first develop sparse modeling based ASR solutions for relatively simpler tasks of isolated word recognition and connected digit recognition. Lessons learned from this research direction are then applied to tackle the harder problem of LVCSR for conversational speech in a meeting scenario where close-talk, as well as far-field microphone conditions, are considered. Along with empirical verification, the conclusions of this thesis are also consolidated by a novel information theoretic analysis approach which serves as a tool for measuring the contribution of sparse and low-rank modeling in improving ASR.

The rest of this chapter is organised as follows. Section 1.1 provides the general motivation behind the approaches proposed in this work. Section 1.4 summarizes the contributions of this thesis. Section 1.5 gives a chapter-wise outline for the rest of the thesis and finally, Section 1.6 lists the notations used in this thesis.

1.1 Motivations

Speech production is a *hierarchical* and *parsimonious* process by nature. Speech utterances are formed as a union of words which in turn consist of phonetic components and sub-phonetic attributes. Each linguistic component is produced through activation of a few highly constrained articulatory mechanisms leading to the generation of speech data in a union of low-dimensional subspaces [Deng (2004); King et al. (2007); Lee et al. (2001)]. While understanding the hierarchical nature of speech is straightforward (as visualized in Figure 1.1), the *parsimonious* nature of speech signifies that out of the many possibilities for a speech unit (e.g., phonemes or words), only one or very few possibilities are realized at any given instance. For example, only one word and its corresponding phonetic sequence will be spoken at a given time instance from the whole vocabulary.

The *hierarchy*[1] and *parsimony* in the structure of speech are interconnected to each other (refer Figure 1.1). Movements of the articulatory mechanisms are low-dimensional phenomena which lead to the production of phonetic and sub-phonetic sounds in a relatively higher dimensional space. Unique sequences of phonemes result in the realization of distinct words which are enormously more in numbers than phonetic sounds. Similarly, sentences formed by sequences of words live in an exponentially high-dimensional space whose dimensionality is limited only by the extent of the human imagination. As we move higher in the hierarchy, the space becomes increasingly high dimensional and sparse. Note that this parsimonious nature of speech is not dependent on our measurement of the speech signal. It is an intrinsic property which is an outcome of the way human languages are hierarchically structured.

An important consequence of the parsimonious hierarchical structure of speech is the existence of class-specific low-dimensional subspaces in speech features [Stevens (1998); Jansen and Niyogi (2006)]. For example, recent advancement in DNN based acoustic modeling relies

[1] Speech production mechanism is often expressed as a motor control system where a sensory feedback loop guides the production process. However, there is evidence [Hickok (2012)] from a psycholinguistic perspective that even a feedback-based speech production process is hierarchically organized across different levels which span phonetic, word and phrase-level units.

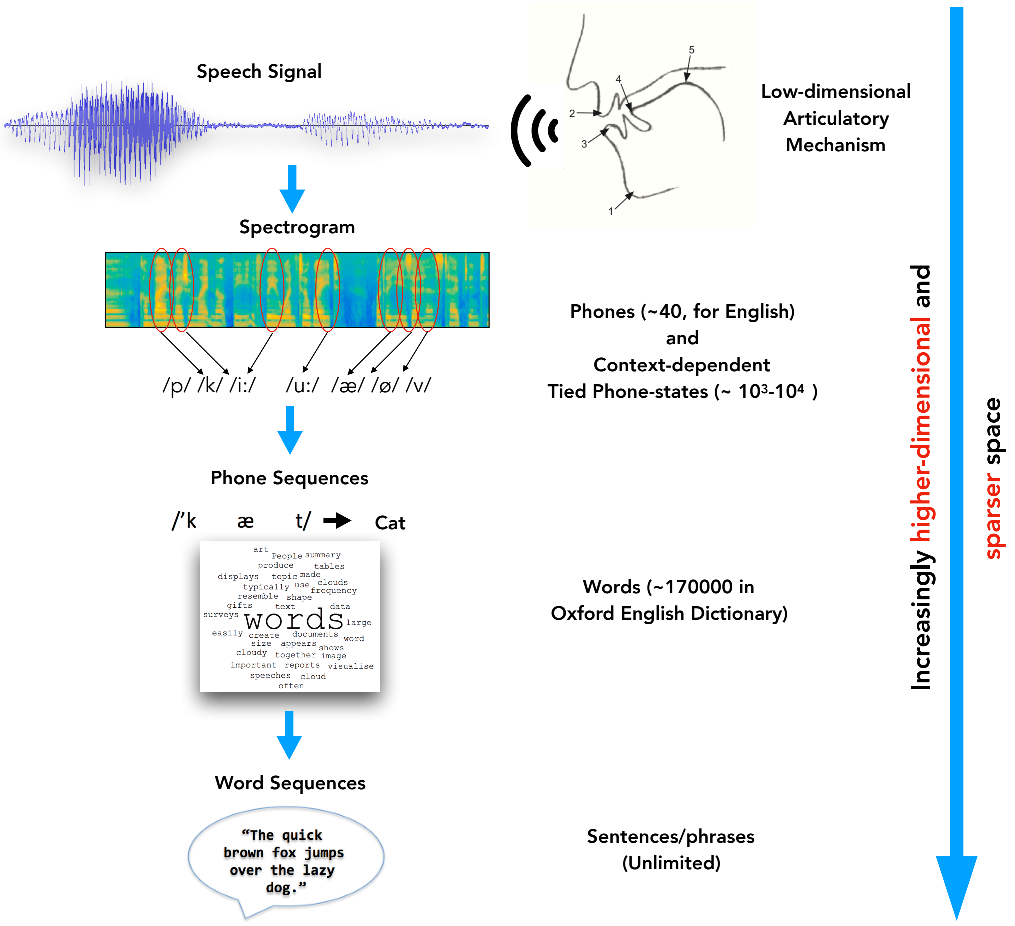
Speech Signal
Low-dimensional Articulatory Mechanism
1
2
3
4
5
Spectrogram
/p/ /k/ /i:/ /u:/ /æ/ /ø/ /v/
Phones (~40, for English) and Context-dependent Tied Phone-states (~ 10³-10⁴)
Phone Sequences
/'k æ t/ ➡ Cat
words
Words (~170000 in Oxford English Dictionary)
Word Sequences
"The quick brown fox jumps over the lazy dog."
Sentences/phrases (Unlimited)
Increasingly higher-dimensional and sparser space

Figure 1.1: Parsimonious hierarchical structure of human speech. Part of the image has been taken from [Lee et al. (2001)].

on the estimation of posterior probabilities of context-dependent subphone units. These probability vectors, termed as *posterior* features or simply DNN posteriors in this thesis, are typically very high dimensional and sparse. Only a few non-zero dimensions in the posterior features are significant as they form different low-dimensional subspaces which belong to factors like (1) underlying acoustic events such as a particular phonetic sound being realized or (2) unique variations in pronunciation which are characteristic of a specific speaker. The current thesis concerns specifically with the modeling of low-dimensional subspaces in DNN posteriors which are occupied by acoustic events like the realization of words, phonemes, and subphonetic components.

As argued in [Bengio (2009)], there are two ways of modeling a phenomenon whose underlying causative factors are low-dimensional- (1) as compressed **low-rank representations** for each factor's subspace individually, or (2) as a common high-dimensional **sparse representa-**

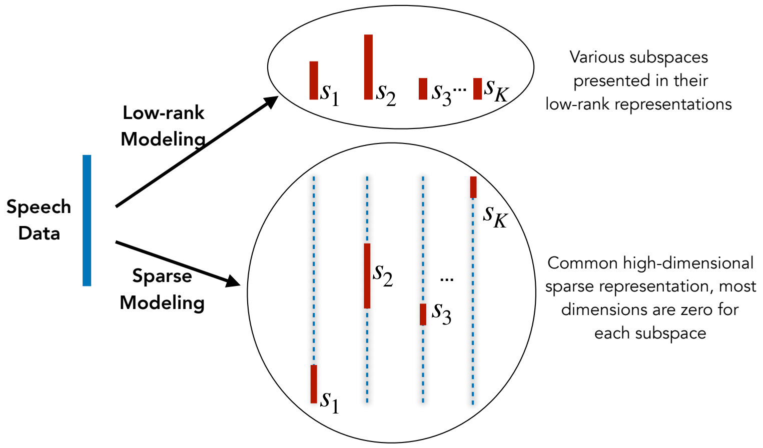
Low-rank
Modeling
Speech
Data
Sparse
Modeling
s_1
s_2
s_3 ...
s_K
Various subspaces presented in their low-rank representations
s_K
s_2
...
s_3
s_1
Common high-dimensional sparse representation, most dimensions are zero for each subspace

Figure 1.2: Modeling class-specific low-dimensional subspaces in speech data using low-rank v/s sparse modeling approach. s_k denotes k^{th} subspace in the data.

tion where all the factors reside together in different subspaces. Low-rank representations model the subspaces in a compressed manner by discarding irrelevant dimensions of the data, whereas sparse representations achieve the same goal by projecting data in a high-dimensional sparse space where the underlying structures are disentangled, and only the relevant dimensions are activated (non-zero). The application of low-rank and sparse representations for modeling speech forms the core of the research conducted in this thesis. In Table 1.1 and Figure 1.2, we contrast the low-rank and sparse modeling approach. Note that the low-rank modeling here refers to classwise low-rank models and not a common low-rank model for the whole multi-class data considered together.

Representation	Low-rank	Sparse
Information	Densely packed	Sparse, in a few non-zero dimensions
Underlying factors	Entangled	Disentangled
Structure	Classwise low-dimensional representation	Common high-dimensionsal representation for all classes
Approaches	Principal component analysis, Independent component analysis, etc.	Dictionary learning and sparse coding using ℓ_0 or ℓ_1-norm constraints

Table 1.1: Contrasting low-rank and sparse modeling approaches.

1.2 ASR as a Compressive Sensing Problem

One of the major approaches considered in this thesis is exemplar-based sparse representations for ASR, which has been explored previously in [Sainath et al. (2011); Gemmeke et al.

(2011, 2009); Sainath et al. (2010)]. The core assumption of this approach is that any possible realization of the data in the test set lies in a vector space spanned by a sparse selection of exemplars already seen in the training set. Thus, a large *collection-of-exemplars* is traditionally used in practice to capture all possible variability in the data. In this thesis, we pose the exemplar-based sparse representation approach for ASR as a *Compressive Sensing* (CS) problem. Instead of relying on the large collection-of-exemplars method, we pursue the goal of finding an over-complete set of basis vectors such that a sparse linear combination of these vectors can be used to generate all the points in the test data. The over-complete basis set termed as a *dictionary* in CS theory, has a much lower cardinality than the typical collection-of-exemplars set used in previous approaches. Moreover, sparse recovery algorithms, borrowed from CS literature, ensure that sparse linear combinations of the dictionary columns (termed as *atoms*) can still adequately span the variability in the space. Further, if the dictionaries are designed in a particular manner, our compressive sensing approach can be given a very intuitive probabilistic interpretation in terms of the ASR theory (as we shall see in Chapter 4). Thus, there is a strong motivation to devise a compressive sensing framework for exemplar-based ASR.

In addition to the reasons mentioned above, one of the driving motivations for exploring sparse modeling in speech research comes from the field of neuroscience and psycho-acoustics which have provided evidence[2] that human brain exploits sparse coding and hierarchical analysis of the stimuli at the level of cognitive processing and neural activities [Allen (1994); Olshausen and Field (1996)].

1.3 ASR as a Low-dimensional Subspace Modeling Problem

Speech data lies on low-dimensional manifolds, which can be efficiently modeled using low-rank [Liu et al. (2013)] or sparse modeling [Elhamifar and Vidal (2013)] approaches. However, state-of-the-art DNN based acoustic modeling in DNN-HMM hybrid approach utilizes low-rankness and sparsity typically only for model compression or model regularization [Kang et al. (2015); Yu et al. (2012); Srivastava et al. (2014)] and not specifically for modeling of class-specific low-dimensional manifolds to improve ASR performance. Therefore, an important focus of this thesis is on explicitly exploiting the low-dimensional multi-subspace structure of speech towards the goal of improving acoustic modeling for ASR.

In a typical large vocabulary ASR system, DNN posteriors usually have a dimension in the order of $\sim 10^3$ which is equal to the number of senones (context-dependent subphone units, defined later in Chapter 2) in the system. If these posterior features are seen as intermediate high-dimensional measurements, then the underlying acoustic information is embedded in unique low-dimensional subspaces which are usually superimposed with high-dimensional unstructured noise. While the low-dimensional structures are global and pertain to the whole population of a class, the noise is local and could be a result of erroneous estimations by the

[2] although not conclusive

DNN.

Our motivation is to extract these class-specific subspaces using either sparse[3] (dictionary based) or low-rank (PCA based) representations, and model them explicitly to bring improvements in ASR performance. Towards this goal, the current thesis postulates acoustic modeling for ASR as a problem of recovering low-dimensional class information from high-dimensional DNN posterior features.

1.4 Summary of Contributions

The goal of this thesis is to devise novel approaches to exploit low-dimensional structures in speech for improving state-of-the-art ASR approaches on challenging tasks (see datasets, described in Section 2.3). More specifically, the main contributions of this thesis can be summarized as follows:

- *Sparse modeling for word classification:* A novel compressive sensing and sparse recovery based framework is implemented for the task of Isolated Word Recognition (IWR) using dictionary learning algorithms and sparse modeling in the context of exemplar-based speech recognition. The proposed system is shown to outperform the conventional '*collection of exemplars*' model commonly used in exemplar-based approaches. Combinations of a variety of dictionary learning and sparse recovery algorithms are evaluated on this IWR task, and the best algorithms suitable for ASR are identified [Dighe et al. (2015)].

- *Sparse modeling for word sequence recognition:* The sparse modeling framework is extended for the task of Connected Digit Recognition (CDR). Use of context appending and Collaborative Hierarchical sparsity is also investigated for modeling sequential information with results showing potential for future research. Unlike previous exemplar-based ASR methods, this approach is a stand-alone sparsity-based method which is not hybridized with HMM outputs [Dighe et al. (2015)].

- *Robust DNN posteriors using dictionary based sparse projection:* Explicit modeling of senone-specific low-dimensional subspaces is proposed using dictionary learning and sparse coding over the DNN posteriors. DNN posteriors are transformed into projected posteriors which are shown to be more suitable targets for training acoustic models. Improvements in ASR performance on CDR task are shown for both clean and noisy conditions paving a way towards an effective robust ASR framework using DNN in unseen conditions. Presence of low-dimensional structures is further confirmed through Robust PCA (RPCA) analysis [Dighe et al. (2016)].

- *Low-rank and sparse soft targets for LVCSR:* In the context of LVCSR on meeting scenario

[3] sparse representations, in this case, are used for classwise low-dimensional modeling and not for the modeling of the whole multi-class data considered together.

conversations, this thesis shows that low-rank and sparse coding based reconstruction of DNN posteriors leads to a more accurate estimation of probabilities for underlying senone classes, thus leading to better ASR accuracy. Enhanced soft targets thus obtained are also shown to enable semi-supervised training using untranscribed data. Proposed approach is advanced further by incorporating sMBR sequence discriminative training criteria for training DNN acoustic models [Dighe et al. (2017a,b, 2018c)].

- *Improving Far-field ASR:* Improvements in far-field ASR are shown using low-rank and soft targets from parallel close-talk speech data. Another direction of research focuses on the enhancement of far-field speech acoustic features using low-rank and sparse models learned on close-talk speech. Under a multi-task learning framework, acoustic models for far-field ASR are improved by a parallel task of mapping distant speech acoustic features to their low-rank and sparse projections. Unlike previous works, the proposed approach does not need a complete parallel close-talk training set, but only the forced alignments and principal component matrices or sparse dictionaries [Dighe et al. (2018a)].
- *Quantifying quality of acoustic models using information theory:* This thesis proposes an information theoretic analysis that quantifies the satisfaction of various conditional independence assumptions made by hidden Markov models in HMM based ASR approaches. The analysis is used to compare the quality of a variety of HMM based ASR acoustic models such as GMMs, DNNs, recurrent neural networks (RNN) and time delay neural networks (TDNN). The analysis also substantiates why the low-rank and sparse reconstruction of DNN posteriors leads to better ASR [Dighe et al. (2018b)].

In addition, this thesis provides scripts [Dighe (2017)] for implementing the proposed techniques under the framework of Kaldi speech recognition toolkit [Povey et al. (2011)].

1.5 Thesis Outline

We describe below the main organization of this thesis, briefly describing the main goal of each of its constituting chapters

Chapter 2, *Background on automatic speech recognition*, presents the key components of the ASR pipeline with a particular focus on state-of-the-art DNN based acoustic modeling.

Chapter 3, *Background on compressive sensing and sparse recovery,* explains the basics of compressive sensing theory and gives details of various dictionary learning and sparse recovery algorithms which are considered in this thesis. A background of low-rank modeling approaches which are relevant to this thesis is also presented.

Chapter 4, *A posterior-based sparse modeling approach towards ASR,* presents our novel ASR approach which is based on hierarchical sparse modeling of DNN posteriors. Evaluation of this approach is shown on the tasks of isolated word recognition and connected digit recognition.

Chapter 5, *A low-dimensional senone subspace modeling approach towards ASR*, proposes the modeling of DNN posteriors in a classwise fashion using CS dictionaries and principal components. The motivation for this idea is provided through a rank analysis of DNN posteriors which reveals that senone-specific low-dimensional subspaces exist beneath the high-dimensional posterior space. The proposed approach is evaluated on a connected digit recognition task as well as a harder LVCSR task.

Chapter 6, *Applications of sparse and low-rank modeling for far-field speech ASR*, exploits the ideas and techniques built in the previous chapters to improve ASR performance on far-field speech. In this context, we use our techniques to perform speech enhancement as well as improve acoustic omdeling of far-field speech using parallelly recorded close-talk data.

Chapter 7, *On quantifying the quality of acoustic models in hybrid DNN-HMM ASR*, presents a novel information theoretic analysis framework for the qualitative assessment of acoustic models used in HMM based ASR.

Chapter 8, *Conclusions and directions for future work*, derives the main conclusions of this thesis and provide some possible directions for future work.

1.6 List of Notations

The conventions followed in this thesis are as follows:

- $\mathbb{R}$: set of real numbers.
- $\mathbb{R}^D$: set of D dimensional vectors over $\mathbb{R}$.
- $\{\cdot\}$ denotes a set.
- $[\cdot]$ denotes a row vector.
- $<\cdot>$ denotes a sequence.
- $[\cdot]^\top$ denotes a column vector.
- Non bold capital letters indicate size of dimensions and random variables.
- Non-bold small letters indicate scalars or functions.
- Bold capital letters indicate matrices.
- Bold small letters indicate column vectors.
- Subscript i to a matrix denotes i^{th} column of the matrix.
- Subscript $[i]$ to a matrix denotes i^{th} row of the matrix.
- Subscripts $i:j$ and $[i:j]$ to a matrix denote a range of columns and rows of the matrix, respectively.
- Subscript i to a vector usually denotes the i^{th} temporal instance of the vector.
- Subscript i to a scalar usually denotes that the scalar is i^{th} element of corresponding bold-faced vector or i^{th} member of a set.
- Use of superscripts is explained in the text wherever necessary.

2 Background on Automatic Speech Recognition

In this chapter, we provide a brief background on hidden Markov models (HMM) and the components of an HMM-based ASR system which are relevant to this thesis in Section 2.1. This section also provide details of the acoustic features and the various hierarchies of speech recognition units that we consider for modeling in this thesis. Section 2.2 describes the overall hybrid ASR pipeline with a special focus on DNN-based acoustic modeling. For a more detailed reading on HMM, conventional HMM-based ASR, and the neural network based hybrid connectionist approach to ASR, we refer the reader to the following resources: [Rabiner (1989); Jelinek (1997); Bourlard and Morgan (1994); Huang et al. (2001)]. Finally, Section 2.3 gives details of the databases that were used for evaluating the methods proposed in this thesis.

2.1 Hidden Markov Models

Over the last 40 years, hidden Markov models have served as the backbone of virtually all large-scale ASR systems [(Jelinek, 1976; Rabiner, 1989; Bourlard and Morgan, 1994)]. As a general framework, HMMs are often considered as the "wheel" of sequence processing in general, and speech processing in particular. Here, we introduce the basics of a Markov chain and use it to define HMMs.

2.1.1 Markov Chain

A Markov chain is a stochastical model that is used to describe random processes that satisfy Markov property. Markov property refers to a memory-less (or very limited memory) process, i.e., during the evolution of the process, the future states dependent only on a limited number of past few states and not on all the past. In a first-order Markov process, the conditional probability distribution of the future state is only dependent on the current state, and the process has no memory of the past. In terms of random variables, if the sequence of random variables $\mathcal{Q} =< Q_1, Q_2, \ldots, Q_T >$ follows a Markovian assumption, then

$$P(Q_t|\mathcal{Q}_1^{t-1}) = P(Q_t|Q_{t-1}) \tag{2.1}$$

where $\mathcal{Q}_1^{t-1} =< Q_1, Q_2, \ldots, Q_{t-1} >$. This also leads to the probability of the whole sequence $\mathcal{Q}$ being simplified as:

$$\begin{aligned} P(\mathcal{Q}) &= P(Q_1)P(Q_2|Q_1)\ldots P(Q_t|\mathcal{Q}_1^{t-1})\ldots P(Q_T|\mathcal{Q}_1^{T-1}) \\ &= P(Q_1)\prod_{t=2}^{T} P(Q_t|Q_{t-1}) \end{aligned}$$

If the random variables can take values from a set of K distinct states $\mathbb{Q} = \{q_1, q_2, \ldots, q_K\}$, then the Markov chain can be defined by the following two sets of probabilities:

- *Prior probabilities:* The probability that the Markov chain will start with a particular state.

$$\pi_k = P(Q_1 = q_k) \qquad s.t. \quad \pi_k \geq 0 \quad \forall k, \quad \sum_{k=1}^{K} \pi_k = 1 \tag{2.2}$$

- *Transition probabilities:* The probability that the Markov chain will go from one particular state to another.

$$a_{k,k'} = P(Q_t = q_{k'}|Q_{t-1} = q_k) \qquad s.t. \quad a_{k,k'} \geq 0 \quad \forall k', \quad \sum_{k'=1}^{K} a_{k,k'} = 1 \tag{2.3}$$

In a Markov chain, the state itself at each time step can be considered as a deterministic observation. A natural extension to Markov chains is a hidden Markov model as defined below.

2.1.2 Hidden Markov Model

A hidden Markov model is a Markov chain where each state generates an observable discrete symbol or a continuous-valued vector as per a state-conditional probability distribution function. While the emitted observations are *visible* to an observer, the underlying Markov process is *hidden*. The hidden state sequence is non-deterministic and can only be probabilistically estimated based on the observation sequence and the parameters of the model. Here, we consider only continuous density HMMs which emit real-valued multi-dimensional vectors as observations. The random variable denoting the observed sequence is defined as $\mathcal{X} =< X_1, X_2, \ldots, X_T >$.

Thus, a HMM can be completely defined by following components:

- Set of states $\mathbb{Q} = \{q_1, q_2, \ldots, q_K\}$: Random variable Q_t, denoting hidden state at time t, takes values from this set

- Set of observations, $\mathbb{R}^m$: Random variable X_t, denoting the observation emitted at time t, takes a value $\mathbf{x}_t \in \mathbb{R}^m$
- Prior probabilities π_k from (2.2)
- Transition probabilities $a_{k,k'}$ from (2.3)
- Emission probabilities $b_k(\mathbf{x})$: Probability of an observation $\mathbf{x} \in \mathbb{R}^m$ being generated when the underlying hidden state is q_k.

$$b_k(\mathbf{x}) = P(\mathbf{x}|q_k) \tag{2.4}$$

An HMM based on a first-order Markov chain involves two important assumptions (which will be revisited again in Chapter 7). The first assumption is the first-order Markovian assumption as explained in (2.1) i.e.

$$P(Q_t|\mathcal{Q}_1^{t-1}) = P(Q_t|Q_{t-1})) \tag{2.5}$$

The second assumption, famously called *HMM conditional-independence* assumption, states that the observation emitted at time t is dependent only on the hidden state at time t, and is conditionally independent of the past hidden state as well as observations, i.e.

$$P(X_t|\mathcal{X}_1^{t-1}, \mathcal{Q}_1^t) = P(X_t|Q_t) \tag{2.6}$$

2.1.2.1 Gaussian Mixture Model HMMs

One of the most commonly used versions of continuous probability density HMMs is based on multivariate Gaussian Mixture Models (GMM). In a GMM-HMM, each hidden state q_k in the set $\mathbb{Q}$ has a GMM associated with it such that the emission function $b_k(\mathbf{x})$ can be defined as:

$$b_k(\mathbf{x}) = \sum_{c=1}^{C} w_{kc}\, \mathcal{N}(\mathbf{x}, \mu_{kc}, \Sigma_{kc}) \tag{2.7}$$

where $\mathcal{N}(\cdot, \mu_{kc}, \Sigma_{kc})$ denotes the Gaussian probability density function with mean μ_{kc} and variance Σ_{kc} for c^{th} Gaussian mixture component of the GMM associated with state q_k, w_{kc} denotes the weight of c^{th} component, and C denotes the total number of Gaussian components.

Employing HMMs for any task usually results in one or more of the following three standard problems - 1) finding the posterior probability of an observation sequence given the HMM parameters, 2) finding the most likely hidden state sequence given an observation sequence and the HMM parameters, and 3) finding the parameters of the HMM given a set of observation sequences. Associated with addressing these three problems are the famous HMM-based algorithms - namely Forward-backward algorithm, Viterbi algorithm, and Baum-Welch algorithm respectively. We refer the reader to [Rabiner (1989)] for complete details on these algorithms.

2.2 Automatic Speech Recognition

ASR is the task of correctly converting a speech signal into the sequence of words which were spoken by the speaker. We formalize this problem mathematically as follows.

2.2.1 Mathematical Formulation of HMM-based ASR

In a typical HMM based ASR framework, the hypothesised word sequence $\hat{\mathcal{W}}$ is estimated from the sequence of acoustic features $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_t, \ldots \mathbf{x}_T\}$, where X_t is a standard acoustic feature at time t, as

$$\hat{\mathcal{W}} = \arg\max_{\mathcal{W}} P(\mathcal{W}|\mathcal{X}) \tag{2.8}$$

$$= \arg\max_{\mathcal{W}} \frac{p(\mathcal{X}|\mathcal{W})P(\mathcal{W})}{p(\mathcal{X})} = \arg\max_{\mathcal{W}} p(\mathcal{X}|\mathcal{W})P(\mathcal{W}) \tag{2.9}$$

where $p(\mathcal{W})$ is the probability of word sequence $\mathcal{W}$ estimated from a language model and $p(\mathcal{X}|\mathcal{W})$ is the likelihood of the acoustic sequence conditioned on the word sequence, estimated from an acoustic model. In the last step, we ignore the denominator probability $p(\mathcal{X})$ as it is independent of the word sequence $\mathcal{W}$ in the maximization argument. Assuming that the observation sequence $\mathcal{X}$ is generated by a hidden Markov model, the task at hand is to compute its probability by marginalizing over all possible hidden state sequences $\mathcal{Q}$ (i.e. using the Forward-Backward algorithm). Thus, $p(\mathcal{X}|\mathcal{W})$ is computed as

$$\begin{aligned} p(\mathcal{X}|\mathcal{W}) &= \sum_{\mathcal{Q}} p(\mathcal{X}|\mathcal{Q},\mathcal{W})P(\mathcal{Q}|\mathcal{W}) \\ &\approx \max_{\mathcal{Q}} p(\mathcal{X}|\mathcal{Q},\mathcal{W})P(\mathcal{Q}|\mathcal{W}) \\ &= \pi(q_{k_1}) \prod_{t=2}^{T} a_{q_{k_{t-1}} q_{k_t}} \prod_{t=1}^{T} p(\mathbf{x}_t|q_{k_t}) \end{aligned} \tag{2.10}$$

where $\hat{\mathcal{Q}} =< q_{k_1}, \ldots, q_{k_t}, \ldots, q_{k_T} >$ is the most probable state sequence obtained from the Viterbi algorithm [Rabiner (1989)] for decoding and $\pi_{q_{k_1}}$, $a_{q_{k_{t-1}} q_{k_t}}$ and $p(\mathbf{x}_t|q_{k_t})$ have usual meanings in context of a HMM as described in Section 2.1. The marginalization over all possible hidden state sequences $\mathcal{Q}$ is typically approximated just by using the most probable hidden sequence.

2.2.2 Key Components in the ASR Pipeline

A typical ASR system has been shown in Figure 2.1. The concerned speech signal for ASR is first passed through a signal processing component. This component enhances the signal to reduce noise and distortions due to the channel and outputs multi-dimensional acoustic features for the signal. The acoustic features are then passed through an acoustic modeling block. An acoustic model looks for the evidence of phonetic information in the acoustic features. It computes the data-likelihoods conditioned on various phonetic states, and this

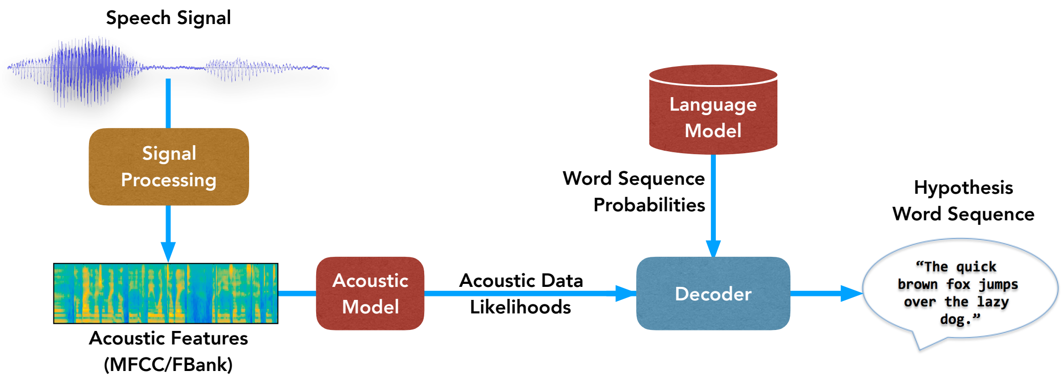
Speech Signal
Signal Processing
Acoustic Features (MFCC/FBank)
Acoustic Model
Acoustic Data Likelihoods
Language Model
Word Sequence Probabilities
Decoder
Hypothesis Word Sequence
"The quick brown fox jumps over the lazy dog."

Figure 2.1: Key components of the ASR pipeline.

information is passed to the decoder. A decoder could be considered as an implementation of an HMM which composes multiple hierarchies of speech together in a graph. The decoder combines the likelihoods from the acoustic model and various word sequence probabilities from a language model to search for the most probable (i.e., the least cost) path in the decoding graph to output the hypothesized word sequence.

The present thesis mainly focuses on improving the acoustic modeling component of the ASR pipeline using sparse and low-rank methods (Chapter 4 and Chapter 5). In the later part of the thesis (Chapter 6), we focus on both signal processing and acoustic modeling components jointly towards the goal of improving acoustic modeling by performing speech enhancement. In Chapter 7, we specifically focus on the acoustic modeling component as we do a qualitative analysis of different acoustic modeling techniques using concepts of information theory. We did not explore modifications or improvements in the language modeling component in this thesis.

Before we delve into details of the acoustic modeling component, we present below some standard type of acoustic features relevant to this thesis. We also discuss the units of speech usually modeled by the acoustic model.

2.2.3 Acoustic Features

In this thesis, we employ two standard types of acoustic features, namely log-Mel filterbank energy features (Fbank) and Mel-frequency cepstral coefficients (MFCC). The procedure to generate Fbank and MFCC features is almost similar except a few extra steps in the case of MFCC, as follows:

1. *Sampling and pre-emphasis:* Speech signal is captured at a fixed sampling rate, typically 8kHz or 16kHz. Sampling at a high-frequency rate can result in low-frequency components of the signal having high energy whereas the high-frequency components

might be subdued. Hence, a pre-emphasis operation is done to the original signal $s(t)$ as follows to amplify the high-frequency components of the signal:

$$s'(t) = s(t) - \alpha s(t-1) \tag{2.11}$$

where α, the first-order filter coefficient, is typically taken to be 0.97.

2. *Windowing and FFT:* The next step is to convert the pre-emphasized time-domain signal to the frequency-domain in order to analyze the evolution of different frequency components over time. A Fourier transform of the whole signal is not a suitable choice since the signal is not stationary and a signal-level transform would result in losing the information about the evolution of the frequency contours. To avoid these issues, the signal is analyzed in a sliding window fashion because we can assume stationarity for very short durations of time. Therefore, we apply fast Fourier transform (FFT) to one window at a time to get a frequency-domain representation of the short duration signal in the window. FFT of each window results in one spectral feature *frame* in frequency-domain, and all the frames of spectral features concatenated together adjacently are called the *spectrogram* of the signal. A typically used window length is 25ms which would contain 400 samples of the pre-emphasized time-domain signal if the sampling rate used is 16kHz. Before applying FFT, a Hamming window function is also applied to compensate for the fact that FFT assumes the time-domain signal to be of infinite length. The spectrogram output from FFT is converted to a power spectrum by taking square of the amplitudes in the spectrogram.
3. *Filterbank analysis:* In this step, Mel-scale filters are applied to the power spectrum to imitate human auditory perception which is more discriminatory for the lower frequencies as compared to the higher ones. Typically 40 overlapping triangular filters are applied whose centers are placed at equal intervals in the Mel-frequency domain. Finally, we compute the log of energy under each Mel-filter. At this stage, each frame of the spectrum has 40 Mel-filterbank energies. These features are called log Mel-filterbank energies or simply Fbank.
4. *Decorrelating by DCT:* The dimensions of the Fbank energy features are highly correlated. In order to decorrelate these features for enabling Gaussian modeling with diagonalized covariance as well as to discard high-frequency components of the power spectrum, we take discrete cosine transform (DCT) of the Fbank features and pick the first few coefficients (usually 13) which are significant for ASR. These coefficients are called Mel-frequency cepstral coefficients or simply MFCC.

In practice, both MFCC and Fbank features are typically used after appending first and second order delta features with them. MFCCs are used for GMM-HMM modeling as they have been decorrelated whereas DNNs can be directly trained with Fbank features as they are not very sensitive to correlated dimensions.

2.2.4 Speech Units for Acoustic Modeling: Words, Phonemes, and Senones

Selecting an appropriate speech unit for acoustic modeling is one of the most crucial aspects of an ASR system. The speech unit chosen should be such that distinct classes of the unit can be modeled with discriminable probability distributions over the acoustic features. There should be enough data available for modeling each class of the unit accurately. Also, the speech unit should be easily generalizable for unseen speech data such that test utterances with new words can still be recognized.

2.2.4.1 Words

One of the most natural choices for modeling speech could be at the word level. Words are language specific units having semantic meanings. For a small vocabulary task, they are a good choice for acoustic modeling as we might have enough data for training separate word models and as well as their context-dependent variations. On the contrary, the need for more data explodes with the size of the vocabulary in a large vocabulary speech recognition task. A significant amount of training data is required in this case, possibly with many variations of each word, for robust and accurate word models. Moreover, word models from the training data are not readily generalizable to new unseen words in the test data. Due to this, LVCSR systems typically do not employ word-based acoustic modeling whereas small vocabulary tasks such as digit recognition can work with word-based models.

2.2.4.2 Phonemes

Phonemes are linguistically distinct speech units which do not have any semantic meaning. Since they are defined only with respect to the constituent sound, they are not language-dependent. However, a given language might have a different set of phonemes than another language due to different sets of sounds needed to pronounce words in their respective vocabularies. There are nearly 40 phonemes in English as compared to about 170,000 words. Therefore, distinct phoneme models are easily trainable than word models as there is usually enough data for each phoneme class. A large vocabulary of words can be modeled as sequences of phoneme models concatenated together in an LVCSR task. Furthermore, a linguist can prepare a dictionary of all the words in the vocabulary mapped to their phonetic sequences, which makes it possible to generalize the phoneme-based acoustic model to unseen words in the test data. An example of a word to phonetic sequence is shown in Figure 2.2(a).

A phonetic model is typically modeled by a 3-state left-to-right HMM topology (Figure 2.2(b)) where each state could be modeled using a separate GMM. The states of the phone-HMM are sub-phonetic units which model the beginning, middle, and end of the phone. A similar HMM model is used to model the *silence* class. In phone-based acoustic modeling (termed as monophone ASR usually), the acoustic features are typically modeled using distinct probability distributions belonging to the phone-states. A downside of the monophone modeling approach is that we consider all the instances of a phoneme in the data to be acoustically similar. For example, when a phoneme q_i appears in two different phonetic contexts, e.g. $q_j - q_i - q_k$ and $q_{j'} - q_i - q_{k'}$, the left and right states of the phoneme q_i HMM might exhibit significant dissimilarities in acoustic features due to the context-dependent variations. A monophone model ignores these variations and models all instances of the phoneme under a common probability distribution function, resulting in a somewhat weak and inaccurate acoustic modeling. The term *(allo)phone* refers to a phoneme's acoustic realization which usually varies depending on the surrounding phoneme context and variations in the coarticulation mechanism. Modeling of phones is typically addressed by context-dependent sub-phonetic modeling described below.

2.2.4.3 Senones

To incorporate context-dependency in phoneme modeling, monophone HMMs are replaced with triphone HMMs. A triphone has a unique left and right context phoneme around the central phoneme. Therefore, a triphone is an example of allophones. States of the triphone HMMs are clustered across different phonetic models if the phonetic context is similar. This state tying limits the number of different models to be trained. Acoustic data assigned to each triphone state is further split using a decision tree [Young et al. (1994); Hwang et al. (1996)] such that each node asks a linguistic question to reduce the entropy or increase the likelihood of the data after the split. The leaves of the decision tree are termed as *senones*, and they are the most commonly used units for acoustic modeling in LVCSR systems because they provide a significant improvement in ASR performance over the monophone acoustic models. While the total number of logical states in a speech modeling HMM could be very large, the number of senones obtained after state tying is typically in the order of few thousands. A senone decision tree is shown in Figure 2.2(b).

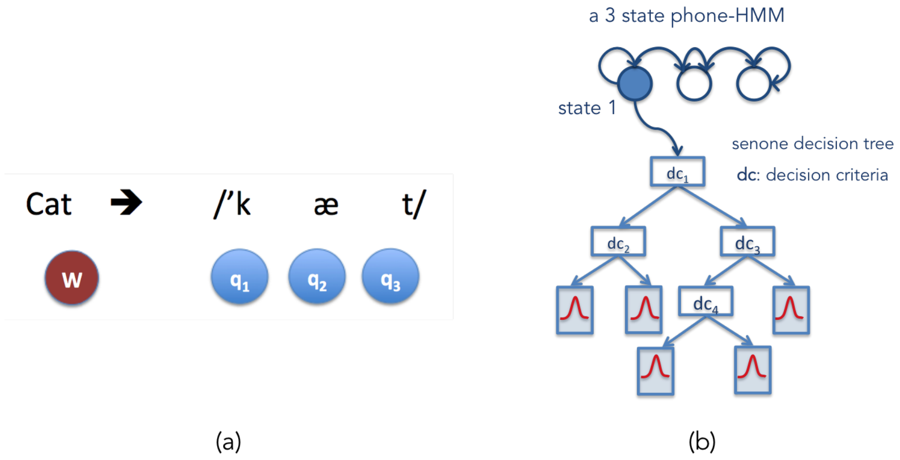

Figure 2.2: (a) Example of the phoneme sequence for a word. (b) An example of 3-state triphone HMM and the senone decision tree.

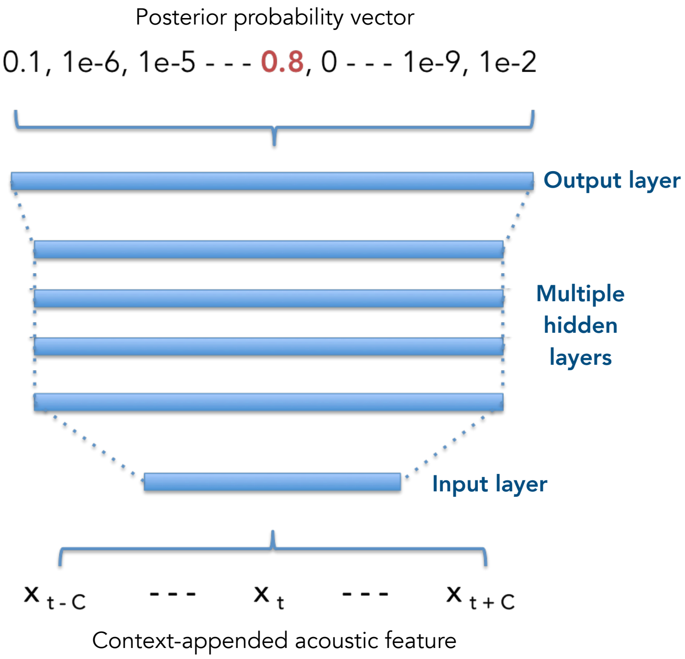
Posterior probability vector
0.1, 1e-6, 1e-5 - - - 0.8, 0 - - - 1e-9, 1e-2
Output layer
Multiple hidden layers
Input layer
x t - C
- - -
x t
- - -
x t + C
Context-appended acoustic feature

Figure 2.3: A deep neural network based acoustic model which predicts HMM state posterior probablities at the output using a context-appended acoustic feature as input.

In this thesis, we explore monophone-based acoustic models in the context of small vocabulary ASR and work with senones-based acoustic models for large vocabulary tasks.

2.2.5 DNN-HMM Hybrid Acoustic Models

In a hybrid DNN-HMM ASR system [Bourlard and Morgan (1994); Hinton et al. (2012)], the traditional GMM based modeling of state-specific probability distribution functions is replaced by a deep neural network model. As depicted in Figure 2.3, the DNN takes as input a context-appended acoustic feature vector and predicts the posterior probabilities of all senone classes at the output layer. The mapping from the acoustic features to the state posterior probabilities is done through multiple layers of non-linear transformations.

In a Bayesian GMM-HMM system, the frame likelihood $p(\mathbf{x}_t|q_t)$ required in (2.10) can be directly computed using the state-specific GMMs. In case of DNN-HMM acoustic models, it has to be indirectly approximated as follows:

$$p(\mathbf{x}|q_k) \sim \frac{p(\mathbf{x}|q_k)}{p(\mathbf{x})} = \frac{P(q_k|\mathbf{x})}{P(q_k)} \tag{2.12}$$

where the state posterior probability $P(q_k|\mathbf{x})$ is obtained at the output of the DNN and $P(q_k)$ is the prior probability of the state q_k obtained from its frequency count in the training data, yielding to an estimate of the *scaled likelihood* $\frac{p(\mathbf{x}|q_k)}{p(\mathbf{x})}$.

The DNN acoustic model shown in Figure 2.3 is a feedforward network but, in practice, it can be replaced by other neural network architectures like recurrent neural networks or convolutional neural networks [Sak et al. (2014); Waibel et al. (1990); Peddinti et al. (2015)]. Since the outputs of the network represent probabilities of the HMM states, we use a softmax layer as the last layer of the network. The vector of state posterior probabilities at the output layer of the DNN is also called a posterior feature (or simply a DNN posterior). Posterior features are more robust to speaker and environmental variations as compared to acoustic features at the input of the DNN. They are also known to be spiky and sparse as they live on a probability simplex. In this thesis, we heavily exploit the posterior features for sparse and low-rank modeling of speech for improving DNN-HMM ASR performance. Due to the dimensions of posterior features having a one-to-one correspondence with HMM states, they unveil the underlying class-specific low-dimensional subspaces very elegantly under sparse and low-rank modeling approaches.

2.2.5.1 DNN Training for ASR

Training of a DNN-HMM ASR system usually starts with training a GMM-HMM system first. For a typical LVCSR task, training the GMM-HMM system involves (1) creating the set of senones using decision tree based state tying and (2) learning the HMM parameters using the training data. Once the GMM-HMM system is learned, we force-align a sequence of senones over the training utterances using their ground-truth text transcript under the Viterbi algorithm. Framewise senone alignments of the training data provide us with outputs for training the DNN acoustic model. For this, the senone labels are converted into binary posterior vectors with probability 1 for the labeled senone and 0 everywhere else.

A DNN acoustic model can be trained either towards the goal of minimizing the framewise senone classification error or towards minimizing the sentence level error. Framewise training of the DNN is typically done by minimizing a cross-entropy (CE) loss function, whereas sentence-level errors can be minimized by using sequence discriminative loss functions such as sMBR or bMMI criteria [Povey]. We provide details of CE and sMBR loss functions in this section as they are relevant to the work in this thesis.

DNNs are trained using the error backpropagation algorithm [Rumelhart et al. (1986)] which utilizes the chain rule of calculus in order to compute the derivate of the loss with respect to each trainable parameter . These loss derivates are used to update the parameters of a feed-forward neural network as follows:

$$\mathbf{W}_{t+1}^{l} \leftarrow \mathbf{W}_{t}^{l} - \epsilon \Delta \mathbf{W}_{t}^{l} \tag{2.13}$$

$$\mathbf{b}_{t+1}^{l} \leftarrow \mathbf{b}_{t}^{l} - \epsilon \Delta \mathbf{b}_{t}^{l} \tag{2.14}$$

where ϵ is the learning rate, $\mathbf{W}_t^l$ and $\mathbf{b}_t^l$ are weight matrix and bias vector of the layer l after t^{th} update, and $\Delta\mathbf{W}_t^l$ and $\Delta\mathbf{b}_t^l$ are the average gradients of the selected loss function $\mathcal{L}$ with respect to the weight and bias respectively over a minibatch of examples from the training data.

Cross Entropy Loss: On a training example, if the target posterior vector is **t** and the DNN predicts a posterior vector **o**, then the cross entropy loss is given by:

$$\mathcal{L}(\mathbf{W},\mathbf{b};\mathbf{t},\mathbf{o}) = -\sum_{k=1}^{K} t_k \log(o)_k \tag{2.15}$$

where t_k and o_k are k^{th} components of DNN target and output vectors, respectively. By minimizing the CE loss over the whole training data, we minimize the Kullback-Liebler distance between the target probability distribution and the DNN output distribution.

sMBR Objective Function: Minimum Bayes risk (MBR) criteria [Goel and Byrne (2000); Veselỳ et al. (2013)] minimize the sentence level expected error with respect to different hierarchies of target alignments. In this thesis, we employ the state alignments based sMBR objective function for sequence discriminative training of DNN acoustic models. The sMBR objective is defined as:

$$\mathcal{L}_{sMBR} = \sum_{u=1}^{U} \frac{\sum_{W} p(\mathbf{X}_u|S^W)^\kappa P(W) A(W,W_u)}{\sum_{W'} p(\mathbf{X}_u|S^{W'})^\kappa P(W)} \tag{2.16}$$

where u is the utterance identifier, W_u and W denote reference and hypothesized word sequences, and $A(W,W_u)$ is the raw accuracy of the labels in state alignment for hypothesized word sequence W with respect to the state alignment for reference word sequence W_u. $\mathbf{X}_u$ denotes the acoustic feature sequence $<\mathbf{x}_1,\mathbf{x}_2,\ldots,\mathbf{x}_T>$ and κ is acoustic score scaling factor. The numerator and denominator in the sMBR objective function are sums taken over all word sequences in decoding lattice for the utterance u such that HMM topology and the effect of language model are taking into consideration. sMBR criterion essentially computes a weighted average raw accuracy $A(W,W_u)$ over all word sequences W's in the lattice. Maximizing this objective results in minimizing the expected state-error rate. Sequence discriminative training of a DNN acoustic model is done by first training a DNN by minimizing the CE loss function. The CE loss based DNN is then used to generate decoding lattices for all the utterances in the training data. Finally, the DNN parameters are tuned to maximize the sMBR criterion.

2.2.6 KL-HMM Based Acoustic Modeling

Kullback-Leibler divergence based HMM (KL-HMM) formulation was proposed in [Aradilla et al. (2007, 2008)]. In this approach, each HMM state q_k is characterized by a target multinomial distribution parametrized via a probability simple vector $\mathbf{y}^k$. The state-conditional data likelihoods required in (2.10) are replaced by a cost function associated with each state. The cost for state q_k at time t is given by the KL divergence between its target distribution $\mathbf{y}^k$ and the current posterior feature $\mathbf{z}_t$ as:

$$c_k(t) = \mathrm{KL}(\mathbf{y}^k \| \mathbf{z}_t) = \sum_{k'=1}^{K} y_{k'}^k \log \frac{y_{k'}^k}{z_{t(k')}} \tag{2.17}$$

where $y_{k'}^k$ and $z_{t(k')}$ represent the k' th element of $\mathbf{y}^k$ and $\mathbf{z}_t$ respectively. With this interpretation, hybrid DNN-HMM framework can be seen as a particular case of a KL-HMM where state target distributions are predefined as binary probability vectors.

In the description above, we have assumed that the state-specific target distributions $\mathbf{y}^k$ and the posterior features $\mathbf{z}_t$ correspond to K-dimensional vectors where K is the total number of states. In practice [Bahaadini et al. (2014)], one can compute D-dimensional posterior features (where $D \neq K$) at the output of a DNN such that the posterior probabilities relate to some speech modeling unit other than the K-class HMM state units. Accordingly, the state-specific distributions are also learned as D-dimensional probability simplex vectors, and the KL divergence based cost can still be computed using (2.17). Formulating the KL-HMM this way provides the liberty to model a different speech modeling unit at the output of the DNN as compared to the one used for constructing the HMM back-end.

2.2.7 ASR Evaluation Metric

The most commonly used metric for measuring the performance of an automatic speech recognition system is *word error rate* (WER). In this thesis, we mainly used WER to compare the performance of various systems. Given the reference word sequences over some test data and the word sequences hypothesized by an ASR system, the word error rate is defined as:

$$\text{WER (in \%)} = 100 \times \frac{\text{Substitutions} + \text{Deletions} + \text{Insertions}}{\text{Number of words in the reference}} \tag{2.18}$$

where the numerator has a count of substitutions, deletions and insertions in the hypothesized word sequences as compared to the reference sequences. WER is typically presented as a percentage, and a lower WER signifies higher accuracy in speech recognition.

In some experiments based on isolated word recognition, we use the *word recogntion accuracy* for evaluating the performance of various ASR systems. This accuracy is defined as follows:

$$\text{Word Recognition Accuracy (in \%)} = 100 \times \frac{\text{Number of words recognized correctly}}{\text{Total number of words}} \tag{2.19}$$

2.3 Databases

The databases considered in this thesis for ASR lie in three different categories of varying difficulty. Experiments based on a novel compressive sensing ASR approach in Chapter 4 have been conducted on databases involving isolated word and connected digit recordings. For LVCSR experiments using enhanced low-rank and soft targets in Chapter 5, we utilize databases involving conversational speech recordings. In addition, we also utilize speech data from other sources for semi-supervised training. Details of all the databases used in this thesis are given below.

2.3.1 PhoneBook: NYNEX Isolated Words

PhoneBook is a phonetically-rich telephone-speech database [Pitrelli et al. (1995)] created specifically for developing isolated word recognition and keyword spotting technology. The database has single channel recordings of isolated words recorded at 8kHz sampling rate. The complete database has nearly 8k distinct words with an average of 11.7 examples of each word spoken by different speakers. More details of PhoneBook database are given in Table 2.1. All the word utterances are spoken by native American English speakers. We perform experiments on two different subsets of PhoneBook for isolated word recognition task - an easier 75 words vocabulary task and a more challenging 600 words vocabulary task. For each word, we use 4 examples for training the word-based models and the rest of the examples (typically 7-8) for testing.

Table 2.1: Details of PhoneBook database.

Detail	Count
Speech data (in hours)	23
Number of utterances	93,667
Distinct words	7,979
Samples per word	11.7 (from different speakers)
Distinct speakers	1,358

2.3.2 Numbers Database

Numbers database is a subset of Numbers'95 corpus [Cole et al. (1995)] which contains strings of spoken digits recorded over a telephone channel at 8kHz sampling rate. The subset contains those utterances of Numbers'95 corpus which contain only the 30 most frequent words in the original corpus. The rest of the utterances are discarded. Therefore, the vocabulary size of Numbers database in 30. Examples of the utterances in the database include phone numbers, zip codes and birthdays. The database is partitioned as shown in Table 2.3.

Another smaller subset of Number database is referred to as Digits database [Aradilla (2008)] containing spoken sequences of only the 10 digits (from 'zero' to 'nine') and an alternative pronunciation 'oh' for zero. For this dataset, there are 12 distinct word classes corresponding to the 11 words plus one for silence or pause. The training set consists of 8253 utterances (4.5 hours of speech) and the test set contains 2820 utterances.

Table 2.2: Details of Numbers database.

Detail	Count (partition-wise)		
	Train	Dev	Test
Speech data (in hours)	6.2	2.2	2.3
Number of utterances	10,441	3,582	3,621
Running words	50,358	17,597	17,835

2.3.3 AMI Meeting Corpus

The AMI corpus [McCowan et al. (2005)] (http://groups.inf.ed.ac.uk/ami/corpus/) contains recordings of spontaneous conversations between a group of participants in meeting scenarios. The meeting scenarios have been designed such that the participants freely discuss and debate over some ideas. The meetings were recorded in English, although the speakers were mostly non-native. The recordings were done in three different rooms with different acoustic environments across three geographical locations in the UK, the Netherlands, and Switzerland. AMI corpus is multi-modal and provides audio recordings from close-talk as well as far-field microphones. Other modalities include individual and room-view video cameras, output from a slide projector and an electronic whiteboard, and individual electronic pens. In this thesis, we focus on speech data recorded using close-talk and far-field microphones. Due to the conversational style of speaking and the speakers frequently overlapping and interrupting other speakers' speech, the AMI corpus has proved to be a challenging task in recent large vocabulary ASR research.

The close-talk microphone speech is termed as individual headset microphone (IHM) condition in AMI, whereas the far-field microphone speech is termed as the single distant microphone condition. All meeting rooms had eight far-field microphones in a circular array between the meeting participants. The first microphone (mic-id 1) is typically used as the source of SDM data for far-field ASR. Both the close-talk and far-field speech streams have been recorded parallelly. They are time synchronized, and the word transcripts are obtained by force-aligning using a speech recognition system.

Table 2.3: Details of AMI database.

Detail	Count (partition-wise)		
	Train	Dev	Test
Speech data (in hours, approx.)	81	9	9
Number of utterances	108,221	13,059	12,612
Running words	802,604	94,914	89,635

The dataset is available at 16kHz sampling rate with nearly 100 hours of meeting recordings divided approximately as 81 hours train set, 9 hours *dev* and 9 hours *eval* set. We use 10% of the training data for cross-validation during DNN training, whereas the *dev* set is usually used for tuning the hyper-parameters of our proposed approaches. We use a pronunciation dictionary of ~47K words and a trigram language model for decoding in our ASR experiments on AMI.

2.3.4 ICSI Meeting Corpus

The ICSI meeting corpus [Janin et al.] is an in-domain speech database with respect to the AMI meeting corpus. It contains ~72 hours of speech recordings from 75 meetings. The meetings were recorded at ICSI Berkeley and consisted of conversational speech among the participants. The recordings were done using both close-talk and far-field microphones at 48kHz sampling rate. We only use the close-talk subset of ICSI corpus for experiments, and it is downsampled to 16kHz to match the conditions in AMI meeting corpus. This database is used in this thesis for semi-supervised training of acoustic models. Therefore, we do not use the transcriptions provided with this database. The details of this corpus are summarized in Table 2.4.

Table 2.4: Details of ICSI meeting corpus.

Detail	Count
Speech data (in hours)	72
Distinct words (approx.)	13k
Running words (approx.)	795k
Distinct speakers	53

2.3.5 Librispeech Corpus

LibriSpeech corpus [Panayotov et al. (2015)] is a speech database consisting of approximately 1000 hours of read English speech at 16kHz sampling rate. The recordings contain read audiobooks which are fairly different than conversational speech. Therefore, we consider LibriSpeech as an out-of-domain database with respect to the AMI meeting corpus. In this thesis, we use a 100 hour subset of Librispeech for semi-supervised training of acoustic models. The details of this 100 hour subset is summarized in Table 2.5.

Table 2.5: Details of 100 hour subset of Librispeech database.

Detail	Count
Speech data (in hours)	100.6
Distinct words (approx.)	34k
Running words (approx.)	990k
Distinct speakers	251

3 Background on Compressive Sensing and Sparse Recovery

This thesis relies heavily on the concepts of compressive sensing (CS) theory towards the goal of improving ASR. Therefore, we devote this chapter to an appropriate background of the same. We provide a brief introduction to CS theory in Section 3.1 followed by an overview of some relevant dictionary learning and sparse coding algorithms in Section 3.2 and Section 3.3 respectively. In Section 3.4, we discuss principal component analysis, a popular data processing technique, which has been exploited in this thesis for low-dimensional modeling of speech.

3.1 Introduction

Compressive sensing is a relatively recent area of research in signal processing [Candès and Wakin (2008)] which deals with efficiently sampling and reconstructing a signal by exploiting its intrinsic sparse structure. Since its inception, CS has found successful applications in a wide variety of technological domains such as compression of natural audio, image and video signals [Plumbley et al. (2010); Usevitch (2001); Schmid-Saugeon and Zakhor (2004)], data denoising [Jafari and Plumbley (2009); Aharon et al. (2006)], medical imaging [Mailhé et al. (2009); Tošić et al. (2010)], netword tomography [Firooz and Roy (2010)], etc.

According to the Nyquist-Shannon sampling theorem [Shannon (1949)], the sampling rate required to capture a signal fully must be at least twice of the maximum frequency present in the signal. While this so-called Nyquist rate puts a lower bound on the minimum number of samples required to capture a signal, the CS theory asserts that under certain conditions, the signal can be efficiently reconstructed from a far fewer number of samples than as governed by the Nyquist rate. In doing so, the CS theory assumes that many natural signals such as images and audio have an internal structure which is inherently sparse and leads to a very small information rate. This assumption is well supported by the famous *Ockham*'s razor[1] which favors fewer variables and simpler explanations to describe a natural phenomenon rather than other competing complex explanations. It is important to mention here that the sampling theorem concerns itself with the bandwidth of the signal to determine the sampling

[1] Also called *"lex parsimoniae"* in Latin, which means "the law of parsimony".

rate for a full recovery of the signal. On the other hand, CS relies on the internal structure of the signal for an efficient but not necessarily an exact reconstruction. Therefore, CS theory dictates that given an appropriate basis set, a sparse signal can be expressed very concisely. In this regard, Compressive Sensing theory has the following two related tasks at hand:

- **Compressive sensing of a signal:** Given a signal $\mathbf{a}$ of dimension M, find a sensing matrix $\mathbf{D} \in \mathbb{R}^{K \times M}$ such that the signal can be captured efficiently and non-adaptively by correlating it with the rows of $\mathbf{D}$, thus making only K measurements where $K < M$. The compressed K-dimensional signal $\mathbf{z}$ can be represented as:

  $$z_k = \langle \mathbf{d}_k, \mathbf{a} \rangle, \qquad k = 1, \ldots, K \tag{3.1}$$

  where z_k is the k^{th} measurement of the signal $\mathbf{a}$ corresponding to its correlation with $\mathbf{d}_k$, the k^{th} row of the sensing matrix.
- **Sparse recovery of a compressed sparse signal:** Given a compressed signal $\mathbf{z}$ of dimension K, find an appropriate dictionary $\mathbf{D} \in \mathbb{R}^{K \times M}$ so as to express $\mathbf{z}$ as a linear combination of a very few atoms (column vectors) of the dictionary as

  $$\mathbf{z} = \mathbf{D}\,\mathbf{a} \tag{3.2}$$

  where the sparse representation of $\mathbf{z}$ over atoms (columns) of $\mathbf{D}$ leads to the reconstruction of the underlying sparse signal $\mathbf{a}$ of dimension M.

The signal $\mathbf{a} \in \mathbb{R}^M$ is called N-sparse if only $N \ll M$ entries of $\mathbf{a}$ have nonzero values. We call the set of indices corresponding to the non-zero entries as the support of $\mathbf{a}$. The CS theory asserts that only $K = O(N \log(M/N))$ linear measurements, denoted by $\mathbf{z} \in \mathbb{R}^K$ and obtained as $z_k = \langle D_k, \mathbf{a} \rangle$ in (3.1), suffice to reconstruct $\mathbf{a}$, where $K < M$. The sensing matrix $\mathbf{D} \in \mathbb{R}^{K \times M}$ can also be interpreted as an over-complete dictionary designed for recovering the sparse representation $\mathbf{a}$ as per (3.2). *Overcompleteness* [Lewicki and Sejnowski (2000)] refers to the dictionary $\mathbf{D}$ having more basis vectors M than the dimension K of the compressed signal space. Please note that the sensing dictionary used in (3.1) for compressing a signal and the overcomplete dictionary used in (3.2) for recovering a sparse signal have theoretically different roles and applications and more details can be found in [Foucart and Rauhut (2013)]. The application of CS theory in this thesis concerns mostly with the sparse recovery (or signal reconstruction) problem described by (3.2) and we do not consider the sensing problem (3.1) here.

Formally, sparse recovery of $\mathbf{a}$ is achieved by solving the following optimization problem:

$$\min_{\mathbf{a} \in \mathbb{R}^M} \|\mathbf{a}\|_0 \quad \text{subject to} \quad \mathbf{z} = \mathbf{D}\,\mathbf{a} \tag{3.3}$$

where the counting function $\|.\|_0 : \mathbb{R}^M \longrightarrow \mathbb{N}$ returns the number of non-zero components in its argument, i.e. the ℓ_0-norm of $\mathbf{a}$. Due to $K < M$, we have more unknowns in (3.3) than the number of equations, leading to infinitely many solutions for $\mathbf{a}$. An underdetermined system of linear equations like (3.3) is in general an NP-hard problem [Donoho (2006)]. Moreover, there are $\binom{M}{N}$ combinatorial choices for an N-sparse solution, and a brute-force search for the best N-sparse solution is still infeasible due to the exponential complexity of the problem.

In practice, the non-convex objective of (3.3) is often relaxed to ℓ_1-norm based convex optimization problem which can be solved in polynomial time. The modified problem is stated as:

$$\min_{\mathbf{a}\in\mathbb{R}^M} \|\mathbf{a}\|_1 \quad \text{subject to} \quad \mathbf{z} = \mathbf{D}\,\mathbf{a} \tag{3.4}$$

where the ℓ_1-norm, $\|\mathbf{a}\|_1$ is defined as sum of the absolute values of the components of **a**. It has been shown in literature [Donoho and Elad (2003)] that (3.3) and (3.4) lead to equivalent sparse solutions for **a**.

According to [Candes and Tao (2005)], the successful reconstruction of **a** by solving the optimization problem in (3.4) requires that (1) **a** is sufficiently sparse and (2) the dictionary **D** satisfies the *restricted isometric property* (RIP). According to this property, if there exists a constant δ_N, where $1 \leq N \leq M$, such that for every N-dimensional vector $\mathbf{a}_N$ and every $K \times N$ submatrix $\mathbf{D}_N$ of **D**, the following condition holds:

$$(1-\delta_N)\|\mathbf{a}_N\|_2^2 \leq \|\mathbf{D}_N\mathbf{a}_N\|_2^2 \leq (1+\delta_N)\|\mathbf{a}_N\|_2^2 \tag{3.5}$$

then, the dictionary **D** is said to satisfy the restricted isometric property with the isometry constant δ_N. Since obtaining an appropriate dictionary is crucial for efficient reconstruction of the signal, dictionary learning is considered as an equally important aspect of CS theory as sparse recovery. We point the reader to the following sources [Elad (2010); Rish and Grabarnik (2014); Tosic and Frossard (2011)] for detailed reviews of various dictionary learning and sparse recovery algorithms.

In the following subsections, we review some dictionary learning and sparse recovery techniques that are relevant to the work done under this thesis. These methodologies mostly focus on solving the sparse recovery problem as expressed in (3.3) and (3.4).

3.2 Dictionary Learning

The goal of dictionary learning is to optimize for an overcomplete basis set such that the training feature vectors can be characterized as a sparse linear combination of the basis vectors. This approach assumes that the training data lives in a low-dimensional (non-Euclidean) space that can be modeled as a union of sub-spaces. An overcomplete dictionary attempts not only to capture the broad range of variability that the data can exhibit but also helps in *decompressing* the initial compact feature-space to a high dimensional sparse space where discrimination between various data phenomena becomes easier. This desirable property of dictionary learning is precisely what we need for the task of speech recognition where the variability comes from countless sources like gender, age, accent, surroundings, etc. The other requirement for our task is the efficient scalability of the dictionary learning algorithm to larger datasets. With the availability of huge datasets, an algorithm which can utilize all the available knowledge is preferred.

Given a training set of features $\mathbf{Z} = [\mathbf{z}_1, ..., \mathbf{z}_T] \in \mathbb{R}^{K\times T}$, a dictionary $\mathbf{D} \in \mathbb{R}^{K\times M}$ and sparse representation $\mathbf{A} = [\mathbf{a}_1, ..., \mathbf{a}_T] \in \mathbb{R}^{M\times T}$ for $\mathbf{Z}$; the ℓ_1-norm based sparse recovery objective function for classical dictionary learning techniques is defined as

$$\min_{\mathbf{D},\mathbf{A}} \frac{1}{T} \sum_{t=1}^{T} \left(\frac{1}{2} \|\mathbf{z}_t - \mathbf{D}\mathbf{a}_t\|_2^2 + \lambda \|\mathbf{a}_t\|_1 \right) \tag{3.6}$$

where λ is the regularization parameter. The first term in this expression, quantifies the *reconstruction error* whereas the second term controls the sparsity of $\mathbf{a}_t$. The joint optimization of this objective function with respect to both $\mathbf{D}$ and $\mathbf{a}_t$ simultaneously is non-convex. On the other hand, it can be solved as a convex objective function by optimizing for one quantity while keeping the other one fixed. Depending on the task, the cost functions other than Euclidean distance (i.e. ℓ_2-norm) may also be preferred such as Kullback-Leibler divergence. In this thesis, we focus on the performance of two main approaches to dictionary learning-namely, K-SVD and online dictionary learning algorithm. Details of these techniques are briefly summarized below.

3.2.1 K-SVD Algorithm

K-SVD algorithm, developed by [Aharon et al. (2006)], is one of the most prominent algorithms for dictionary learning. It roughly generalizes the idea of K-means clustering to the task of dictionary learning. The dictionary is learned atom by atom using the singular value decomposition (SVD) technique to minimize the quadratic reconstruction error associated to each atom. In the original paper [Aharon et al. (2006)] on K-SVD algorithm, K denotes the number of atoms of the dictionary which is denoted by M in this thesis. Therefore, as per the notations defined in this thesis for (3.6), the dictionary has M atoms and it projects M-dimensional data to a K-dimensional space. The K-SVD algorithm works as follows. The dictionary is first (warm) initialized as $\mathbf{D}$ and the sparse representation $\mathbf{A}$ of the training features $\mathbf{Z}$ is obtained through any convenient sparse recovery algorithm. Then, a residual error $\mathbf{E}_j$ is defined when atom $\mathbf{d}_j$ is removed from the dictionary along with the its corresponding coefficients, i.e. j^{th} row of $\mathbf{A}$ which is denoted as $\mathbf{a}_{[j]}$. Therefore, the residual can be written as:

$$\mathbf{E}_j = \mathbf{Z} - \sum_{i\neq j} \mathbf{d}_i \mathbf{a}_{[i]} \tag{3.7}$$

In terms of the residual E_j, the reconstruction error term of (3.6) can be written as:

$$\|\mathbf{Z} - \mathbf{D}\mathbf{A}\|_F^2 = \|\mathbf{E}_j - \mathbf{d}_j \mathbf{a}_{[j]}\|_F^2 \tag{3.8}$$

Now, the goal is to update each dictionary atom and its associated sparse coefficients through

$$\mathbf{d}_j^{\text{new}}, \mathbf{a}_{[j]}^{\text{new}} = \arg\min \|\mathbf{E}_j - \mathbf{d}_j\, \mathbf{a}_{[j]}\|_{\text{F}}^2, \qquad \text{for } j = 1, \dots, M \tag{3.9}$$

The SVD is then used to find the closest (in least square terms) rank-1 decomposition of $\mathbf{E}_j$ to update $\mathbf{d}_j$ and $\mathbf{a}_{[j]}$. To ensure that sparsity constraint is enforced on $\mathbf{a}_{[j]}^{\text{new}}$ in (3.9), only those columns of $\mathbf{E}_j$ are used for SVD decomposition that correspond to $\mathbf{z}_t$'s in $\mathbf{Z}$ which use the atom $\mathbf{d}_j$ in their sparse representation. This procedure is repeated for all atoms of the dictionary.

$$\mathbf{d}_j^{\text{new}}, \mathbf{a}_{[j]}^{\text{new}} = \underset{\text{Rank-1}}{\text{SVD}} \left(\mathbf{E}_j^{\textit{using relevant columns}} \right) \tag{3.10}$$

The column $\mathbf{d}_j$ in the dictionary is updated with $\mathbf{d}_j^{\text{new}}$ whereas $\mathbf{a}_{[j]}^{\text{new}}$ is discarded.

3.2.2 Online Dictionary Learning

An online optimization algorithm for dictionary learning was proposed by [Mairal et al. (2010)] based on stochastic approximations. The algorithm alternates between a step of sparse recovery for the current training feature $\mathbf{z}_t$ and then optimizes the previous estimate of dictionary $\mathbf{D}^{(t-1)}$ to determine the new estimate $\mathbf{D}^{(t)}$ using stochastic gradient descent. The algorithm employs LARS-Lasso algorithm for the sparse recovery which is explained in Section 3.3. Due to its online nature, this algorithm can handle very large datasets which makes it a favorable candidate for application in ASR. It has also been shown to be dramatically faster as compared to full-batch algorithms [Aharon et al. (2006); Olshausen and Field (1997)] typically used for learning dictionaries from large-scale datasets. The algorithm has been shortly summarized in Algorithm 1 and complete details can be found in [Mairal et al. (2010)].

Algorithm 1 Online Dictionary Learning

```
Require: : $\mathbf{Z} = [\mathbf{z}_1, ..., \mathbf{z}_T] \in \mathbb{R}^{K \times T}, \lambda \in \mathbb{R}$ : regularization parameter, initial estimate for dictionary $\mathbf{D}^{(0)} \in \mathbb{R}^{K \times M}$
for $t = 1$ to $T$ do
    Sparse Coding of $\mathbf{z}_t$ to determine $\mathbf{a}_t$:
        $\mathbf{a}_t = \arg\min_{\mathbf{a}} \left\{ \frac{1}{2} \|\mathbf{z}_t - \mathbf{D}^{(t-1)}\mathbf{a}\|_2^2 + \lambda \|\mathbf{a}\|_1 \right\}$    (3.11)
    Updating $\mathbf{D}^{(t)}$ with $\mathbf{D}^{(t-1)}$ as warm restart:
        $\mathbf{D}^{(t)} = \arg\min_{\mathbf{D}} \left\{ \frac{1}{t} \sum_{i=1}^{t} (\frac{1}{2} \|\mathbf{z}_i - \mathbf{D}\mathbf{a}_i\|_2^2 + \lambda \|\mathbf{a}_i\|_1) \right\}$    (3.12)
end for
return $\mathbf{D}^{(T)}$
```

3.3 Sparse Recovery

The computational methods to solve the sparse recovery problem expressed in (3.3) and (3.4) are reviewed in [Tropp and Wright (2010)]. Two of the major sparse recovery approaches reviewed in this chapter are based on- 1) greedy pursuit of basis vectors using orthogonal matching pursuit (OMP) and 2) convex relaxation of the problem to ℓ_1-norm using Lasso (Tibshirani, 1996). We also discuss some variants of Lasso here- namely hierarchical group Lasso and its collaborative version. The details of these algorithms are summarized below.

3.3.1 Orthogonal Matching Pursuit

One of the major algorithmic approaches to sparse recovery relies on a greedy pursuit of basis vectors referred to as orthogonal matching pursuit (OMP). The OMP is an iterative greedy method which finds a solution for the ℓ_0-norm sparse recovery problem mentioned in (3.3) by repeatedly identifying one or more atoms of the dictionary that yield the highest improvement in minimization of the reconstruction error [Davis et al. (1997); Tropp and Wright (2010)]. A significant advantage of this approach is that it does not need to relax the ℓ_0-norm criterion, so one can control the sparsity as required. The stopping criterion can be chosen by fixing the number of dictionary atoms which will have non-zero coefficients. The steps followed by OMP have been summarized in Algorithm 2.

Algorithm 2 Orthogonal Matching Pursuit

```
Require: : A signal $\mathbf{z} \in \mathbb{R}^K$ and a dictionary $\mathbf{D} \in \mathbb{R}^{K \times M}$
1: Initialize an index set $\Omega_0 = \emptyset$, a residual error $\mathbf{r}_0 = \mathbf{z}$, and a counter $c = 1$.
2: Find an atom $\mathbf{d}_{\hat{j}}$ of $\mathbf{D}$ which has the highest correlation with the previous residual $\mathbf{r}_{c-1}$ as:

       $$\hat{j} = \arg\max_{j} |\langle \mathbf{r}_{c-1}, \mathbf{d}_j \rangle| \quad (3.13)$$

   and set $\Omega_c$=$\Omega_{c-1} \cup \{\hat{j}\}$.
3: Find the best coefficients corresponding to the dictionary columns that have been chosen so far.

       $$\mathbf{a}_c = \arg\max_{\mathbf{a}} \|\mathbf{z} - \mathbf{D}_{\Omega_c}\mathbf{a}\|_2 \quad (3.14)$$

   where $\mathbf{D}_{\Omega_c}$ is a sub-matrix of $\mathbf{D}$ corresponding to the columns indexed by the index set $\Omega_c$.
4: Update the residual:

       $$\mathbf{r}_c = \mathbf{z} - \mathbf{D}_{\Omega_c}\mathbf{a}_c \quad (3.15)$$

   Repeat steps (2)-(4) until desired number of atoms have been selected from $\mathbf{D}$.
5: Return sparse representation $\mathbf{a}$ with non-zero coefficients from the final execution of (4).
```

3.3.2 Lasso

An alternative to the greedy sparse recovery is to relax the problem stated in (3.3) by replacing the non-convex ℓ_0-norm based objective with a convex ℓ_1-norm based objective function referred to as the least absolute shrinkage and selection operator (Lasso) [Tibshirani (1996)]. It is known that relaxing the combinatorial problem of ℓ_0-norm to ℓ_1-norm constraint leads to equivalent sparse solutions for **a** (as shown in Figure 3.1(a)). While the former is NP-hard, the relaxed formulation admits efficient polynomial time algorithms. Furthermore, the solutions of ℓ_1-norm minimization are less sensitive to noise. The optimization problems which is solved in Lasso is

$$\min_{\mathbf{a}} \frac{1}{2}\|\mathbf{z}-\mathbf{D}\mathbf{a}\|_2^2 + \lambda\|\mathbf{a}\|_1 \tag{3.16}$$

The standard Lasso problem can be solved by various convex optimization techniques. One of the efficient and computationally fast techniques is least angle regression (LARS) implementation whose details can be found in [Efron et al. (2004)]. In this thesis, we employ LARS implementation for sparse recovery of **a** using Lasso objective function in (3.16).

The compressive sensing based ASR framework proposed in Chapter 4 poses DNN posterior features (as described in Section 2.2.5) elegantly into formulations which lead to *group*, *hierarchical* and *collaboratively structured* sparsity. Therefore, we leverage some variants of Lasso which specifically deal with these cases, as briefly discussed below.

Group Lasso In some sparse recovery problems, the goal is not to identify individual atoms of the dictionary which are responsible for the signal reconstruction, but to determine a group of atoms which when activated together can define a particular subspace spanned by the dictionary. If a dictionary **D** has M atoms, we define a set of groups $\mathcal{G} = [G_1, \cdots, G_l, \cdots, G_L]$, which is simply a partitioning over the dictionary atom indices, where $G_l \subseteq \{1, \ldots, M\}$. Using the partitioning $\mathcal{G}$, the Group Lasso objective proposed in [Yuan and Lin (2006)] can be written as:

$$\min_{\mathbf{a}} \frac{1}{2}\|\mathbf{z}-\mathbf{D}\mathbf{a}\|_2^2 + \lambda\psi_{\mathcal{G}}(\mathbf{a}) \tag{3.17}$$

where $\psi_{\mathcal{G}}$ is defined as $\psi_{\mathcal{G}} := \sum_{G\in\mathcal{G}} \|\mathbf{a}_G\|_2$ and $\|\mathbf{a}_G\|_2$ is the ℓ_2 norm of a subvector of **a** which corresponds to the indices in the partition G. Therefore, group lasso objective generalizes the ℓ_1-norm sparsity to the level of groups. While a very few groups from $\mathcal{G}$ are selected during sparse recovery, the atoms inside the selected groups can be densely activated (as shown in Figure 3.1(b)).

Hierarchical Lasso (HiLasso) In HiLasso [Friedman et al. (2010); Meier et al. (2008); Sprechmann et al. (2011)], sparsity is sought at a group level as well as the level of the individual atoms of the dictionary (Figure 3.1(c)). The objective for hierarchical group sparse recovery is

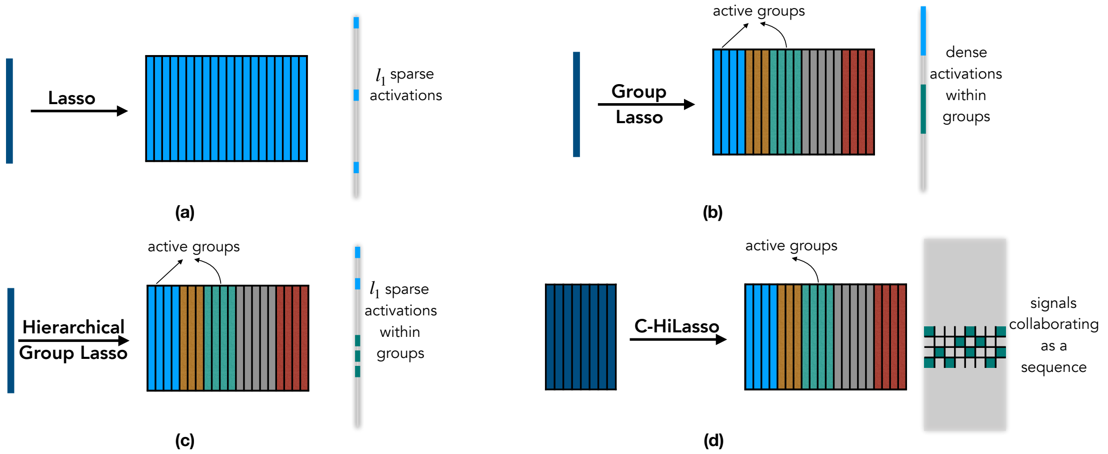

Figure 3.1: Lasso and its variants for sparse recovery: (a) Lasso, (b) group Lasso, (c) hierarchical group Lasso, and (d) collaborative hierarchical group Lasso.

expressed as

$$\min_{\mathbf{a}_t} \frac{1}{2}\|\mathbf{z}_t - \mathbf{D}\mathbf{a}_t\|_2^2 + \lambda\psi_{\mathcal{G}}(\mathbf{a}_t) + \lambda_1\|\mathbf{a}_t\|_1 \tag{3.18}$$

where $\psi_{\mathcal{G}}$ is the group Lasso regularizer as defined before.

Collaborative Hierarchical Lasso (C-HiLasso) C-HiLasso, as developed in [Sprechmann et al. (2011)], enables to incorporate the dependency among a sequence of signals $\mathbf{z}_t$'s by defining a collaborative objective function. By collaboration, we mean that the sequence of $\mathbf{z}_t$'s share the same non-zero components in the sparse representation $\mathbf{a}_t$'s. Thus, the collaborative group Lasso problem is simply formulated by extending the vector expression in (3.18) to its matrix counterpart as defined in (3.19). Here the ℓ_1-norm constraint is extended to all frames. Thus, the collaborative group lasso objective [(Sprechmann et al., 2011)] is defined as:

$$\min_{\mathbf{a}} \frac{1}{2}\|\mathbf{Z} - \mathbf{D}\mathbf{A}\|_F^2 + \lambda_2\psi_{\mathcal{G}}(\mathbf{A}) \tag{3.19}$$

By further introducing hierarchical sparsity in this formulation, the resulting objective function becomes the C-HiLasso problem as:

$$\min_{\mathbf{a}} \frac{1}{2}\|\mathbf{Z} - \mathbf{D}\mathbf{A}\|_F^2 + \lambda_2\psi_{\mathcal{G}}(\mathbf{A}) + \lambda_1\sum_{t=1}^{T}\|\mathbf{a}_t\|_1 \tag{3.20}$$

where we also seek ℓ_1-norm sparsity in individual $\mathbf{a}_t$'s (Figure 3.1(d)).

Note that in all group-based Lasso objective functions, it is assumed that the dictionary has an internal subspace-wise structure as defined by the partitioning $\mathcal{G}$.

The following subsection focuses on principal component analysis (PCA) and its relationship with dictionary learning and sparsity in general.

3.4 PCA: Links to Compressive Sensing

PCA is a commonly used technique for data analysis and dimensionality reduction. Given a dataset, it sequentially projects the data in mutually orthogonal directions of maximum variation in the data. In other words, the goal of PCA is to express the dataset in terms of a new basis set which has the following properties: 1) the basis vectors correspond to directions of maximum variance of the data in a decreasing order, and 2) the basis vectors are orthonormal to each other which results in decorrelated dimensions, thus reducing the redundancy in data representation. The basis vectors are termed as *principal components* and they can be computed by either calculating SVD of the original dataset or eigenvector decomposition of the covariance of the dataset (for more details refer [Shlens (2014)]).

Computing principal components In this thesis, we employ the eigenvector decomposition based approach for performing PCA.

Let $\tilde{\mathbf{X}} \in \mathbb{R}^{M \times N}$ be the given dataset where M denotes the dimension of data and N denotes number of data samples. First, we mean-center the data by subtracting off the mean of each dimension:

$$\mathbf{x}_i = \tilde{x}_i - \mu_{\tilde{\mathbf{X}}}, \qquad i = 1, \cdots, N$$

where $\tilde{x}_i$ is i^{th} column of $\tilde{\mathbf{X}}$, $\mu_{\tilde{\mathbf{X}}}$ is a column vector having mean of each row of $\tilde{\mathbf{X}}$, and $\mathbf{X} = [\mathbf{x}_1, \cdots, \mathbf{x}_N]$ is the mean-centered data matrix. Next, we compute the covariance of the mean-centered data as follows:

$$\mathbf{C} = \frac{1}{N-1}\mathbf{X}\mathbf{X}^\top$$

The covariance matrix $\mathbf{C}$ is factorized using eigenvalue decomposition:

$$\mathbf{C} = \mathbf{P}\mathbf{S}\mathbf{P}^\top$$

where $\mathbf{P} \in \mathbb{R}^{M \times M}$ are eigenvectors of $\mathbf{C}$ and $\mathbf{S}$ is a diagonal matrix containing the sorted eigenvalues. The columns of $\mathbf{P}$ are the principal components of $\mathbf{X}$ and the associated eigenvalues in $\mathbf{S}$ convey the amount of variance captured by each principal component.

The principal components can be used to transform the original dataset as follows:

$$\mathbf{Y} = \mathbf{P}^\top\mathbf{X} \tag{3.21}$$

where $\mathbf{Y}$ is a rotated and shifted version of $\mathbf{X}$ such that the dimensions are decorrelated and they are more meaningful in terms of capturing the dynamics of the data. Data in **Y** is said to be in the principal component space. We can project the data back to the original space simply by transforming **Y** by multiplying with **P** as:

$$\mathbf{PY} = \mathbf{PP}^{\top}\mathbf{X} = \mathbf{X}$$

where $\mathbf{PP}^{\top}$ results in an identity matrix as **P** is orthonormal.

Low-rank modeling using PCA The major utility of PCA for dimensionality reduction arises from the observation that, for most natural data, the eigenvalues associated with principal components die off to insignificantly small values very quickly. For a dataset whose intrinsic structure is low-dimensional whereas the measurement dimension M is very large, PCA would reveal that a very few eigenvalues will be large and significant. Principal components corresponding to the large eigenvalues in **S** constitute the frequent regularities in the data, whereas other components carry the high-dimensional unstructured noise. We exploit this observation as follows. We first define a low-rank projection matrix $\mathbf{D}_{LR}$ as:

$$\mathbf{D}_{LR} = \mathbf{P}_{1:l} \in \mathbb{R}^{M\times l}$$

where $\mathbf{P}_{1:l}$ is a truncation of **P** that keeps only the first l principal components and discards the variations captured by other $M - l$ components. The original data **X** can be projected on the low-dimensional principal component space as:

$$\mathbf{Y}_{LR} = \mathbf{D}_{LR}^{\top}\mathbf{X} \tag{3.22}$$

where $\mathbf{Y} \in \mathbb{R}^{l\times N}$ and we can reconstuct an approximation of the data, $\mathbf{X}_{LR}$, as

$$\mathbf{X}_{LR} = \mathbf{D}_{LR}\mathbf{Y}_{LR} = \mathbf{D}_{LR}\mathbf{D}_{LR}^{\top}\mathbf{X} \tag{3.23}$$

Here $\mathbf{X}_{LR}$ is a reconstructed version of **X** which contains the variability captured by only first l principal components. This procedure is often used to either reduce the dimension of a dataset by projecting it into the PC space using (3.22), or for denoising the data by low-rank reconstruction using (3.22) followed by (3.23).

Principal Components v/s Over-complete Dictionary The principal components matrix is a basis set with as many basis vectors as the dimensions of the space, thus rendering it a *complete* dictionary.

In terms of compressive sensing, let us assume that we have a signal $\mathbf{z} \in \mathbb{R}^{M}$ and a dictionary $\mathbf{D} \in \mathbb{R}^{M\times M}$ composed of principal components as column vectors. The goal is to represent **z** as a sparse linear combination of very few atoms of the dictionary. If we ignore the need for sparsity, then we can trivially find a dense representation using the properties of principal components as follows:

$$\mathbf{z} = \mathbf{D}\mathbf{D}^{\top}\mathbf{z} = \mathbf{D}\mathbf{a}_{dense} \tag{3.24}$$

where $\mathbf{a}_{dense} = \mathbf{D}^{\top}\mathbf{z}$.

Next, we consider the case where we expect the input signal $\mathbf{z}$ to have a sparse representation over $\mathbf{D}$. This implies that only a few atoms of the dictionary (i.e. a few principal components) would be selected for a linear combination to express $\mathbf{z}$. Due to the variance-wise ordering of the principal components in the dictionary $\mathbf{D}$, we know that a sparse representation $\mathbf{a}_{sparse}$ should have non-zero values only for the first few components, and zero elsewhere. We can construct $\mathbf{a}_{sparse}$ as a l-sparse vector by choosing first l principal components from $\mathbf{D}$ as follows:

$$\mathbf{a}_{sparse}(i) = \begin{cases} \mathbf{D}_i{}^{\top}\mathbf{z} & 1 \le i \le l \\ 0 & l < i <\le M \end{cases} \tag{3.25}$$

where $\mathbf{a}_{sparse}(i)$ is the i^{th} coefficient of $\mathbf{a}_{sparse}$ and $\mathbf{D}_i$ is the i^{th} principal component. The above operation essentially projects the original data into a l-dimensional PCA space and appends $M - l$ zeros to make $\mathbf{a}_{sparse}$ a M-dimensional vector. Now, the signal $\mathbf{z}$ can be conveniently approximated as a sparse linear combination of the dictionary atoms as:

$$\mathbf{z} \approx \mathbf{D}\mathbf{a}_{sparse} \tag{3.26}$$

The approximation here results from the fact that we discard the information stored in the last $M - l$ principal components resulting in an inexact reconstruction. Assuming that the data is actually low-dimensional, a good choice of l (i.e. the number of principal components to be used) can still reconstruct the data without much loss of information.

From the discussion above, we can conclude that PCA based low-rank reconstruction of data can be viewed as a complete dictionary based compressive sensing. However, it has some major differences with compressive sensing using overcomplete dictionaries. Firstly, the principal components are ordered as per their importance in capturing the dynamics of the data. Therefore, we always choose the same set of l components for every input signal during reconstruction. In the case of an over-complete dictionary, there is no ordering among the basis vectors. For each signal, a different set of dictionary atoms can be chosen for linear combination. Thus, an overcomplete dictionary can model data on non-linear manifolds using a union of low-dimensional subspaces. On the other hand, PCA assumes that the data lives in a linear subspace. Another significant difference is due to the orthogonality of principal components. PCA assumes that the data has a Gaussian distribution and it can not model non-orthogonally distributed data accurately. On the other hand, an overcomplete dictionary does not make any such assumptions, and it can easily fit a non-orthogonal distribution by having appropriate basis vectors [Lewicki and Sejnowski (2000)].

In this thesis, we employ both principal components and overcomplete dictionaries towards the common goal of modeling the low-dimensional subspaces in speech. While sparse recovery using overcomplete dictionaries allows modeling of non-linear subspaces, a PCA based linear modeling also stands accurate if the underlying subspaces are linear.

4 A Posterior-based Sparse Modeling Approach Towards ASR

4.1 Introduction

Speech data lies on or near non-linear manifolds [Stevens (1998); Jansen and Niyogi (2006)]. An efficient way of modeling non-linear manifolds is through overcomplete dictionaries which express the manifolds as unions of low-dimensional subspaces. Therefore, the main goal of this chapter is to fully exploit the unique low-dimensional multi-subspace structure of speech through a dictionary learning and sparse modeling approach. The major hypothesis that this thesis and particularly the current chapter are based on can be summarized in the following statement:

> *Sparse modeling using overcomplete dictionaries can accurately characterize speech data lying on non-linear manifolds; exploiting this structure appropriately can lead to improvements in ASR.*

Verifying the above hypothesis stands crucial not only for building new sparse modeling-based ASR solutions but also for modifying the existing frameworks to exploit the unique structure of speech data. Since *sparse modeling of speech using overcomplete dictionaries* is a novel direction of research, this chapter conducts a preliminary study where each aspect of the proposed framework is analyzed. Once the key elements of the proposed approach have been thoroughly investigated, we expose our technique to two relatively simpler ASR tasks- namely isolated word recognition (IWR) and connected digit recognition (CDR). The lessons learned from these tasks are used to understand the benefits and limitations of our approach. Finally, the conclusions of this chapter lead to the development of sparse and low-rank modeling approaches that are used in later chapters for improving ASR on full-fledged LVCSR tasks.

Hereafter, this chapter is organized as follows: Section 4.2 discusses the exemplar-based sparse representation approach which is central to our idea of employing dictionary based modeling of speech. Section 4.3 introduces our novel sparse modeling approach to ASR. In Section 4.4, we analyze the key aspects of our approach and evaluate it on ASR tasks. Section 4.5 summarizes the lessons learned through this study and gives directions for research presented in the next chapters.

4.2 Exemplar-based ASR

4.2.1 Background

Hidden Markov model (HMM) based modeling and exemplar-based template matching technique are the two major lateral approaches towards automatic speech recognition (ASR). In the last three decades, HMM-based approaches have been dominant because of their flexibility and their ability to be trained and generalized to unseen data. In comparison, exemplar-based techniques use labeled speech segments (called *exemplars*) directly for speech recognition. Assuming an “infinite” amount of such exemplars, as well as the “right” representation space and the “right” distance measure, “optimal” recognizers could be sought in theory [Devijver and Kittler (1982)]. It was argued in [Banko and Brill (2001)] that “The more training data used, the greater the chance that a new sample can be trivially matched to samples in the training data, thereby lessening the need for any complex reasoning that may be beneficial in the cases of sparse training data.” Exemplar-based approaches typically employ template matching techniques for speech recognition instead of a *model learning* step as done in HMM-based systems. However, there is still a necessity to train an HMM based segmentation model to generate labeled speech segments for phonetic and sub-phonetic units. A trade-off with an exemplar-based system is that it may have a huge space and time complexity. However, with the ever-increasing amount of training data, as well as the growing computational and memory resources, it has become possible to exploit the potential of exemplar-based approaches. In context of speech recognition, they have been explored extensively in [Sainath et al. (2012, 2011); Gemmeke et al. (2011); De Wachter et al. (2007)].

One of the significant approaches in exemplar-based ASR relies on exemplar-based sparse representations in which a test speech exemplar is expressed as a sparse linear combination of the exemplars in the training dataset. Thus, a large collection of training exemplars is used to capture all possible variability in the data. The core assumption of this approach is that any possible realization of the data in the test set lies in a vector space spanned by a sparse

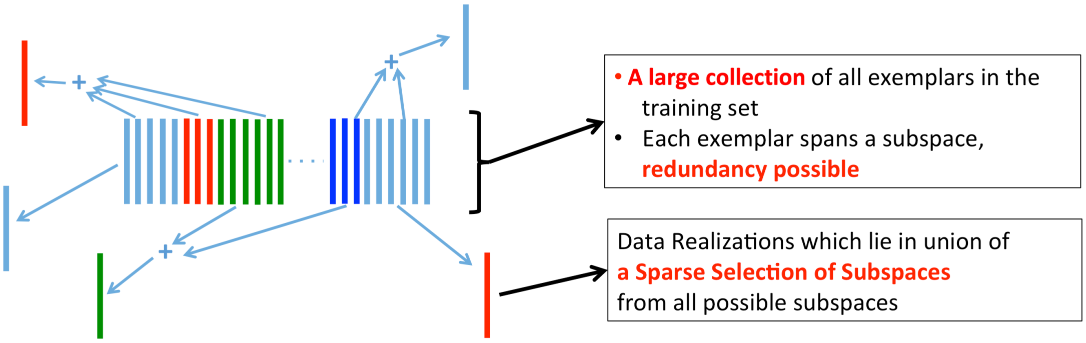
+
+
+
. . .
• A large collection of all exemplars in the training set
• Each exemplar spans a subspace, redundancy possible
Data Realizations which lie in union of a Sparse Selection of Subspaces from all possible subspaces

Figure 4.1: Exemplar-based sparse representation of speech which employs a union of low-dimensional subspace modeling.

selection of exemplars already seen in the training set i.e. the speech exemplars live in a union of low-dimensional subspaces (refer to Figure 4.1).

4.2.2 Motivations of this Work

Existing exemplar-based sparse approaches typically use spectral or cepstral features as templates [Sainath et al. (2011); Gemmeke et al. (2011, 2009); Sainath et al. (2010)]. Deviating from the previous work, this thesis proposes to use DNN-based posterior features as exemplars. There are two sources of motivations to use posterior features as exemplars.

Firstly, posterior features are computed after multiple layers of non-linear transformations using a DNN acoustic model. As they get transformed through these non-linear operations, they become increasingly robust to noise, speaker, and environmental variations. Therefore, posterior features are expected to be more accurate and better representatives of the underlying acoustic information as compared to the spectral features.

Secondly, DNNs are discriminative models trained to project the input speech features onto a probability simplex such that the outputs are the phone or subphone posterior probabilities. When posterior probability vectors are used as exemplars for dictionary learning, we can derive a very elegant probabilistic interpretation for the sparse recovery problem. In short, if the DNN outputs a phone posterior probability vector $z_t = [P(q_1|\mathbf{x}_t), \cdots, P(q_k|\mathbf{x}_t), \cdots, P(q_K|\mathbf{x}_t)]$ for the input feature $\mathbf{x}_t$, we demonstrate an efficient way of converting this vector to a sparse word posterior probabilities $P(W|\mathbf{x}_t)$ using the dictionary learning and sparse recovery approach. The underlying idea is that phone probability vectors are a compressed version of the actual high-dimensional word probability vectors which are inherently sparse. This follows from our discussion in Section 1.1. The probabilistic interpretation of posterior based dictionary learning and sparse modeling is developed in detail in Section 4.3. Note that the direct recovery of word posterior probabilities using our approach is a unique feature of this work. In comparison, existing exemplar-based sparse representation methods [Sainath et al. (2012, 2011); Gemmeke et al. (2011)] generally rely on computing state conditional data likelihoods from the sparse recovery process and use these likelihoods under HMM-based decoding for inferring the word sequence.

Another inspiration for exploring the sparse modeling based ASR approach comes from the issues faced by the previous exemplar-based sparse representation approaches in dictionary designing. Typically, prior works have used a collection of training data exemplars as dictionaries in their frameworks. We term such a dictionary as "*collection-of-exemplars*" in this thesis. It was reported in [Gemmeke et al. (2009)] that increasing the size of exemplar collection in exemplar-based ASR systems improves the ASR performance only up to a certain limit, after which the improvement becomes sub-linear. At a certain point, the additional information brought by new exemplars is insignificant as they lie close to the already existing exemplars in the collection. This suggests a need for a better procedure to design a limited-sized dictionary that can exploit the information in all available training data without the need of continually increasing the dictionary size. In this work, we propose to use the well-principled dictionary learning algorithms (refer to Section 3.2) to precisely address this need. We demonstrate experimentally in Section 4.4.3 that the dictionary learned using an appropriate algorithm can have a far smaller cardinality than the size of a collection-of-exemplars based dictionary. Moreover, such a dictionary also improves the characterization of the vector space as compared to the collection-of-exemplars dictionary.

Next, we focus on the modeling of temporal dependencies in speech features in the context of exemplar-based ASR. Existing exemplar-based sparse representation approaches follow two techniques to exploit the temporal information in speech- 1) using context-appending of adjacent features and 2) using Viterbi decoding on the data likelihoods generated from the sparse representation approach [Gemmeke et al. (2011, 2009)]. In this work, we aim to develop an all *dictionary learning plus sparse modeling* approach for ASR. Consecutive speech features generally belong to the same class out of many possible classes (e.g. the same word out of very many words). Hence, the sparse representations of consecutive exemplars should have non-zero activations for the same group of dictionary atoms which correspond to the currently active class. This structure in the sparse representations is leveraged by using a collaborative group sparse recovery approach [Sprechmann et al. (2011)]. This sparse recovery approach seeks a collaboration among the sparse codes of the consecutive exemplars. Therefore, it enables us to exploit the temporal dependencies in the input exemplars. Our idea is to demonstrate how sparse modeling can lead to capturing temporal information by offering a collaborative structure instead of enforcing only the Markovian inter-dependency. In addition, we also let our approach benefit from the existing context-appending and Viterbi decoding techniques.

To summarize, the sparse modeling framework proposed in this chapter is different from the existing exemplar-based sparse representation approaches in the following ways:

- **Exemplar type:** Spectral features are replaced by phone posterior features as they are more robust and provide a way to directly recover underlying word posterior probabilities.

- **Dictionary type:** Collection-of-exemplar based dictionaries are replaced by algorithmically learned dictionaries which can limit the dictionary size, while still being able to extract information from all the training data.

- **Temporal Modeling:** Context-appending and Markovian structure based Viterbi decoding techniques for exploiting temporal dependencies in speech are complemented by a collaborative hierarchical sparse recovery approach.

- **Word Inference:** HMM-based decoding of the best phone sequence to infer the underlying word sequence is contrasted with the direct word inference which is enabled by the sparse modeling of posterior exemplars in a sliding window-based fashion.

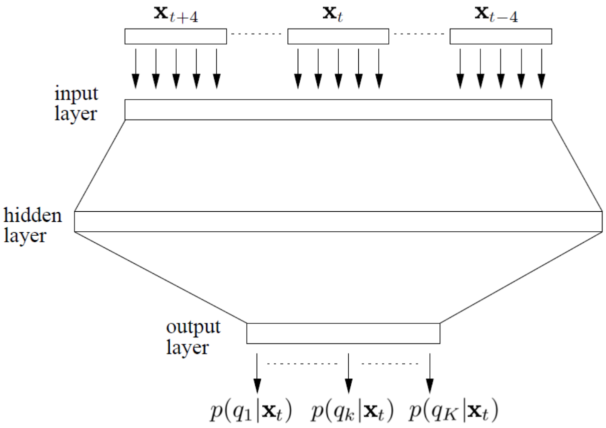

Figure 4.2: Posterior features are extracted using a neural network taking the acoustic features as input. Context of c=4 frames is shown here at input.

The next section introduces the novel compressive sensing and sparse modeling based approach to ASR.

4.3 A Compressive Sensing Perspective to Posterior-based Sparse Modeling

In this section, we formulate the traditional HMM-based ASR approach as a sparse modeling task where isolated words or word sequences can be inferred using sparse recovery over learned dictionaries.

4.3.1 Formalism

Speech recognition aims to recover the sequence of words from the observed acoustic features. The space of sub-word[1] units is low-dimensional and comparatively denser than the space of words (as explained in Section 1.1). Let K be the number of sub-word units in a DNN based ASR system and L denotes the size of the word vocabulary, then $L \gg K$. Given the compressed sub-word observation sequence, the goal of speech recognition can be defined as reconstructing the high-dimensional word observation sequence.

The sub-word information in an utterance can be extracted using a DNN-based acoustic model in terms of the framewise phone posterior probability vectors. The setup for extracting the phone posterior features is illustrated in Figure 4.2 [Aradilla and Bourlard (2009)]. Acoustic features, e.g. MFCC vectors, with their first and second order derivatives, are computed over a sliding window of 25ms with a shift of 10ms. A DNN takes as input a context of these features and generates the phone posterior probabilities. In a traditional hybrid DNN-HMM ASR

[1] In this study, we consider phones as the observed sub-word units. In principle, the sub-word units can also corresponded to other entities like phone-HMM states [Bahaadini et al. (2014)] or syllables.

framework, the output phone probability vectors can be directly processed by a decoder for outputting the word sequences. In this thesis, we consider the posterior features as new kind of acoustic features for further modeling. Therefore, the DNN acoustic model acts as a feature extractor. Note that the output layer includes an additional unit for representing the silence/pause along with the other phones.

While the phone posterior observations live in $\mathbb{R}^K$, the equivalent word posteriors would live in $\mathbb{R}^L$ - a very high dimensional space compared to $\mathbb{R}^K$. The key idea is that the representation of linguistic information in the word representation space is highly sparse. This is because only a few words are spoken in an utterance and typically only one word is expected to be spoken during a very short segment of speech. The word posterior probabilities should have a *nil* probability for all the words but a few. Therefore, we propose to cast the speech recognition problem as sparse reconstruction of word posterior probabilities given the compressed (low-dimensional) sub-word posterior probabilities.

More precisely, we consider phones as the sub-words units modeled by the DNN. We define the set of phones as $\{q_k\}_{k=1}^K$. Given an input of context-appended feature vectors $[\mathbf{x}_{t-c},\cdots,\mathbf{x}_t,\cdots,\mathbf{x}_{t+c}]$ at time t, the posterior probability $p(q_k|\mathbf{x}_t)$ is estimated at the DNN output where q_k is associated with the k^{th} phone. The phone posterior probability relates to the word level posterior probabilities through a marginalization over L latent word variables w_l as follows:

$$p(q_k|\mathbf{x}_t) = \sum_{l=1}^{L} p(q_k, w_l|\mathbf{x}_t) = \sum_{l=1}^{L} p(q_k|w_l,\mathbf{x}_t)p(w_l|\mathbf{x}_t) = \sum_{l=1}^{L} p(q_k|w_l)p(w_l|\mathbf{x}_t), \tag{4.1}$$

where w_l denotes the l^{th} word in the vocabulary and the last equality holds due to the assumed conditional independence of the acoustic observation $\mathbf{x}_t$ and the current phone q_k given a super-phone lexical unit such as the word w_l. Dropping the dependence of the phone posterior probability on the current acoustic observation $\mathbf{x}_t$ enables us to define a static dictionary below which comprises of word-conditional phone posterior probabilities without depending on the current acoustic observation being modeled. However, we lose the information about how phone probabilities evolve within the words. As discussed later, this issue is taken care of by modeling each word using a word-based dictionary.

The marginalization in (4.1), when expanded and expressed for all the phones $\{q_k\}_{k=1}^K$ together, leads to the following matrix multiplication equation:

$$\underbrace{\begin{bmatrix} p(q_1|\mathbf{x}_t) \\ p(q_2|\mathbf{x}_t) \\ \vdots \\ p(q_K|\mathbf{x}_t) \end{bmatrix}}_{\mathbf{z}_t} = \underbrace{\begin{bmatrix} p(q_1|w_1) & \cdots & p(q_1|w_l) & \cdots & p(q_1|w_L) \\ p(q_2|w_1) & \cdots & p(q_2|w_l) & \cdots & p(q_2|w_L) \\ \vdots & \vdots & & & \\ p(q_K|w_1) & \cdots & p(q_K|w_l) & \cdots & p(q_K|w_L) \end{bmatrix}}_{\text{Dictionary: }\mathbf{D}=[\mathbf{d}_1\ldots\mathbf{d}_l\ldots\mathbf{d}_L]} \times \underbrace{\begin{bmatrix} p(w_1|\mathbf{x}_t) \\ \vdots \\ p(w_l|\mathbf{x}_t) \\ \vdots \\ p(w_L|\mathbf{x}_t) \end{bmatrix}}_{\mathbf{a}_t} \tag{4.2}$$

where we consider the phone posterior feature $\left[p(q_1|\mathbf{x}_t),\cdots,p(q_K|\mathbf{x}_t)\right]^\top$ as an observation $\mathbf{z}_t$. The matrix on RHS of the equation naturally takes form of an overcomplete dictionary $\mathbf{D}$ such that the atoms are exemplars obtained by conditioning the phone posterior probabilities on different words w_l's. Each column of this matrix is word-specific and defined as:

$$\mathbf{d}_l = [p(q_1|w_l)\cdots p(q_k|w_l)\cdots p(q_K|w_l)]^\top$$

Designing the dictionary in this manner, we now view the column vector on RHS as $\mathbf{a}_t$, a word posterior probability based sparse representation where:

$$\mathbf{a}_t = \left[p(w_1|\mathbf{x}_t),\cdots,p(w_l|\mathbf{x}_t),\cdots,p(w_L|\mathbf{x}_t)\right]^\top \tag{4.3}$$

Equation (4.1) can be expressed in the form of the ℓ_1-norm based sparse recovery equation (3.4) as:

$$\min \|\mathbf{a}_t\|_1 \quad \text{subject to} \quad \mathbf{z}_t = \mathbf{D}\,\mathbf{a}_t \tag{4.4}$$

Based on (4.1), if $\mathbf{z}_t$ and $\mathbf{D}$ are composed of posterior features, $\mathbf{a}_t$ is also a posterior vector. The hidden variable w_l does not necessarily need to be associated with a word only; it can be interpreted as any other linguistic unit. In fact, (4.2) demonstrates how a posterior feature $\mathbf{z}_t$ for a given linguistic class can be used for recovering the posterior probabilities of a linguistically super-class as a high-dimensional sparse vector $\mathbf{a}_t$. The dictionary $\mathbf{D}$ which enables this process needs to be constructed using appropriate exemplars i.e. representatives of the associated super class. For instance, phone or phone-HMM state posteriors can be converted using (4.1) to word posterior probabilities which are high-dimensional and sparse.

In practice, construction of the dictionary as described in (4.2) requires modeling the subspace of each word using the word-conditional phone posterior probabilities. To characterize the posterior probabilities of each word, we learn word-specific dictionaries such that each column $\mathbf{d}_l$ of the dictionary in (4.2) has a sparse representation stated as

$$\underbrace{\begin{bmatrix} p(q_1|w_l) \\ p(q_2|w_l) \\ \vdots \\ p(q_K|w_l) \end{bmatrix}}_{\mathbf{d}_l} = \underbrace{\begin{bmatrix} p(q_1|sw_1^{w_l}) & \cdots & p(q_1|sw_s^{w_l}) & \cdots & p(q_1|sw_{S_{w_l}}^{w_l}) \\ p(q_2|sw_1^{w_l}) & \cdots & p(q_2|sw_s^{w_l}) & \cdots & p(q_2|sw_{S_{w_l}}^{w_l}) \\ \vdots & \vdots & & & \\ p(q_K|sw_1^{w_l}) & \cdots & p(q_K|sw_s^{w_l}) & \cdots & p(q_K|sw_{S_{w_l}}^{w_l}) \end{bmatrix}}_{\text{Word manifold modeling dictionary:}\mathbf{D}_{w_l}} \times \underbrace{\begin{bmatrix} p(sw_1^{w_l}|w_l) \\ \vdots \\ p(sw_s^{w_l}|w_l) \\ \vdots \\ p(sw_{S_{w_l}}^{w_l}|w_l) \end{bmatrix}}_{\mathbf{a}_{w_l}} \tag{4.5}$$

where $sw_s^{w_l}$ denotes the s^{th} basis vector required to span the subspace of the word w_l and S_{w_l} is the total number of basis vectors in $\mathbf{D}_{w_l}$. Each basis vector can also be interpreted as spanning the subspace of some sub-segment of the word, for instance the beginning, middle or end of the word. Therefore, such a multicolumn word-specific dictionary takes care of how the phone probabilities change in different parts of the word. If the dictionary $\mathbf{D}_{w_l}$ is overcomplete, then it models the word w_l as a union of low-dimensional subspaces. Consequently, it can be used for sparse recovery of the vector $\mathbf{a}_{w_l}$ which contains the posterior probabilities of various sub-segments of the w_l. The overcompleteness of $\mathbf{D}_{w_l}$ also allows it to capture the variations in the pronunciation of word w_l. Equations (4.2) and(4.5) lead us to a very intuitive and natural representation of speech in terms of posterior features and word-to-subword hierarchical dictionaries. Therefore, the phone posterior-based sparse modeling dictionary is obtained as:

$$\mathbf{D} = [\mathbf{D}_{w_1} \cdots \mathbf{D}_{w_l} \cdots \mathbf{D}_{w_L}] \tag{4.6}$$

The dictionary $\mathbf{D}$, has an internal partitioning defined by the boundaries of individual subdictionaries $\mathbf{D}_{w_l}$. Ideally, an input posterior feature $\mathbf{z}_t$ belonging to a realization of word w_l, when sparse coded using the dictionary above, will have a sparse representation $\mathbf{a}_t$ such that only the atoms corresponding to the subdictionary $\mathbf{D}_{w_l}$, henceforth denoted as $\mathbf{a}_t^{w_l}$, will have non-zero values. This sparse recovery process can be visualized as:

$$\mathbf{z}_t = \mathbf{D}\mathbf{a}_t = [\mathbf{D}_{w_1} \cdots \mathbf{D}_{w_l} \cdots \mathbf{D}_{w_L}] \times \begin{bmatrix} \mathbf{a}_t^{w_1} \\ \vdots \\ \mathbf{a}_t^{w_l} \\ \vdots \\ \mathbf{a}_t^{w_L} \end{bmatrix} \tag{4.7}$$

where each subvector $\mathbf{a}_t^{w_l}$ can be expressed using $\mathbf{a}^{w_l}$ from (4.5) as:

$$\mathbf{a}_t^{w_l} = \begin{bmatrix} p(sw_1^{w_l}|\mathbf{x}_t) \\ \vdots \\ p(sw_s^{w_l}|\mathbf{x}_t) \\ \vdots \\ p(sw_{S_{w_l}}^{w_l}|\mathbf{x}_t) \end{bmatrix} = \begin{bmatrix} p(sw_1^{w_l}|w_l) \times p(w_l|\mathbf{x}_t) \\ \vdots \\ p(sw_s^{w_l}|w_l) \times p(w_l|\mathbf{x}_t) \\ \vdots \\ p(sw_{S_{w_l}}^{w_l}|w_l) \times p(w_l|\mathbf{x}_t) \end{bmatrix} = \mathbf{a}^{w_l} p(w_l|\mathbf{x}_t) \tag{4.8}$$

The sparse representation $\mathbf{a}_t$'s can be directly used as features for modeling state-specific multinomial distributions under the KL-HMM framework as described in Section 2.2.6. This approach has been successfuly explored in [Bahaadini et al. (2014)]. In the present thesis, we focus on posterior features only to devise novel sparse modeling based ASR paradigms, and do not consider the KL-HMM approach. In the following sections, we describe the application of the posterior-based sparse modeling formalism for automatic speech recognition tasks.

4.3.2 Isolated Word Recognition

Given a posterior feature $\mathbf{z}_t$ and the dictionary $\mathbf{D}$ defined in (4.6), we first obtain the sparse representation $\mathbf{a}_t$ using sparse recovery methods[2] described in Section 3.3. The coefficients of the sparse code $\mathbf{a}_t$ corresponding to the word-specific dictionary $\mathbf{D}_{w_l}$ are denoted by $\mathbf{a}_t^{w_l}$ as expressed in (4.8). Therefore, the posterior probability $p(w_l|\mathbf{x}_t)$ for word w_l is estimated as

$$p(w_l|\mathbf{x}_t) := \|\mathbf{a}_t^{w_l}\|_1 \tag{4.9}$$

assuming a union of disjoint events due to sparse recovery over the multi-word dictionary $\mathbf{D}$. After sparse recovery, the sparse representation $\mathbf{a}_t$ is normalized to sum 1 so that the wordwise ℓ_1-norm terms $\|\mathbf{a}_t^{w_l}\|_1$ lie in the probability simplex and represent word posterior probabilities conditioned on the acoustic observation $\mathbf{x}_t$.

Consider a sequence of posterior features $\mathbf{Z}$ estimated using a DNN acoustic model from acoustic features $\mathbf{X}$. A sequence of word posterior sparse representations $\mathbf{A}$ is obtained using the sparse recovery algorithms on $\mathbf{Z}$. Using the frame level word-posterior probabilities $p(w_l|\mathbf{x}_t)$'s from equation (4.9), the maximum-a-posteriori word recognition can be obtained for $\mathbf{X}$ through

$$w_{\text{recognized}} := \arg\max_{w_l} p(w_l|\mathbf{X}) = \arg\max_{w_l} \prod_{t=1}^{T} p(w_l|\mathbf{x}_t) \tag{4.10}$$

where T indicates the length of the test utterance in IWR. In the last step in (4.10), we assume framewise independence to use the product rule of probability to ensure the continued realization of word w_l from time frame $t = 1$ to $t = T$.

For IWR tasks, another approach can be used to compute the framewise word posterior probability $p(w_l|\mathbf{x}_t)$ in (4.9). Sparse recovery can be done using word-specific dictionaries $\mathbf{D}_{w_l}$ exploiting the prior knowledge of the internal partitioning of the multi-word dictionary $\mathbf{D}$. In this case, we obtain the sparse codes $\mathbf{a}_t^{w_l}$ for each $\mathbf{D}_{w_l}$ directly, instead of $\mathbf{a}_t$. This approach leads to word-wise sparse recovery with a caveat that the word posterior probabilities stated in (4.2) as $\mathbf{a}_t$ can not be directly obtained. This is because for each word w_l, a sparse representation $\mathbf{a}_t^{w_l}$ is computed through an independent non-competing sparse coding process using dictionary $\mathbf{D}_{w_l}$. Therefore, $\mathbf{a}_t^{w_l}$'s can not be simply vertically concatenated to get $\mathbf{a}_t$. Word recognition decisions for a sequence of posterior features $\mathbf{Z}$ can now be made using minimization of least-square reconstruction error over all dictionaries $\mathbf{D}_{w_l}$. This reconstruction error has been successfully applied for classification task [Wright et al. (2009)] and linear predictive HMM [Kenny et al. (1990)]. If the reconstruction error for sparse recovery of $\mathbf{z}_t$ using dictionary $\mathbf{D}_{w_l}$ is denoted by $e_t^{w_l}$ such that:

$$e_t^{w_l} = \|\mathbf{z}_t - \mathbf{D}_{w_l}\mathbf{a}_t^{w_l}\|_2^2 \tag{4.11}$$

[2] To obtain the non-negative sparse word posterior probabilities, the algorithm are revised to project the non-zero coefficients onto the non-negative orthant. These are separable constraints on the coordinates so it does not compromise the convergence of the method. Lastly, they are ℓ_1 normalized.

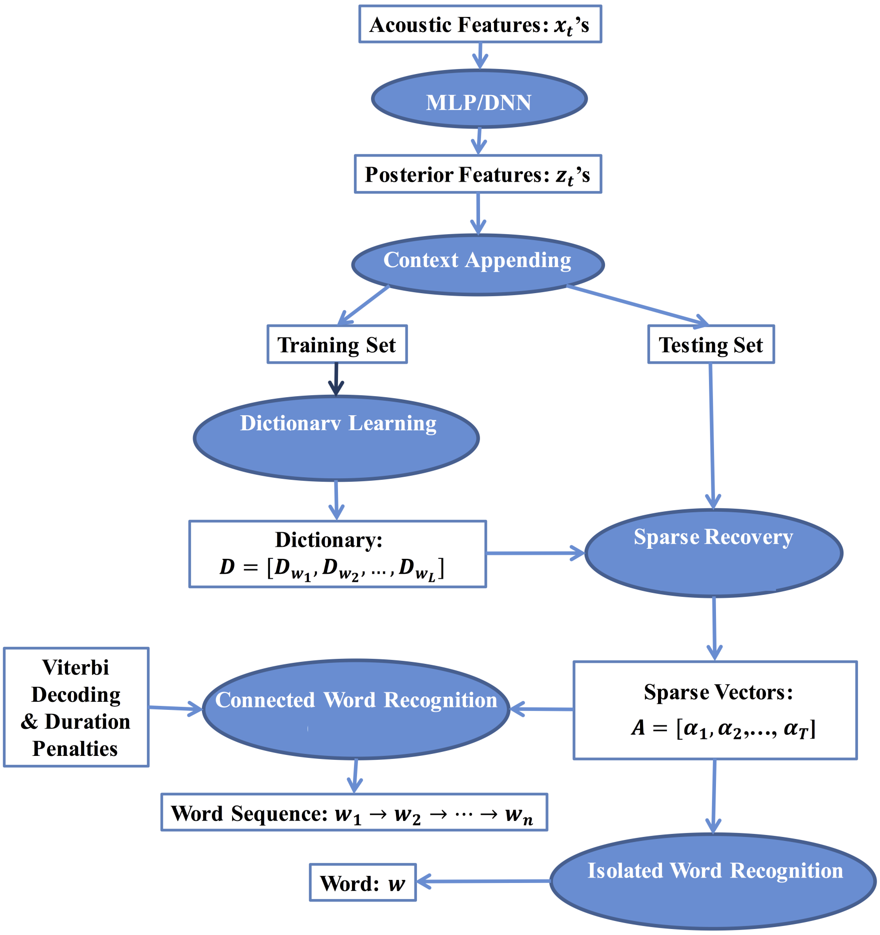
Acoustic Features: x_t's
MLP/DNN
Posterior Features: z_t's
Context Appending
Training Set
Testing Set
Dictionarv Learning
Dictionary: $D = [D_{w_1}, D_{w_2}, \dots, D_{w_L}]$
Sparse Recovery
Viterbi Decoding & Duration Penalties
Connected Word Recognition
Sparse Vectors: $A = [\alpha_1, \alpha_2, \dots, \alpha_T]$
Word Sequence: $w_1 \to w_2 \to \cdots \to w_n$
Isolated Word Recognition
Word: w

Figure 4.3: Flowchart of the ASR framework using the proposed posterior-based sparse modeling approach.

The word recognition for the complete sequence **Z** can again be done using (4.10). By assuming a Gaussian noise with zero mean and unit variance, the product of probabilities in (4.10) can be expressed as a sum of squared errors in the following way:

$$w_{\text{recognized}} := \arg\min_{w_l} \sum_{t=1}^{T} e_t^{w_l} \tag{4.12}$$

Equation (4.10) and (4.12) directly output the word posterior probabilities and do not rely on data likelihoods and word prior probabilities as required under a Bayes decision rule based approach.

4.3.3 Continuous Speech Recognition

The difficulty in continuous speech recognition is rooted in the unknown word boundaries. Hence, T frames may encapsulate several word classes with pauses in between. We learn a separate dictionary for the pause/silence class. The pause state is also defined in the output layer of the neural network. However, the neural network is not perfect in pause detection and learning a pause dictionary is beneficial for sparse modeling of continuous speech. For continuous speech recognition, we can either employ a sliding window based analysis or a C-HiLasso based approach. These techniques are discussed below.

4.3.3.1 Word Dictionary based Sparse Recovery

Similar to IWR, sparse recovery can be done using word-specific dictionaries $\mathbf{D}_{w_l}$. We just need to convert the reconstruction errors $e_t^{w_l}$ into *empirical* word posterior probabilities. Let M denote the maximum value of the error $e_t^{w_l}$ over all words w_l. The empirical word posterior probabilities are then obtained through

$$p(w_l|\mathbf{x}_t) := \frac{M - e_t^{w_l}}{\sum_{l=1}^{L}(M - e_t^{w_l})} \tag{4.13}$$

Since the end goal is continuous speech recognition using a sliding window, we compute such empirical word probabilities for each slide of the window and use them under a dynamic programming based Viterbi decoder with appropriate word transition probabilities and duration penalties to decode a word sequence.

4.3.3.2 C-HiLasso based Sparse Recovery

When a sequence (matrix) of consecutive posterior feature vectors $\mathbf{Z} = [\mathbf{z}_1, \ldots, \mathbf{z}_{t_1}, \ldots, \mathbf{z}_{t_2}, \ldots, \mathbf{z}_T]$, extracted from a speech utterance, is sparse coded using dictionary $\mathbf{D}$ (4.2), it yields a sparse representation matrix $\mathbf{A} = [\mathbf{a}_1, \ldots, \mathbf{a}_{t_1}, \ldots, \mathbf{a}_{t_2}, \ldots, \mathbf{a}_T]$ that exhibits a *collaborative hierarchical group sparsity structure* among its coefficients. This has been shown in Figure 4.4. Consecutive posterior feature vectors $[\mathbf{z}_{t_1}, \ldots, \mathbf{z}_{t_2}]$ that belong to occurrence of the same word w_l excite only those atoms of dictionary $\mathbf{D}$ that correspond to the word-specific subdictionary $\mathbf{D}_{w_l}$. Thus, they *collaborate* from time instant t_1 to t_2 to activate a higher level group $\mathbf{a}_t^{w_l}$ corresponding to $\mathbf{D}_{w_l}$. Moreover, the sparse representation $\mathbf{a}_t$ is sparse at two *hierarchical* levels: (i) in terms of the number of groups $\mathbf{a}_t^{w_l}$ activated (which is equal to one when only one word is spoken at a given time) and (ii) in terms of the non-zero coefficients of $\mathbf{a}_t^{w_l}$. This collaborative hierarchical structure is leveraged by using the C-HiLasso algorithm for the objective function formulated in (3.20) [Sprechmann et al. (2011)] and depicted in Figure 4.4. It may be noted that C-HiLasso forces activation of the same group (or groups) for all the posterior feature vectors that are being sparse coded together. Thus, an utterance with a sequence of words spoken one after another has to be sparse coded using C-HiLasso in a sliding window fashion. This ensures activation of a single group (word) in each position of the sliding window (more details in

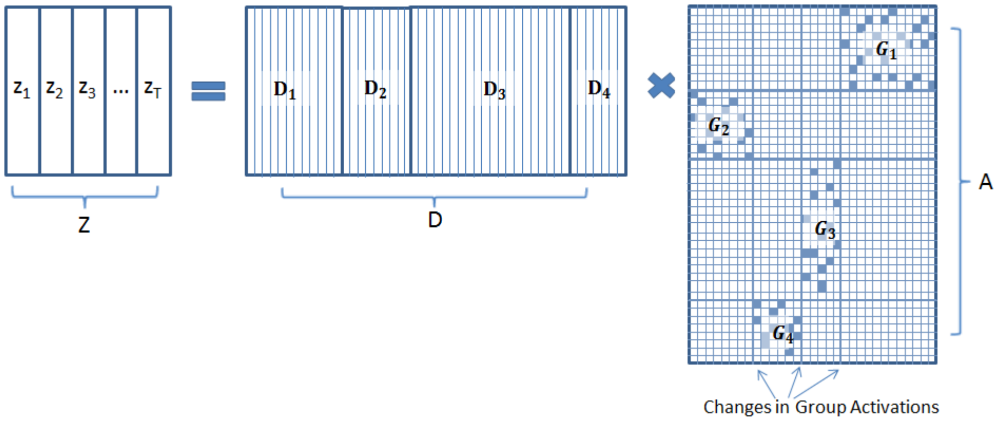

Figure 4.4: Given a sequence of acoustic features **Z**, the sparse representation matrix **A** will have a sparse block structure associated to the word-specific dictionaries ($\mathbf{D}_{w_l}$)'s where the inner block coefficients are sparse as well. This collaborative hierarchical sparsity structure can be exploited using C-HiLasso algorithm [Sprechmann et al. (2011)] based on the sparse recovery objective expressed in (3.20).

Sections 4.3.3 and 4.4.7).

Given test utterance $\mathbf{Z}^{\text{test}}$, a sliding window of appropriate length T' can be used to process a collection of frames $\mathbf{Z}^{\text{test}}_{t...t+T'-1}$ using C-HiLasso, where the window contains frames from time instant t to $t+T'-1$. The window length T' should be short enough to capture only a single word and long enough to group the sequence of frames into a single consistent class. Hence, the choice of T' is not trivial and should be learned during the recognition task. It may be noted that the collaborative hierarchical Lasso requires the full dictionary **D** for computing sparse representation $\mathbf{a}_t$. The word posterior probabilities from (4.9) are then simply used under a Viterbi decoder to obtain the word sequences.

Figure 4.3 illustrates the flow chart of the proposed posterior-based sparse modeling approach for speech recognition.

4.4 Key Aspects of the Sparse Modeling Approach To ASR

In this section, we present a series of experiments for empirical evaluation and analysis of various key aspects of the proposed sparse modeling approach. The experiments are devised to-

- evaluate different computational methods for dictionary learning and sparse recovery (Section 4.4.2),
- compare algorithmically learned dictionary with collection-of-exemplars based dictionary(Section 4.4.3), and
- provide empirical insights into structured sparsity and context modeling (Section 4.4.4

and Section 4.4.5),

- compare the union of subspace model with DTW and HMM models (Section 4.4.6).

Finally, we study the performance of the proposed posterior-based sparse modeling approach for exemplar-based automatic speech recognition in Section 4.4.7.

4.4.1 Databases and Features

Two databases are used for experiments in this section: (1) PhoneBook speech corpus [Pitrelli et al. (1995)] for IWR task and (2) Digits database, a subset of Numbers 95 [(Cole et al., 1995)], for connected word recognition task. We perform two sets of experiments with PhoneBook for IWR task - an easier 75 words vocabulary task and a more challenging 600 words vocabulary task (refer Section 2.3.1). Each word has around 11-12 utterances from different speakers, out of which we use 4 for learning dictionaries and the rest for testing. Due to different speakers in training and testing conditions, this setup is expected to generalize on speech data from unseen speakers. The setup is similar to the experiments in [Soldo et al. (2011)]. Since the amount of training data is small for PhoneBook, each dictionary is initialized with one of the four templates in the training data, and the rest are used for dictionary learning. Hence, the dictionary size is 25% of the size of training data.

For connected word recognition, we work with Digits database (refer Section 2.3.2). For training word or digit specific dictionaries, the digit sequences in the training data are split into digit-specific utterances where the digit-based segmentation is obtained using GMM-HMM based forced alignment. We use a concatenation of 100 such utterances for initializing the word-specific dictionaries, and the rest of the training data utterances for each word are used for learning the respective dictionaries. As there are ~3000 training exemplars per word, the dictionary size is ~3% of the size of training data.

For both databases, the features are extracted in the following manner. We compute 13-dimensional MFCC feature vectors over a sliding window of 25 ms with a shift of 10 ms. Along with their first and second order derivatives, the MFCC features are mapped to the phone posterior probabilities using a DNN acoustic model as explained in Section 2.2.5. A context of 9 frames is used at the input of the DNN. Phone posteriors at the output of the DNNs correspond to monophone units for PhoneBook and Digits database which are 42 and 27 respectively (including a phone for silence class in each case). For Phonebook, the DNN architecture has 351 (= 13 × 3 × 9) input units, 600 hidden units, and 42 output units. For Digits, the DNN architecture has 351 inputs units, 1000 hidden units and 27 output units. The networks have sigmoid non-linearities and they are trained using cross entropy loss minimization criterion.

4.4.2 Comparison of Dictionary Learning and Sparse Recovery Algorithms

Dictionary learning and sparse recovery are the two pillars of our sparse modeling framework. We conduct our experiments using the state-of-the-art dictionary learning and sparse recovery techniques described in Chapter 3 to learn dictionaries from posterior-based exemplars and obtain word posterior sparse representations for speech recognition. The comparison of various algorithms is done through their evaluation PhoneBook 75-vocabulary IWR task. The results are listed in Table 4.1. In this study we consider the online dictionary learning [Mairal et al. (2010)] and KSVD [Aharon et al. (2006)] algorithms for learning the dictionary of context-appended posterior exemplars ($c = 20$) (Section 4.4.5). The LARS implementation of LASSO [Efron et al. (2004)] and OMP [Tropp and Wright (2010)] algorithms are used for reconstruction of word posterior sparse representation. The word recognition is obtained through (4.12).

The best recognition performance is obtained using the online dictionary learning algorithm with Lasso sparse recovery with an accuracy of 97.2%. The online dictionary learning algorithm has been found to work fast with LARS Lasso [Efron et al. (2004)] with higher accuracies. K-SVD performs poorly in comparison. One of the weaknesses of K-SVD is that the algorithm can get stuck in local minima because of the non-convexity of the problem [Aharon et al. (2006)]. Hereafter, we use the online algorithm for dictionary learning and LARS Lasso implementation for sparse recovery in all the experiments. An optimization of sparsity controlling parameter λ resulted in values 0.1 and 0.2 for PhoneBook and Digits database, respectively.

	Lasso	OMP
Online Algorithm	**97.2**	93.5
KSVD	55.8	88.9

Table 4.1: Word recognition rate (%) on PhoneBook 75-vocabulary dataset using different computational methods to dictionary learning and sparse recovery.

4.4.3 Dictionary Learning vs Collection of Exemplars

In this section, we conduct experiments to compare dictionaries learned through principled dictionary learning algorithms discussed in Section 3.2 versus the collection-of-exemplars based dictionaries for posterior-based sparse representation.

In IWR experiment on PhoneBook 75-vocabulary dataset, the posterior exemplars from a single utterance of each word are used as a warm start for dictionary initialization. The posterior exemplars from the remaining three utterances in the training set are then used for updating the dictionary columns using the online dictionary learning algorithm. Alternatively, the posterior exemplars from all the four training utterances are concatenated to form a dictionary for sparse representation. This is the *collection-of-exemplars* dictionary. Similarly, for CDR on Digits database, we can either learn word-specific dictionaries, or we directly represent each word using all training exemplars (Gemmeke et al., 2011). The results are listed in Table 4.2. We observe that the dictionary learning procedure is more effective than the collection-of-exemplars model. It benefits from the abundance of the training data and enables us to keep the dimensionality of the dictionary small while simultaneously improving the performance. The size of training data in PhoneBook is small as we only have four training exemplars per word. In this case, the dimension of dictionary exemplars (number of learned atoms) is 25% of the full training set. The amount of training data in Digits corpus is larger than PhoneBook, and the dimension of dictionary exemplars is ~3% of the full training set.

Approach	PhoneBook	Digits
Dictionary Learning	**97.2**	**85.4**
Collection-of-exemplars	97.0	78.6

Table 4.2: Comparing the speech recognition accuracy (in %, 100−WER) on PhoneBook and Digits database using dictionary learning versus collection-of-exemplars approach.

4.4.4 Structured Sparsity

The high dimensional sparse representations obtained using (4.7) exhibit some structures that can be exploited for speech recognition. We discuss these structures below.

4.4.4.1 Sequencing pattern

We demonstrate that controlled initialization of the word-level dictionaries $\mathbf{D}_{w_l}$ as defined in (4.5) and (4.6) enables preserving the temporal information during the learning procedure. Using PhoneBook data, the word-specific dictionary is initialized with an utterance of the word. Dictionary learning explained in Section 3.2 leads to the atoms being updated such that the temporal evolution of the word is embedded in the sequence of the atoms. We can verify this hypothesis from the sparse representation **A** of a sequence of posterior features **Z** using the word-specific dictionaries. Figure 4.5 illustrates the sparse representations **A** obtained for a sequence of the posterior features of the word '*Accumulation*'. This text utterance is sparse coded using the the correct dictionary, i.e., $\mathbf{D}_{\text{Accumulation}}$ as well as using a wrong dictionary, e.g. $\mathbf{D}_{\text{Alleviatory}}$. The sequence pattern is exhibited as a left-to-right *descending ladder* activations when $\mathbf{D}_{\text{Accumulation}}$ is used for sparse recovery. On the other hand, the sequencing pattern is distorted when the wrong dictionary $\mathbf{D}_{\text{Alleviatory}}$ is used. The word utterance and corresponding dictionaries in this analysis are taken from the setup on PhoneBook database.

The sequencing pattern can be justified using (4.5): each of the dictionary column behaves like the subword probabilities $p(q_k|sw_s^{w_l})$ which are evolving with time. As a sequence of subwords $sw_s^{w_l}$ comprise the word w_l, the subword sparse representations get activated for coefficients corresponding to various subword probabilities $p(sw_s^{w_l}|\mathbf{x}_t)$ sequentially, thus exhibiting a ladder pattern. The sequence pattern encourages us to look for mechanisms of incorporating the temporal information in sparse recovery process. One approach is through the use of structured sparse recovery based on C-HiLasso (3.20) that is studied below.

4.4.4.2 Collaborative Hierarchical Sparsity

The existence of collaborative hierarchical sparsity in word posterior probability based sparse representations is discussed in Section 4.3.3.2. We verify this intuition using C-HiLasso ob-

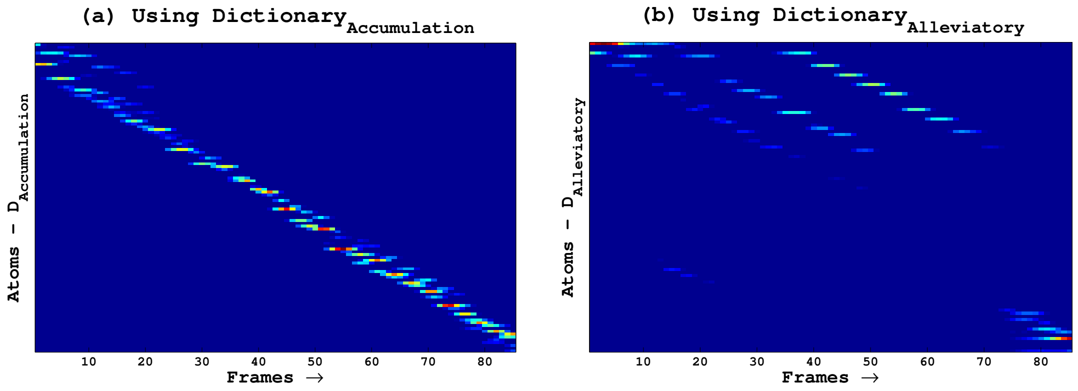

Figure 4.5: Sparse representation of the word "Accumulation" when the dictionary used for sparse recovery corresponds to (a) $\mathbf{D}_{\text{Accumulation}}$ and (b) $\mathbf{D}_{\text{Alleviatory}}$. In case of the correct word dictionary (a), all the dictionary atoms are activated whereas only a few atoms get non-zero activations when an incorrect dictionary (b) is used for sparse modeling. The sequencing pattern observed in the sparse representation obtained from the correct word dictionary (a) is due to the correct temporal ordering of the atoms within the dictionaries.

jective function ((3.20) in Section 3.3.2) to obtain the word posteriors for a connected digit sequence. A sample utterance is taken from Digits database. Figure 4.7 demonstrates the sparse representation of this test digit sequence *0-2-1-4-4* using C-HiLasso when it is sparse coded using the complete multi-word dictionary **D**. The results are contrasted with Figure 4.6 where the collaborative hierarchical sparsity structure is ignored during sparse recovery.

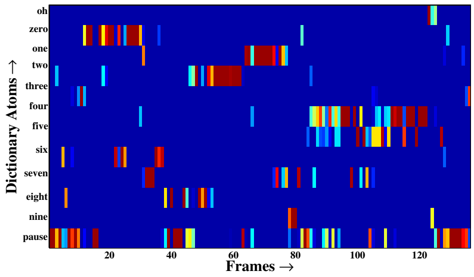

Figure 4.6: Sparse representation of connected digit sequence *0-2-1-4-4* using full dictionary (**D**) and Lasso sparse recovery.

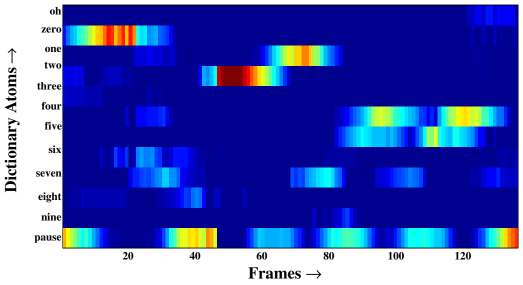

Figure 4.7: Sparse representation of connected digit sequence *0-2-1-4-4* using full dictionary **D** and C-HiLasso sparse recovery.

The sparse representations in these figures demonstrate that exploiting the structured sparsity of the sparse coefficients leads to better discrimination of the individual classes. C-HiLasso based sparse recovery uses a sliding window of 3 frames with a shift of 1 frame, whereas traditional Lasso based recovery uses frame-level posterior features. An alternative strategy to exploit the temporal information is by using the context-appended posterior features. This technique is discussed next.

4.4.5 Context Size Optimization

To incorporate the contextual information associated with the temporal evolution of posterior features, one effective way is to append the neighboring frames to the current posterior vector. More specifically, for a context size of c, a frame-level posterior feature $\mathbf{z}_t \in \mathbb{R}^K$ is mapped to a segmental feature $\tilde{\mathbf{z}}_t \in \mathbb{R}^{K(2c+1)}$ by appending c features on its right and left accordingly. This technique was successfully applied in [Bahaadini et al. (2014)]. Learning a dictionary this way improves the effectiveness of word-specific sub-dictionaries significantly, as shown below.

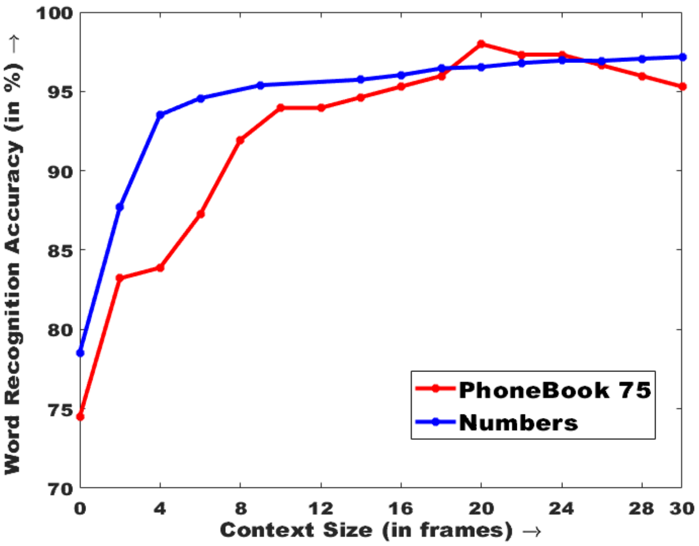

Figure 4.8: Optimization of the context size that is appended to the posterior exemplars for improving isolated word recognition using word-specific dictionaries for sparse recovery. The best performance is achieved when a context of 20 frames for PhoneBook database and a context of 30 frames for Digits Corpus is used.

Figure 4.8 illustrates the improvement in IWR word recognition accuracy for different context sizes using PhoneBook and Digits database. A context size of $c = 20$ frames was found to be optimal for PhoneBook corpus and the performance stagnates for larger values of c. This context size is applied for the rest of the ASR experiments on PhoneBook data. Similar obeservations are made on Digits database where the recognition performance initially improves with increase in the context size and then stagnates. The average word length in Digits database is ~ 30 frames. Therefore, longer context indicates that each context-appended posterior vector might represent one complete word utterance. In this case, the sparse representation models the input posterior feature as a linear combination of the full word posterior exemplars. The context size optimization is done using word-specific dictionaries.

It may be noted that the use of context-appended features is complementary or even alternative to the collaborative hierarchical structured sparse recovery. In fact, our experiments on CDR presented in Section 4.4.7.2 reveal that once the “optimal” context size is applied, the

	HMM	Template-matching	Sparse Modeling
Theory	Data is generated from probability distribution	Data lives in space spanned by all training templates	Data lives in a union of low-dimensional subspaces
Modeling	GMM/KL-HMM (Multinomial fitting)/DNN	Collection of Templates	Dictionary Learning
Algorithm	Viterbi Decoding	DTW Matching	Sparse Recovery

Table 4.3: Comparing HMM, DTW-based template matching, and sparse modeling approaches to ASR.

block-wise sparse recovery using word-specific dictionaries outperforms C-HiLasso based sparse recovery. Nevertheless, the compromise between smaller context and structured sparse recovery is an interesting feature of this work.

4.4.6 Contrasting Exemplar-based Sparse Modeling with HMM and DTW

In this section, we discuss the links between sparse modeling, HMM, and DTW-based template matching. Table 4.3 summarizes the key features of each approach. A detailed comparison between template-based approaches and HMM is also given in [De Wachter et al. (2007)].

The HMM and DTW are devised to find the best match between the acoustic input and a set of reference exemplars. In the case of HMM, the training exemplars are exploited to learn the parameters of a statistical model. Assuming that a probability distribution is a good hypothesis for the underlying generative process of the data, the HMM framework enables modeling the speech manifold with a Markovian structure through the design of a parametric dictionary where each atom characterizes the underlying probability distribution. The parametric design approach can lead to better generalization of the model with a fewer amount of training data. Modern HMM-based models use DNNs for estimating the probability distribution of the data. As compared to generative GMM models, a DNN is a discriminative model and requires a large amount of data for training. Instead of parametric probability distributions, DNNs rely on multiple layers of non-linear transformations of the data for finding discriminatory boundaries between the classes. On the other hand, DTW is a non-parametric approach where the word manifold is assumed to be spanned by all the exemplars from the training data. A test data point is characterized by the closest training exemplar based on DTW distance. In that sense, a DTW dictionary is the set of all training exemplars.

The sparse modeling approach relies on modeling the low-dimensional word manifold through dictionary learning rather than parametric design developed through HMM. In its essence, the dictionary models a union of low-dimensional sub-spaces through an overcomplete set of basis vectors instead of learning parameters of a GMM, multinomial distribution or DNN[3].

[3] The multinomial distribution arises in the derivation of KL-HMM (Aradilla et al., 2008) framework which has been shown to be a suitable acoustic modeling framework using posterior features.

Given K exemplars in a dictionary, the sparse modeling approach uses a linear combination of k exemplars, where $1 \leq k \ll K$, to characterize a test data point. In comparison, DTW based template matching uses exactly one closest exemplar from the training data to match the test data point, thus, resulting in a 1-sparse representation. An HMM-based model, on the other hand, is an *all-averaged* model because the probability distribution parameters are learned by combining all the training data points (e.g., KL-HMM state representatives or GMM means).

Investigating the Union of Subspaces Model Our hypothesis is that the sparse modeling approach is more accurate in the characterization of the test data posterior exemplars. More specifically, we want to verify that representing a posterior exemplar as a k-sparse combination of the training exemplars is more accurate than a 1-sparse (DTW-based) or an all-averaging (KL-HMM-based) characterization. To validate this hypothesis, we perform a simple experiment using DTW-based template matching for the 75-word dataset of PhoneBook. Out of ~11 utterances for each word, we keep 4 utterances as training templates and use the rest for testing. For each word, the 4 training utterances are time-aligned using DTW with respect to the longest utterance among them. The aligned utterances are then averaged to obtain a single template. The averaging is done by choosing from 1 up to 4 utterances at a time, thereby resulting in $\sum_{k=1}^{4} \binom{4}{k} = 15$ combinatorial ways of averaging. For example, if a word has 4 training utterances- U_1, U_2, U_3, and U_4, then the various k-sparse templates are as follows:

- 1-sparse templates : $T_{U_1}, T_{U_2}, T_{U_3}, T_{U_4}$
- 2-sparse templates: $T_{U_1U_2}, T_{U_1U_3}, T_{U_1U_4}, T_{U_2U_3}, T_{U_2U_4}, T_{U_3U_4}$
- 3-sparse templates: $T_{U_1U_2U_3}, T_{U_1U_2U_4}, T_{U_2U_3U_4}, T_{U_1U_3U_4}$
- 4-sparse(all averaged) template: $T_{U_1U_2U_3U_4}$

We then quantify the distance of the test utterances for each word with all of the different k-sparse templates constructed above. The distance used in this context is the weighted symmetric KL divergence as it was shown to be an appropriate distance measure in the posterior feature space [Aradilla et al. (2008)]. A smaller distance indicates a better characterization of the test template. The experiment is run on 464 test utterances from the 75-word vocabulary. For each test utterance, we determine the closest matching template for it. If this closest matching template is k-sparse, we assign the current test utterance to a group which is best characterized by k-sparse templates. In this way, we count the total number of test utterances assigned to each value of k. The observations of this experiment are given in Table 4.4.

We observe that only 4.9% of the test utterances have the least characterization error using a single closest template (DTW assumption). Moreover, only 9.7% utterances are best characterized by the model obtained from averaging the full training set (KL-HMM assumption [Aradilla et al. (2008)]). On the other hand, all remaining 85.4% of the utterances have the least characterization error using the templates which are obtained as a combination of a few (2 or 3) training utterances. This observation confirms the effectiveness of the union of subspace approach to model the posterior feature space.

	1-sparse	2-sparse	3-sparse	4-sparse
# of test utterances (out of 464)	23	177	219	45
% of test utterances	4.9	38.1	47.2	9.7

Table 4.4: Comparison of k-sparse templates for characterization of the test word utterances.

4.4.7 ASR Experiments Using Proposed Approach

In this section, we focus on evaluation of the proposed system on some simple automatic speech recognition tasks, through which we can understand the benefits and limitations of our approach.

4.4.7.1 Exemplar-based Isolated Word Recognition

The IWR evaluation is conducted on PhoneBook database. A word utterance is a sequence of 42-dimensional phone posterior vectors obtained from a DNN acoustic model. Prior studies [Aradilla and Bourlard (2009); Soldo et al. (2011)] have shown that posterior features perform well under DTW based template matching algorithm when the training data is limited to a few exemplars. Hence, we consider the DTW-based template matching as the baseline for this study. The DTW approach keeps all the utterances of a word in the training data available during testing. For each test utterance, the DTW matching is done with all the training data utterances to determine the closest matching training template. The class of the closest matching training template is thus assigned to the test utterance.

In this work, we follow the posterior-based sparse modeling approach explained in Section 4.3.2 and Figure 4.3. Word specific dictionaries are learned using the online dictionary learning algorithm from the 4 training exemplars for each word. The posterior features are context-appended with $c = 20$ neighboring frames from both left and right. Sparse recovery is done using Lasso algorithm with $\lambda = 0.1$, and sparse representations are projected in the positive quadrant as mentioned earlier in Section 4.3.2. The inference of the word is made using the dictionary which fulfills the least *accumulation of error* criteria in (4.13). Since the sparse modeling approach uses Euclidean distance metric for reconstruction, we keep the same metric for template matching using DTW.

System	PB75	PB600
DTW	84.7	73.5
Sparse Modeling	**97.8**	**93.2**

Table 4.5: Word recognition accuracies (in %) for the IWR task on PhoneBook database with 75-word vocabulary (PB75) and 600-word vocabulary (PB600) sets.

Table 4.5 shows the results for these experiments on 75-vocabulary and 600-vocabulary sets. We observe that the proposed posterior based sparse modeling framework performs better than the DTW template-matching system in both cases. It should be noted that the word recognition accuracy of a state-of-the-art hybrid DNN-HMM system presented in [Pinto et al. (2009)] is 98.8% for the 75-vocabulary set and 96.0% for the 600-vocabulary set. This system uses a 3-state left-to-right HMM for each phone, and the corresponding DNN is used to estimate posterior probabilities of these states. In comparison, our DNN acoustic model is non-complex as it is trained to simply predict the monophone probabilities instead of the HMM state probabilities. Previous work by [Soldo et al. (2011)] showed that DTW-based template matching approach gives its best performance when the weighted symmetric KL divergence is used as the distance metric. We did not experiment with KL divergence based approach here because an equivalent combination of dictionary learning and sparse modeling approach for exploiting KL divergence is not available.

4.4.7.2 Exemplar-based Connected Digit Recognition

The continuous speech recognition evaluation is conducted on Digits subset of Numbers database (details in Section 2.3.2) which contains connected sequences of digits 'zero' to 'nine' plus an alternative pronunciation 'oh' for digit zero. On a similar digit recognition task on Aurora-2 corpus [Hirsch and Pearce (2000)], previous approaches [Gemmeke et al. (2009, 2011)] employ a collection-of-exemplars based dictionary for sparse representation. Typically, the two-dimensional spectrogram of a word utterance is flattened by these approaches to form a spectral feature exemplar of the word. A collection of such exemplars is referred to as the dictionary in these approaches.

In this thesis, we employ the continuous speech recognition system discussed in Section 4.3.3 and Figure 4.3. We consider exemplars which are based on posterior features. Context appending is done similar to the prior approaches, but we do not create one exemplar per word. Instead, we optimize the context size by tuning it on a development set to encode the dynamics of the posterior features. The whole posterior feature data is therefore transformed into a context-appended posterior feature space. A context of 17 frames ($c = 8$) is found to be optimal on a development set, and the context-appended posterior exemplars are generated with a shift of one frame at a time.

CDR can now be approached using two techniques for computing the word posterior probabilities (Section 4.3.3)- block-wise word search and C-HiLasso based sparse recovery. In the former technique, the word posterior probabilities are estimated for each input frame using the reconstruction error based formulation in (4.13). On the other, the latter technique uses (4.9) for directly estimating the word posterior probabilities from the sparse representation. A sequence of $T' = 3$ context-appended feature frames are considered for C-HiLasso to exploit the collaborative group sparsity structure in the sparse representations. After obtaining the word posterior probabilities from block-wise search or C-HiLasso sparse recovery, a Viterbi decoder [Rabiner (1989)] is employed to decode the word sequence from these probabilities. The decoder uses a flat language model for digit sequences. For each digit, we learn the maximum and minimum durations from the training set. The decoder applies duration penalties

#	System	WER(in %)
1	Collection of (posterior) exemplars	21.4
2	Word Dictionary (block-wise search)	14.6
3	Word Dictionary (C-HiLasso)	18.5

Table 4.6: Performance of dictionary-based sparse modeling versus collection-of-exemplars based approach (in WER %) on CDR task on Digits database.

to all the paths where these duration constraints are violated. Duration modeling is crucial here to distinguish a single instance of a digit from the consecutive occurences of the same digit more than once. Without duration penalties, our sparse modeling approach has no way of finding word boundary between repeated occurence of the same word.

The results are presented in Table 4.6. System 1 presents the performance of a collection-of-exemplar based approach where each word dictionary has posterior exemplars from ~3000 training utterances. To create such a dictionary, the digit sequences in the training data of Digits database split into short digit-specific segments. The algorithmically designed dictionaries in our approach use a concatenation of 100 utterances for initialization of the dictionary, and the remaining utterances are used for updating the atoms of the dictionary using the online dictionary learning algorithm. Systems 2–3 in Table 4.6 depicts the results of our approach. We observe that the block-wise dictionary learning (System 2) performs better (14.6% WER) as compared to the baseline collection-of-exemplars approach (System 1). Although the performance of C-HiLasso approach (System-3) is worse than the block-wise search approach, it still achieves satisfactory results. The window size for C-HiLasso is an important parameter, and we obtain the best results for this approach after optimizing it to a window size of 3 frames. The most interesting aspect of C-HiLasso approach is that it generates the true word posterior probabilities using (4.9). We speculate that the block-wise search performs better than the C-HiLasso approach due to a more tractable sparse recovery problem enabled by the smaller word-specific dictionaries.

4.5 Conclusions

The present work demonstrates a novel study on exemplar-based sparse modeling of speech using DNN based posterior features. In this context, the posterior features not only prove competent for exemplar-based ASR but also provide an elegant probabilistic interpretation to the sparse modeling approach. We show that exemplar-based speech recognition systems can benefit from dictionary learning algorithms by reducing the collection of all training exemplars into a small learned "basis" set. The dictionary learned in this manner is effective in characterizing the non-linear manifolds associated with the linguistic units, e.g., words. We confirm the hypothesis stated in 4.1 that the posterior features can be effectively characterized using sparse modeling over overcomplete dictionaries.

We observe that the temporal sequencing information can be exploited by using either context-appended segmental features or collaborative hierarchical sparse recovery. Structured sparse modeling ensures that the representation coefficients collaborate to activate a common set of dictionary atoms corresponding to the same word. The choice of appropriate window size to be used for context-appending as well as structured sparse recovery is a parameter dependent on the speech units being recognized.

Our approach also has certain limitations. Firstly, extending the exemplar-based sparse representation approach to LVCSR tasks is not trivial. For a large vocabulary, we might not have enough data for each word to learn the individual word-specific overcomplete dictionaries. Therefore, some word dictionaries might not capture enough variability as required. Secondly, sparse recovery over a large vocabulary of word classes is a L-way classification problem where L is the vocabulary size. As L increases, the time complexity for computing word posterior probabilities by sparse recovery using block-wise search increases linearly and may blow up quickly. Similarly, seeking for structured sparsity using C-HiLasso over the complete multi-word dictionary $\mathbf{D}$ is not scalable as the size of the dictionary grows linearly with increasing L. Nevertheless, our approach has huge potential for tasks like keyword spotting and query-by-example spoken word detection. The diagonal sequence pattern (in Figure 4.5) visible in the sparse representation when a sequence of test posterior exemplars are sparse coded with the correct dictionary can be detected using a DTW matching [Ram et al. (2018a)] algorithm or a convolutional neural network [Ram et al. (2018b)] in order to identify a keyword or a query-by-example.

In the next chapter, we devise clever ways to overcome the limitations of our approach. We extend the application of our framework to LVCSR tasks, perform a thorough analysis of its workings, and get deeper insights into its implications.

5 A Low-dimensional Senone Subspace Modeling Approach Towards ASR

5.1 Introduction

This chapter focuses on explicitly exploiting the low-dimensional senone subspaces in speech towards the goal of improving acoustic modeling for ASR. Specifically, it investigates the application of (1) dictionary based sparse modeling and (2) PCA based low-rank modeling for characterizing the senone subspaces. This chapter is organized as follows. In Section 5.1.1, we present the motivations and contributions of this work. Section 5.2 discusses the presence of low-dimensional senone subspaces that underlie beneath DNN posteriors. Section 5.3 shows how to employ sparse representations to explicitly enhance DNN acoustic modeling. In Section 5.4, we exploit sparse and low-rank soft targets to train enhanced DNN acoustic models under a student-teacher framework. Lastly Section 5.5 draws the conclusions of the work presented in this chapter.

5.1.1 Motivation and Our Approach

A typical large vocabulary ASR system works with DNN acoustic models that output posterior probabilities for $\sim 10^3$ senones (defined in Section 2.2.4 and [Young et al. (1994)]). If the DNN posteriors are seen as intermediate features in the ASR pipeline (Section 2.2), we hypothesize that there exist low-dimensional senone-specific subspaces embedded beneath them. These latent subspaces carry the important speech information which is crucial for ASR. However, posterior features are often corrupted with high-dimensional noise arising from data mismatch or inaccuracies in DNN estimations which make the senone subspaces inaccessible. Therefore, the major goal of this chapter is to extract the senone subspaces from noisy DNN posteriors and consequently improve DNN-HMM based ASR.

To motivate our approach, we invoke the subspace-sparse recovery (SSR) property developed in [Elhamifar and Vidal (2013)] as follows. According to this property, we consider DNN posteriors as compressed signals which exhibit a unique subspace belonging to each underlying senone. If there are K senones, the DNN posteriors are K-dimensional vectors. Let $\mathbf{S} = \{\mathcal{S}_k\}_{k=1}^{K}$

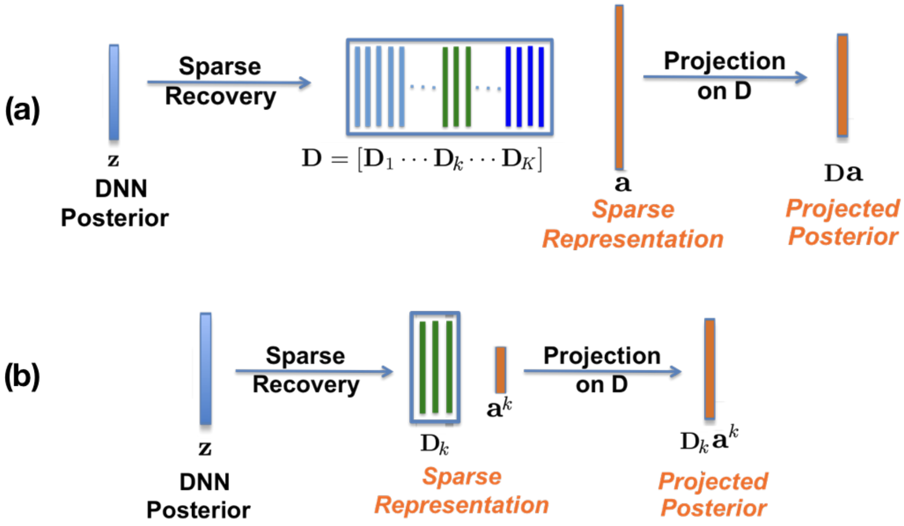

Figure 5.1: Extracting the senone-specific subspaces in a DNN posterior by projecting it to a high-dimensional space. In (a) an overcomplete multi-class dictionary is used for projection, and in (b) a senone-specific undercomplete dictionary is used.

be the set of linear disjoint senone-specific subspaces associated with the K senone classes in $\mathbb{R}^K$ such that the dimensions of individual subspaces $\{R_k\}_{k=1}^K$ are smaller than the dimension of the posterior space, i.e. $\forall k$, $R_k < K$.

Posterior features $\mathbf{z}$ lie in the union $\cup_{k=1}^K \mathcal{S}_k$ of these low-dimensional subspaces. Let $\mathbf{D}_k \in \mathbb{R}^{K \times N_k}$ be the class-specific overcomplete dictionary for senone-specific subspace $\mathcal{S}_k$ where N_k is the number of atoms in $\mathbf{D}_k$ and $N_k > R_k$. Each data point in $\mathcal{S}_k$ can then be represented as a sparse linear combination of the atoms from $\mathbf{D}_k$. The *subspace-sparse recovery* (SSR) property [Elhamifar and Vidal (2013)] for union of disjoint subspaces asserts that the ℓ_1-norm sparse representation of a data point over the collection of all class-specific dictionaries $\{\mathbf{D}_k\}_{k=1}^K$ can lead to separation of the class-specific subspaces by selecting atoms only from the underlying class of the data point for its reconstruction. The collection of all class-specific dictionaries is the multi-class overcomplete dictionary $\mathbf{D}$. Thus, the sparse representation obtained for a posterior $\mathbf{z}$ belonging to senone class k has activations only for those atoms in $\mathbf{D}$ which correspond to the subspace $\mathcal{S}_k$ where $\mathbf{z}$ lives.

Note that, considering a speech utterance as a union of words, phones or sub-phonetic components, the subspaces $\mathcal{S}_k$ can be modeled at different levels (time granularity) corresponding to any of these speech units. Consequently a dictionary $\mathbf{D}$ can be constructed by learning basis sets $\mathbf{D}_k$ for individual classes. In the present study, we focus on context-dependent senones due to their superior quality for acoustic modeling in DNN-HMM framework. Nevertheless, there is no theoretical/algorithmic impediment in applying it for larger units such as words. The rigorous proof of SSR property (see Theorem 2 in [Elhamifar and Vidal (2013)]) requires specific conditions and assumptions on disjoint subspaces. Since we train DNNs with binary senone target outputs, the intersection of senone subspaces is expected to be a rare event and suggests disjointedness of subspaces. Although we consider further theoretical analysis beyond the scope of the present work, the experiments conducted in this chapter empirically confirm that SSR property indeed holds for subspace-sparse modeling of senones.

As shown in Figure 5.1(a), the multi-class dictionary $\mathbf{D}$ captures each senone-specific subspace using a sub-dictionary $\mathbf{D}_k$. Posterior feature $\mathbf{z}$ can be projected on this overcomplete dictionary so as to disentangle the underlying low-dimensional subspaces. As per SSR property, only the correct senone subspace is activated through the appropriate dictionary atoms in the sparse representation $\mathbf{a}$. The projected vector $\mathbf{Da}$ in Figure 5.1(a) retains only the information of the correct underlying subspace and discards the high-dimensional noise. We term this vector as the *projected* or *enhanced* posterior.

In practice, for any labeled training data, we already know the senone classes of DNN posterior features. Therefore, we do not need to employ sparse recovery using the complete dictionary $\mathbf{D}$ and rely on the SSR property to pick the correct sub-dictionary. Instead, we can directly project a posterior $\mathbf{z}$ on the class-specific dictionary $\mathbf{D}_k$ to get the projected posterior $\mathbf{D}_k\mathbf{a}^k$ as shown in Figure 5.1(b). While the multi-class dictionary $\mathbf{D}$ is overcomplete, the senone-specific dictionaries $\mathbf{D}_k$'s are *undercomplete*. Sparse recovery using $\mathbf{D}_k$ assumes that the senone subspace $\mathcal{S}_k$ underlying the posterior $\mathbf{z}$ has lower dimensionality than the size of the sub-dictionary, i.e., the condition $R_k < N_k$ holds. In this chapter, we explore both of the approaches shown in Figure 5.1.

Following the theory explained above, we first confirm that the senone specific posterior data is indeed compressible by showing that it has a very low rank as compared to the dimension of the posterior space. Next, we propose the modeling of senone subspaces using basis sets obtained from CS based *dictionary learning* as well as a principal component analysis (PCA) approach. DNN posteriors exhibit class specific low-dimensional structures which are usually superimposed with high-dimensional unstructured noise. While the structures are global and pertain to the whole population, the noise is local and could be a result of erroneous estimations by the DNN on the individual input frames. The basis sets learned using dictionary learning or PCA focus on capturing the global patterns and do not model the local misinformation present in individual posteriors. To be specific, a CS dictionary used for sparse recovery learns to model the non-linear speech manifold as a union of low-dimensional subspaces whereas the strength of PCA lies in capturing the linear regularities in the data [Hutchinson et al. (2015)]. Once the basis set is learned using a dictionary or PCA, any posterior sample can be expressed concisely by a simple projection over the basis set. The projection process discards the random high-dimensional noise present in the posterior whereas the global low-dimensional patterns of the correct subspace are enforced. Finally, we project the concise representation back onto the original dimensions of the posterior space to get an enhanced version of the original posterior. We call this process as *enhancement* of DNN posteriors using sparse (dictionary based) or low-rank (PCA based) modeling of the senone subspace.

We provide experimental evaluation of our approach on Numbers database [Cole et al. (1995)] and AMI corpus [McCowan et al. (2005)]. Details of these databases can be found in Section 2.3. For Numbers database, we utilize the multi-class dictionary based sparse recovery and use the enhanced posteriors directly for decoding. In case of AMI corpus, we employ the enhanced DNN posteriors as soft targets (non-binary probability vectors) to train more accurate DNN acoustic models.

Apart from the empirical analysis provided in this chapter, we also offer a more theoretical insight into why enhanced DNN posteriors act as better targets for acoustic model training. To do so, we develop an information theoretic analysis of our approach which is explained in Chapter 7. Please note that this work investigates low-dimensional subspaces at the hierarchy of senones and not of phones, phone states or words. We consider senone subspaces because senone posteriors have interpretable correlations as explained in Section 5.2.2 and because modern hybrid ASR systems work directly with data likelihoods conditioned on senone states for HMM-based decoding. Another choice we make is performing senone subspace modeling on DNN posteriors instead of acoustic features. As explained in Chapter 4, posteriors are more robust, and they are expected to extract acoustic information which is invariant towards the speaker and environmental variations in speech.

5.1.2 Prior Research

Earlier works on exploiting low-dimensionality in DNN acoustic modeling focus on exploiting low-rank and sparse representations to modify DNN architectures for small footprint implementation. In [Xue et al. (2013); Sainath et al. (2013)] low-rank decomposition of the neural network's weight matrices enables a reduction in DNN complexity and memory footprint. Similar goals have been achieved by exploiting sparse connections [Yu et al. (2012)] and sparse activations [Kang et al. (2015)] in hidden layers of DNN. An important work in exploiting sparsity for model regularization is Dropout training of neural networks [Srivastava et al. (2014)] where a fraction of hidden neurons are randomly turned off during model training to reduce the capacity of the network. This leads to the training of a collection of many sparsely connected DNNs which are then averaged into a combined network during testing. Manifold regularization has been explored in [Tomar and Rose (2014)] to preserve the underlying low dimensional manifold based relationships amongst speech features during DNN training. Another way of exploiting low-rankness of speech features is by using a bottleneck layer in DNN acoustic models. A low-dimensional bottleneck layer achieves model compression as well as model regularization. Bottleneck features have been famously employed in the tandem ASR approach [Hermansky et al. (2000)] and for transfer learning [Pan and Yang (2009)] for various speech processing tasks. A major difference between bottleneck layer DNNs and the low-rank approaches proposed in this chapter is that a bottleneck layer assumes all the underlying senone classes to live in a common low-dimensional subspace whereas our approach learns unique low-dimensional subspaces for each senone class separately.

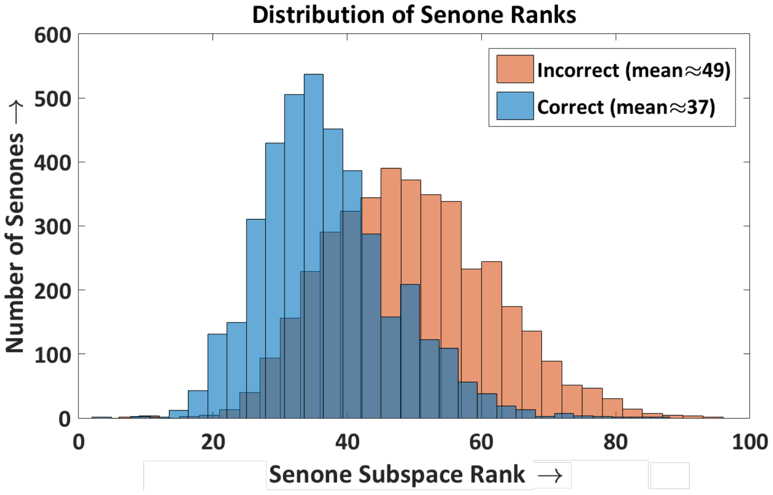

Figure 5.2: Rank of senone subspaces in *"correct"* DNN posteriors is found to be very small (mean=37) as compared to the dimension of DNN posteriors which is 4007 here. "Incorrect" posteriors are noisy resulting in a higher rank(mean=49).

In another line of research, soft targets based DNN training has been found effective for enabling model compression [Hinton et al. (2015); Chan et al. (2015)] and knowledge transfer from an accurate complex model to a smaller network [Jinyu Li (2014); Price et al. (2016)]. Sparse subspace modeling has also been successfully utilised with state-of-the-art performance in spoken term detection [Ram et al. (2018a, 2016, 2015)] and for extracting deep sparse representations for speech recognition [Sharma et al. (2017)].

5.2 Senone Subspaces in DNN Posteriors

This section presents a study of DNN posteriors which is conducted to confirm the presence of low-dimensional senone subspaces. The study comprises of a rank-analysis of posteriors followed by a discussion on why senone classes are correlated with other.

5.2.1 Rank Analysis

The presented rank-analysis is based on posteriors generated from DNN acoustic models trained for ASR on Numbers database and AMI corpus. Details of the acoustic models are given in Section 5.3.1 for Numbers and Section 5.4.2 for AMI. These systems have 557 and 4007 senones respectively, which are also the dimensions of DNN posterior space in each case. With the help of forced senone alignment from a GMM-HMM system, we segregate the posteriors in senone-specific matrices. For each matrix, we compute the number of singular values required to preserve 95% variability of the data in it. Due to the skewed distribution of the posteriors, we convert the data matrices to the logarithmic domain before performing singular value decomposition. We refer to the number of required singular values as an approximation of the true *rank* of senone matrices.

	Numbers	AMI
Rank-Correct	36.6	36.9
Rank-Incorrect	45.5	48.9

Table 5.1: Comparison of "Rank" of "*correct*" DNN posteriors versus "*incorrect*" DNN posteriors for Numbers database and AMI corpus.

An ideal posterior should have its maximum component at the support indicating its associated class. Hence, we categorize the posteriors as "correct" if the maximum component corresponds to the correct class and "incorrect" if the maximum component corresponds to the incorrect class. Table 5.1 provides the results of this analysis. For "correct" posteriors, the mean over all classes is found to be ~ 37. Since the dimension of the posterior spaces is much higher for both the databases, this indicates the presence of low-dimensional senone subspaces underlying the DNN posterior matrices. In contrast, the "incorrect" posteriors have a higher mean rank of ~ 46 for Numbers and ~ 49 for AMI. While both categories have a rank far lower than the dimension of DNN posterior space (= 4007), it is important to note that the information bearing components in "correct" senone posteriors are fewer than those in "incorrect" posteriors resulting in matrices which have a lower rank. For AMI, we also depict this analysis in the form of histograms in Figure 5.2. Our analysis suggests that the "incorrect" posterior are exposed to some high-dimensional spurious noise which degrades their quality. Therefore, we conclude that there is a scope to enhance the posterior probability estimates by discarding the unwanted noise and enforcing the information bearing components in DNN posteriors.

5.2.2 Correlation Among Senone Classes

If all senones classes were mutually uncorrelated, then DNN posteriors labeled as a particular senone would have an ideal rank of 1 corresponding to the dimension of that senone. However, the rank analysis in Figure 5.2 shows that senone subspaces are complex and even for the "correct" posteriors, the rank(~ 37) is much higher than unity. These multi-dimensional senone subspaces are formed due to correlations among various senones, and we identify here- sequential and structural dependencies- as two possible reasons for these correlations.

5.2.2.1 Sequential Correlations

During training of a conventional DNN acoustic model for ASR (Section 2.2.5.1), hard targets are used to assign a particular senone label to a relatively long sequence of (~10 or more) input acoustic frames. In contrast, senone transitions are quite frequent, and their durations are usually shorter than the length of the input context window. Thus, a long context of input frames may lead to a presence of acoustic features corresponding to multiple senones in the input (Figure 5.3(a)) which renders the assumption of binary outputs inaccurate. We argue

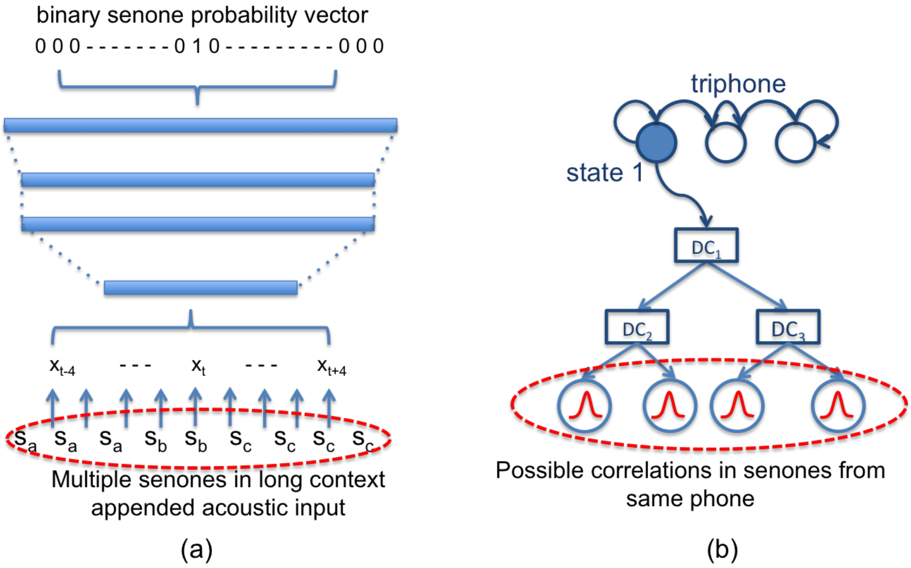
binary senone probability vector
0 0 0 - - - - - - - 0 1 0 - - - - - - - - - 0 0 0
x_{t-4}
- - -
x_t
- - -
x_{t+4}
s_a s_a s_a s_b s_b s_c s_c s_c s_c
Multiple senones in long context appended acoustic input
(a)
triphone
state 1
DC_1
DC_2
DC_3
Possible correlations in senones from same phone
(b)

Figure 5.3: Correlation among senones due to: (a) long input context and (b) acoustically similar root in decision trees.

that soft DNN posteriors quantify such sequential information using non-zero probabilities for multiple senone classes. Senones which frequently appear in the neighboring context of each other would exhibit these correlations in their DNN posterior space. The contextual senone dependencies arising in soft targets can also be attributed to the ambiguities due to phonetic transitions [Gillick et al. (2011)].

5.2.2.2 Structural Correlations

The procedure of senone extraction using decision trees [Young et al. (1994)] can lead to correlations among multiple senone classes. A family of senones corresponding to the same phone-HMM state are context dependent acoustic variations of each other as they all share the same root in the decision tree (Figure 5.3(b)). Due to this structural correlation, these senones may be confused with one another during DNN based posterior estimation, and this can result in correlated dimensions in the DNN outputs.

Figure 5.4(c) depicts the presence of unique global patterns in the population of DNN posteriors belonging to a particular senone. These patterns are visible clearly when the unstructured high-dimensional noise is removed. The noise in DNN posteriors may originate from random local effects, inaccuracies in DNN training or training-testing data mismatch. In the next section, we propose our approach of enhancing the noise-prone posteriors obtained from a hard target based DNN acoustic model.

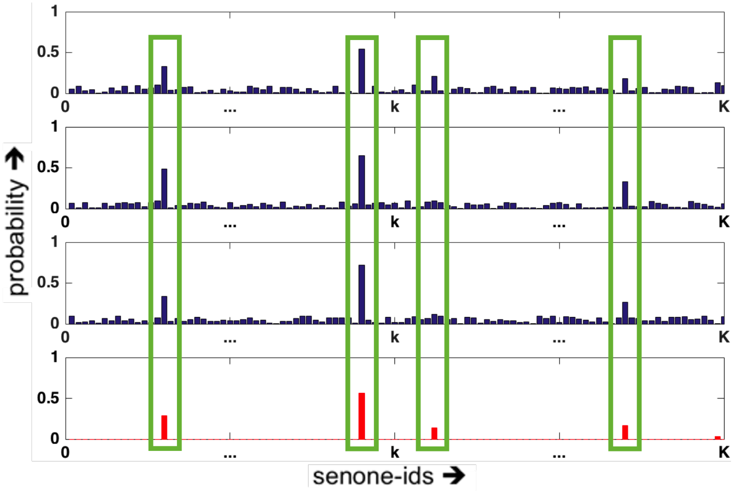

Figure 5.4: We show examples of DNN posterior features for a particular senone class (in blue barplots) which highlight low-dimensional patterns (green boxes) super-imposed with unstructured noise. PCA and dictionary-based projection (Section 5.4) enable recovery of the underlying patterns by discarding the unstructured noise, and provide more reliable soft targets for DNN training. K denotes the size of DNN outputs which is equal to total number of senones.

5.3 Sparse Representations Based Enhanced Acoustic Modeling

In this section, we provide an empirical analysis of the theoretical discussion presented in Section 5.1.1. Specifically, we focus on sparse modeling of DNN posteriors using multi-class dictionaries as shown in Figure 5.1(a). The experiments in this section are based on Numbers database. They confirm that the information bearing components of DNN posteriors can be enhanced using projection on an overcomplete dictionary which removes the effect of high-dimensional noise leading to improvement in DNN-HMM based ASR performance.

5.3.1 Database and Speech Features

We use Numbers subset of Numbers'95 corpus (Section 2.3.2) for this study where only the utterances with 30 most frequent words are kept from the original corpus. Context-dependent triphone state-tying results in 557 senones for this database. The training data is forced aligned to these senones Kaldi speech recognition toolkit [Povey et al. (2011)]. A DNN is trained with cross-entropy loss minimization criteria with 3 hidden layers each having 1024 nodes. For every 10 ms speech frame, the DNN input is a vector of MFCC+Δ+$\Delta\Delta$ features with a context of 9 frames (39×9=351 dimension). The DNN output is a vector of posterior probabilities corresponding to 557 senone classes.

5.3.2 Generating Enhanced Posteriors Using Sparse Modeling

Building on our experiments presented in Chapter 4 on dictionary learning for sparse modeling of posterior features, we use the online dictionary learning Mairal et al. (2010) algorithm and Lasso sparse solver for learning dictionaries and solving the ℓ_1 sparse coding problem.

Class-specific data of senone posterior features is obtained through GMM-HMM based forced alignment on training data, which is then used to learn individual over-complete basis set $\mathbf{D}_k$ for each senone subspace S_k using dictionary learning algorithm. These class-specific dictionaries are concatenated into a larger dictionary $\mathbf{D} = [\mathbf{D}_1 \cdots \mathbf{D}_k \cdots \mathbf{D}_K]$ for subspace-sparse acoustic modeling. Since any posterior feature obtained from DNN lies in a union of subspaces $\cup_{k=1}^{K} \mathcal{S}_k$, a test posterior feature $\mathbf{z}$ can be reconstructed using the atoms of dictionary $\mathbf{D}$. According to SSR property, only the atoms associated to the correct class (underlying subspace) of $\mathbf{z}$ will be used for sparse representation.

We use group sparsity based hierarchical Lasso algorithm [Sprechmann et al. (2011)] for sparse coding to enforce group sparsity in $\mathbf{a}$ based on the internal partitioning of dictionary $\mathbf{D}$ into senone-specific sub-dictionaries $\mathbf{D}_k$. For each test data DNN posterior feature $\mathbf{z}^{\text{test}}$, the high dimensional group sparse representation $\mathbf{a}^{\text{test}}$ is computed by sparse recovery over $\mathbf{D}$. Since dictionary $\mathbf{D}$ is learned from training data posteriors, the projection of the test posterior feature $\mathbf{z}^{\text{test}}$ on training data space is given by computing $\mathbf{D}\mathbf{a}^{\text{test}}$.

Note that $\mathbf{D}\mathbf{a}^{\text{test}}$ is an approximation of posterior feature $\mathbf{z}^{\text{test}}$ based on ℓ_1-norm sparse reconstruction using atoms of $\mathbf{D}$. Consequently, it has the same dimension as $\mathbf{z}^{\text{test}}$ and it is forced to lie in a probability simplex by normalization after the sparse recovery. Figure 5.1(a) summarizes this procedure.

5.3.3 Rank Analysis Continued

To supplement the rank analysis done in Section 5.2.1, here we provide a similar study on the sparse modeling based enhanced posteriors. Table 5.2 shows that the reconstruction of the DNN posteriors using overcomplete dictionaries significantly reduces the rank of DNN posteriors. The analysis in both cases has been done on a development set. This observation confirms that using sparse recovery, the low-dimensional senone subspaces beneath DNN posterior become accessible and the high-dimensional noise is removed.

To further study the *true* underlying dimension of the senone-specific subspaces, we con-

	DNN	Projected	Robust PCA
Rank-Correct	36.6	11.9	7.6
Rank-Incorrect	45.5	21.7	11.7

Table 5.2: Comparison of “Rank” of DNN posterior matrix, projected posterior matrix and RPCA senone posterior matrix. The dimension of corresponding posterior vectors is 557 here.

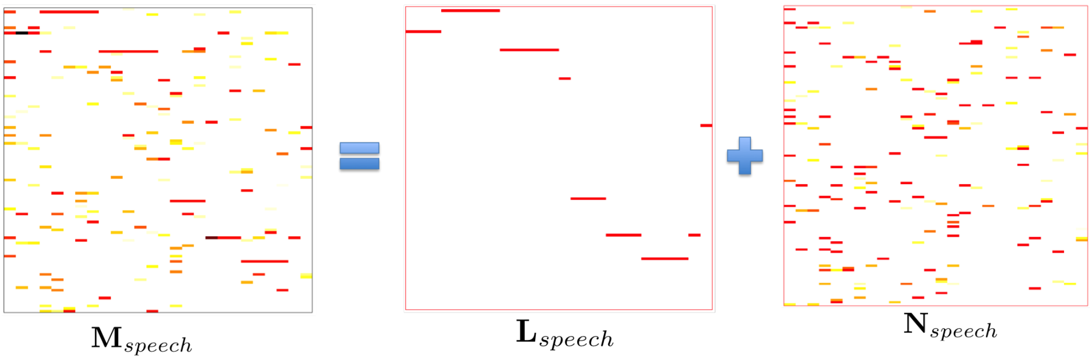

Figure 5.5: Decomposing a DNN estimated senone posterior matrix $\mathbf{M}_{\text{speech}}$ into a low-rank matrix $\mathbf{L}_{\text{speech}}$ of enhanced posteriors and a sparse matrix $\mathbf{N}_{\text{speech}}$ of spurious noise using RPCA.

sider a robust principle component analysis (RPCA) based decomposition of the senone posteriors [Candès et al. (2011)]. The idea of RPCA is to decompose a data matrix **M** as

$$\mathbf{M} = \mathbf{L} + \mathbf{N} \tag{5.1}$$

where matrix **L** has low-rank and matrix **N** is sparse (Figure 5.5). The low-rank component **L** corresponds to the enhanced posteriors whereas the high dimensional erroneous estimates are separated out in the sparse matrix **N**.

We collect posterior features for each senone from training data using ground truth based GMM-HMM forced alignment. RPCA decomposition is applied to data of each senone-class to reveal the *true* underlying dimension of the class-specific senone subspaces. The rank of senone posteriors (i.e., rank of **L**) obtained after RPCA decomposition for both “Correct” and “Incorrect” classes are listed in Table 5.2. We observe that the *true* dimension (rank~7.6) of the class-specific subspaces of senone posteriors is indeed far lower than the DNN posteriors (rank~36.6) and yet lower than the projected posteriors (rank~11.9).

5.3.4 Enhanced DNN-HMM Speech Recognition

In this section, ASR decoding is done using both DNN posteriors and projected posteriors in the framework of conventional hybrid DNN-HMM. The same HMM topology learned during training of the hybrid DNN-HMM is used in all cases. Hence, all parameters of different ASR systems shown here are the same, and the only difference is in terms of senone posterior probabilities at each frame which results in different best paths being decoded under the Viterbi algorithm.

To demonstrate the increased robustness in projected posteriors as compared to the DNN posteriors, we also examined their performance in noisy conditions where an artificial white Gaussian noise was added at the signal level to the test utterances at signal-to-noise (SNR) ratios of 10 dB, 15 dB and 20 dB. The DNN acoustic model trained on clean speech is used for computing posteriors from the noisy test spectral features so that the artificially added noise acts as an unseen variation in the data for the DNN.

Posteriors	Test Data Condition			
	Clean	20 dB	15 dB	10 dB
DNN Output	2.6	4.0	6.8	14.0
Projected	2.2	3.5	6.2	13.9

Table 5.3: Comparison of ASR performance (in WER %) using DNN posteriors and projected posteriors in clean and noisy test conditions on Numbers database. Clean data is used for training.

Comparison of ASR performance is shown in Table 5.3 in terms of Word Error Rate (WER) percentage. We observe that the projected posteriors outperform DNN posteriors in all cases suggesting that projection based on **Da** is more accurate for DNN-HMM decoding than original posteriors. We also compare the GMM-HMM based forced senone alignment (ground truth) with senone alignments achieved by best Viterbi paths in projected posterior and DNN posterior systems. Senone classification error of 24.1% in case of DNN posteriors is reduced to 19.8% in case of projected posteriors. Improvement in senone alignments and subsequent reduction in WER proves the superior quality of projected posteriors over DNN posteriors and supports the hypothesis that projection moves the test features closer to the subspace of the correct classes.

Finally, RPCA posteriors (matrix **L** obtained from low-rank and sparse decomposition as explained in Section 5.3.3) which have ranks close to the true underlying dimensions of senone subspaces perform exceptionally well in ASR. WER of 2.6% using DNN posteriors (rank~36.6) reduces to a WER of 2.2% using projected posteriors (rank~11.9), i.e., a relative improvement of 15.4%, and when RPCA posteriors (rank~7.6) are used, it is reduced to a mere 0.4%. Since RPCA based low-rank reconstruction of posteriors has been done using ground truth senone alignment, ASR performance, in this case, is the best case scenario and indicates the possible scope of improvement.

5.4 Low-rank and Sparse Soft Targets Based Enhanced DNN Acoustic Modeling

In Section 5.3, we exploit the multi-class dictionary for enhancing the DNN posterior probabilities. The experiments conducted on Numbers database involved a set of 557 senones only owing to a small-sized vocabulary used in the utterances (which comprise of sequences of numbers). On the contrary, an LVCSR task may need a much larger set of senones which is usually in the order of $\sim 10^3 - 10^4$. Consequently, the DNN posteriors are equally high-dimensional and our approach needs to learn as many senone-specific dictionaries as the dimension of the posterior. For a large value of K (denoting the number of senone classes), the sparse recovery problem solved in Section 5.3 is not trivially scalable. Firstly, the size of the dictionary would be very huge to encompass the variability of all the senones subspaces and be overcomplete as well. Secondly, framewise group sparse recovery algorithm [Sprechmann et al. (2011)] in such a high-dimensional space is not tractable and computationally very slow. Owing to these reasons, our efforts were not fruitful when we employed a multi-class dictionary for enhancing DNN posteriors in the ASR setup for the large-vocabulary AMI meeting corpus. Therefore, we employ two different strategies to enhance DNN posteriors for improving ASR on large-vocabulary tasks like the AMI corpus. We discuss these strategies in the following sections.

5.4.1 From DNN Posteriors to Enhanced Soft Targets

5.4.1.1 Sparse Soft Targets Using Dictionary-based Reconstruction

For an LVCSR task where the number of senones is very large, we do the sparse recovery based enhancement of DNN posteriors only on the training data. Since the senone labels are known for the training data, we can directly use senone-specific dictionaries for sparse recovery. This approach is shown in Figure 5.1(b). The enhanced training data posteriors, thus obtained, are then considered as soft targets for training an enhanced DNN acoustic model under the student-teacher training framework. Student-teacher training of DNNs has been a well-known technique for knowledge transfer and distillation [Jinyu Li (2014); Hinton et al. (2015); Chan et al. (2015); Price et al. (2016)]. The basic idea behind this technique is that a teacher DNN (often trained with hard targets) provides soft targets for training a student DNN. The intuition is that the soft targets encode the knowledge of the teacher-DNN through the inter-dependencies among the output dimensions.

Note that the original DNN posteriors can also be considered as soft targets for training a student DNN. However, the potential of original posteriors is reduced due to the presence of unstructured noise. Therefore, to obtain reliable soft targets, we rely on the sparse modeling based enhancement procedure.

We do a brief recap of the sparse modeling based posterior enhancement procedure here with respect to senone specific dictionaries. Given an already learned overcomplete dictionary $\mathbf{D}_{\text{SP}}$ for senone class k, where subscript *SP* denotes *sparse*, the sparse representation $\mathbf{a}$ of a DNN posterior z is obtained by the Lasso optimization problem stated in (3.4) as:

$$\hat{\mathbf{a}} = \arg\min_{\mathbf{a}} \|\mathbf{z} - \mathbf{D}_{\text{SP}}\,\mathbf{a}\|_2^2 + \lambda \|\mathbf{a}\|_1. \tag{5.2}$$

The enhanced posterior is obtained by projecting the sparse code $\mathbf{a}$ on dictionary $\mathbf{D}_{\text{SP}}$ as:

$$\mathbf{z}^{SP} = \mathbf{D}_{\text{SP}}\,\hat{\mathbf{a}} \tag{5.3}$$

The regularization parameter λ in (5.2) controls the sparsity of the sparse code $\mathbf{a}$. Owing to sparsity constraints, the reconstruction term in (5.2) enforces the sparse recovery process to extract and emphasize only the subspace specific global patterns in the reconstruction $\mathbf{z}^{SP}$ whereas the random noise, which is local to $\mathbf{z}$, is discarded. While a low value of λ (less sparse solutions) may result in inefficient noise reduction, a higher value (more sparse solutions) may discard even the essential information contained in the posterior. Thus, an optimal value of λ is desired for dictionary learning and sparse recovery. We tune the value of λ for better ASR on a development set.

5.4.1.2 Low-rank Soft Targets Using PCA-based Reconstruction

The rank analysis using RPCA approach discussed in Section 5.3.3 suggests that the end goal for obtaining accurate posterior probabilities through an acoustic model should be to access the low-dimensional senone subspaces. Towards this end, we propose a low-rank modeling of DNN posteriors for enhancing their quality. Explicitly, we rely on principal component analysis (refer to Section 3.4) to characterize the subspace of each senone separately. If a large population of DNN posteriors is collected for a particular senone class, the frequent dependencies (visible in Figure 5.4) are exhibited as the regularities among the correlated dimensions in senone posteriors. As a result, the matrix formed by concatenation of senone-specific posteriors has an intrinsic low-rank structure.

If $\mathbf{M}_k \in \mathbb{R}^{K\times N}$ denotes a matrix of N mean-centered posteriors in the log-domain such that they are labeled as senone k in the forced alignment, then we can obtain the principal component matrix $\mathbf{P} \in \mathcal{R}^{K\times K}$ using eigenvector decomposition. The columns of $\mathbf{P}$ are eigenvectors of $\mathbf{M}$ arranged in the order of decreasing singular values associated with them. Eigenvectors in $\mathbf{P}$ which correspond to the large eigenvalues in constitute the frequent regularities in the subspace, whereas others carry the high-dimensional unstructured noise. Hence, we define a low-rank projection matrix as:

$$\mathbf{D}_{\text{LR}} = \mathbf{P}_{1:l} \in \mathcal{R}^{K\times l} \tag{5.4}$$

where the subscript LR stands for low-rank. $\mathbf{P}_{1:l}$ is truncation of $\mathbf{P}$ that keeps only the first l eigenvectors and discards the erroneous variability captured by other $K-l$ components. We select l such that relatively σ% variability is preserved in the low-rank reconstruction of original senone matrix $\mathbf{M}$.

The eigenvectors stored in the low-rank projection $\mathbf{P}_l$ are referred to as ***"eigenposteriors"*** of the senone space. Using the eigenposterior matrix $\mathbf{D}_{LR}$, the low-rank reconstruction of a mean-centered log posterior $\tilde{\mathbf{z}}_t$, denoted by $\tilde{\mathbf{z}}_t^{\text{LR}}$ can be estimated as:

$$\tilde{\mathbf{z}}_t^{\text{LR}} = \mathbf{D}_{\text{LR}}\mathbf{D}_{\text{LR}}{}^{\top}\tilde{\mathbf{z}}_t \tag{5.5}$$

Finally, we add the mean of the log-posteriors for senone class k to the reconstructed posterior $\tilde{\mathbf{z}}_t^{\text{LR}}$ and exponentiate it to obtain a low-rank senone posterior $\mathbf{z}_t^{LR}$. The enhanced posterior $\mathbf{z}_t^{LR}$ can now be used as soft targets for learning improved DNN acoustic models (Fig.5.6).

We assume that σ% variability, that quantifies the low-rank regularities in senone spaces, is a parameter independent of the senone class. Tuning σ changes the dimensions of the senone-specific subspaces in low-rank modeling. While σ = 100% leads to perfect reconstruction, i.e., no noise reduction, a very small σ might result in losing subspace specific information. Similar to the hyperparameter λ used in sparse recovery, we tune the value of σ for better ASR on a development set.

5.4.1.3 Overview of the Student-Teacher Training Approach Using Enhanced Soft Targets

The low-rank and sparse enhancement procedure described above requires ground-truth based forced senone alignments so that the correct senone-specific dictionary or eigenposterior matrix can be picked for enhancing each posterior frame. Thus, this procedure can be applied only to transcribed training data for which the forced alignments are readily available. We term this procedure as *supervised enhancement* of training data posteriors.

To enhance the posteriors of unseen (test) data where the alignments are missing, we propose an alternative approach. First, we train a DNN using training data acoustic features as input and the enhanced posteriors as soft targets. Enhanced soft targets are obtained using the dictionary or eigenposterior based reconstruction process as explained in the previous sections. Now, we can forward pass unseen test data through this newly trained DNN to get posteriors for ASR decoding. The new DNN can be considered as an enhanced student network which combines the knowledge from two sources:

1. Baseline teacher model, which has the acoustic modeling knowledge, and
2. Dictionaries/eigenposterior matrices, which store the knowledge of the low-dimensional senone subspaces.

Therefore, we argue that the student DNN learns to estimate the posterior probabilities on globally characterized low-dimensional subspaces.

In the next section, we give details of our experimental setup, results and subsequent analysis.

5.4.2 Database and Speech Features

The AMI corpus [McCowan et al. (2005)] contains recordings of spontaneous conversations between a group of participants in meeting scenarios. The meeting scenarios have been designed such that the participants freely discuss and debate over some ideas. Due to the conversational style of speaking and the speakers frequently overlapping and interrupting other speakers' speech, the AMI corpus has proved to be a challenging task in recent large vocabulary ASR research. In this chapter, we use the close-talk speech recordings in AMI from the individual headset microphone (IHM) setup. The dataset has nearly 100 hours of recordings divided approximately as 80 hours train set, 10 hours *dev* and 10 hours *eval* set. 10% of training data is used for cross-validation during DNN training in all cases, whereas the

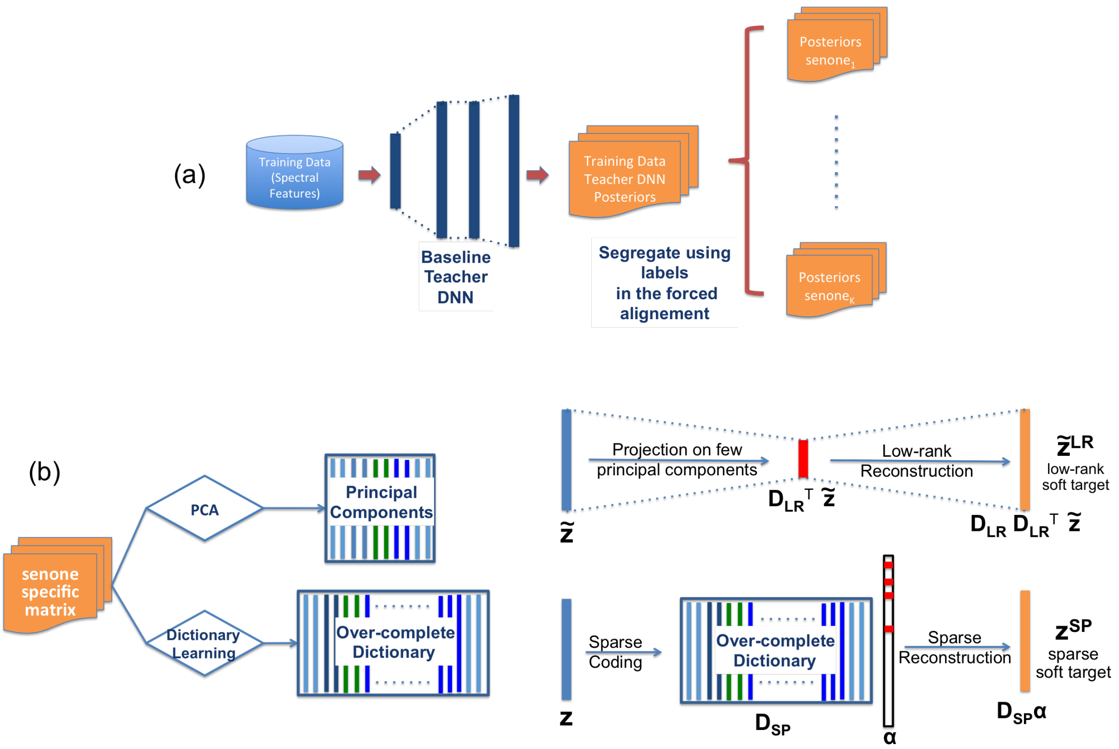

Figure 5.6: (a) Segregating DNN posteriors into senone-specific matrices. (b) Low-dimensional reconstruction of posterior features is done to achieve more accurate soft targets for enhanced DNN acoustic model training: PCA is used to extract eigenposteriors of the linear subspaces of individual senone classes. Sparse reconstruction over a dictionary is used for non-linear recovery of low-dimensional structures.

dev set is used to tune the σ and λ parameters discussed before.

The Kaldi toolkit [Povey et al. (2011)] is used for training the DNN-HMM systems. In this section, we use *tri2* Kaldi scripts where a context dependent senone set and subsequent GMM-HMM forced alignments are learned using MFCC+$\Delta + \Delta\Delta$ features. All DNNs have 9 frames of temporal context at input and 4 hidden layers with 1024 neurons each. Target dimension of DNNs correspond to 4007 senones and the input features have a dimension of 351 (39 dimensional MFCC+Δ+$\Delta\Delta$ features × 9 frame context). For dictionary learning and sparse coding, SPAMS toolbox [Mairal et al. (2014)] is used. All the results reported in this paper are reproducible using the standard AMI Corpus [McCowan et al. (2005)] setup, Kaldi toolkit [Povey et al. (2011)] and our scripts provided in [Dighe (2017)].

Details of Baseline Model Our baseline is a hybrid DNN-HMM system trained using forced aligned targets from a GMM-HMM system. The baseline, as well as all other DNNs, are randomly initialized and trained using cross-entropy (CE) loss backpropagation. We discuss the use of sMBR objective based sequence discriminative training later in Section 5.4.4. Word error rate (WER%) using the baseline system is 32.4% on AMI *test* set. We use this baseline network to generate the DNN posteriors which are then used to learn principal component matrices and dictionaries for sparse coding as shown in Figure 5.6. After enhancement of DNN posteriors, the soft targets thus obtained are used to train better acoustic models.

Another baseline is a student DNN trained using non-enhanced soft targets from the baseline. Non-enhanced soft targets refer to training data DNN posteriors which have been generated using the baseline DNN, but they were not enhanced using sparse or low-rank post-processing. This system gives a WER of 32.0%.

5.4.3 ASR Using Enhanced Student Models

For both dictionary learning and PCA based approach, we collected at most $N = 10^4$ posteriors from each senone to form senone specific data matrices. The analysis described in Section 5.2.1 revealed that the average rank of senone classes for AMI *tri2* setup based DNN posteriors is ∼ 37 for correctly classified posteriors. We set the dictionary size for sparse coding to be 500 columns big to fulfill the condition for the undercomplete dictionary that the dictionary size should be more than the dimension of the subspace being modeled. The procedure as depicted in Fig. 5.6(b) is implemented to generate sparse and low-rank soft-targets.

In order to make the soft targets based training of DNNs fast and feasible, we need to store the target senone probabilities for all the frames of training and cross-validation data on disk. In doing so, we encounter memory issues as soft targets for the complete training data require a significant amount of storage space (similar to [Chan et al. (2015)]). Hence, we preserve precision only up to the first two decimal places in soft targets, followed by normalizing each vector to sum 1 before storing the data on the disk. We assume that essential information might not be in dimensions with very small probabilities. Although such thresholding can be a compromise to our approach, we did some experiments with higher precision (up to 5 decimal places), but there was no significant improvement in ASR. Both low-rank and sparse reconstruction were still computed on full soft-targets without any rounding-off. We perform thresholding only when storing the final soft targets on the disk.

First, we need to tune the sparsity regularizer λ and the variability preserving low-rank reconstruction parameter σ for achieving better ASR performance in AMI *dev* set (shown in Table 5.4 and Table 5.5). A value of $\lambda = 0.1$ was found to be the optimal value for sparse reconstruction. When $\sigma = 70\%$ of variability is preserved using eigenposteriors, the most accurate soft targets are achieved for DNN acoustic modeling resulting in the smallest WER on the development set. It may be noted that in both low-rank and sparse reconstruction, there is an optimal amount of enhancement needed for improving ASR. While less enhancement leads to a continued presence of noise in soft targets, too much of it results in loss of essential information.

We then compare the ASR performance using DNNs trained with the new soft targets obtained from low-rank and sparse reconstruction (column Approach-0 of Table 5.6). System-1 is built by using senone posteriors from System-0 as soft targets for training the DNN acoustic model.

Table 5.4: Tuning λ for sparse modeling on AMI *dev* set.

λ	0.01	0.05	0.1	0.2	0.5	1.0
WER	29.7	29.5	**29.4**	29.6	29.8	30.0

Table 5.5: Tuning σ for low-rank modeling on AMI *dev* set.

σ (in %)	50	60	70	80	90	95
WER	29.7	29.7	**29.0**	29.5	29.4	29.5

These non-enhanced soft targets bring a small improvement of 0.4% in WER. In comparison, the supervised enhancement of soft outputs obtained from System-0 using PCA (System-2) reduces the WER by 1.2% absolute and dictionary-based System-3 achieves 0.8% absolute reduction in WER.

We also verified how the enhanced student models affect the lattice generation process in Kaldi and subsequently affect the ASR decoding. This analysis is required to confirm that the ASR performance does not improve simply because the beam search gets pruned differently in the case of low-rank and sparse soft targets based student DNNs. In all experiments, we used standard Kaldi *nnet1* decoding scripts with a settings of `-min-active=200` and `-max-active=7000` to constrain the number of active states at each frame. Along with this, a decoding `-beam=13` was used. During lattice generation[1], the pruning is governed by the minimum of maximum active states at each frame and the decoding beam. Same parameters were used for decoding in all experiments. An analysis experiment on a split of the development data showed that- with similar parameters, the pruning is governed by both the beam and the maximum active states without any observable patterns using different acoustic models. Specifically, the maximum active states are in the same range for different experiments.

5.4.4 Integration with Sequence Discriminative Training

Sequence discriminative training [Veselỳ et al. (2013)] enforces the acoustic model to learn utterance level sequential dependencies in the acoustic features such that the model prefers one particular alignment of senones over other competing ones. In contrast, the procedure of low-rank and sparse enhancement relies on the characterization of the senone level global dependencies in the DNN posteriors at the individual frame level. The enhancement process, while projecting the posteriors to class-specific subspaces, can result in breaking the sequential dynamics of the utterance initially present in the DNN posteriors. Thus, combining sequence discriminative training with our approach is not straightforward, and we discuss it in detail below.

We consider employing the sMBR objective for sequence discrimination which directly optimizes the DNN parameters to minimize the Bayes risk in state-level alignment [Veselỳ et al.

[1] More details are available at http://kaldi-asr.org/doc/lattices.html

Table 5.6: ASR performance on AMI IHM eval set (in WER%). Approach-0 shows improvements using our approach without sequence training. Approach-1 and Approach-2 provide ways to combine sequence discriminative training with low-rank and sparse enhancement approach. Right arrow '→' depicts the sequence of training/processing steps. Enhance refers to generating enhanced soft targets and subsequent cross-entropy loss based training of a student network.

Sys#	Training Targets	Approach-0	Approach-1	Approach-2
		CE→Enhance	CE → Enhance → sMBR	CE → sMBR → Enhance
0	Hard (Baseline)	32.4	29.6	29.6
1	Soft (Non-enhanced)	32.0	28.8	**29.4**
2	PCA ($\sigma = 70$)	**31.2**	28.3	30.7
3	Dictionary ($\lambda = 0.1$)	31.6	**28.2**	30.2

(2013)]. sMBR training can be incorporated with posterior enhancement in following two ways:

- Approach-1 (sMBR training after enhancement): A student DNN is first trained using the enhanced soft targets coming from cross-entropy trained baseline DNN. Sequence training using sMBR objective is then applied on this improved student network. This approach essentially shields the low-rank and sparse enhancement procedure from the effects of sequence discriminative training.

- Approach-2 (Enhancement after sMBR training): We can use the sequence discriminatively trained DNN as our baseline teacher acoustic model. Thus, we generate posteriors for training data using sMBR based teacher DNN and then use these posteriors to learn principal components or dictionaries for senone classes. Training data posteriors are then enhanced using eigenposteriors or dictionary to train an improved DNN acoustic model. The task requires that the student model not only learns to generate enhanced low-rank posteriors but also captures the sequence discrimination knowledge of the teacher DNN. Note that we use forced alignment from the sMBR based teacher DNN instead of GMM-HMM based alignments in this approach for PCA and dictionary-based enhancement. Also, the student model is trained for frame level classification using cross entropy loss.

Experimental results on the above two approaches are provided in Table 5.6. In Approach-1, sMBR based baseline DNN gives a WER of 29.6% as compared to 32.4% by the cross-entropy loss based baseline. Thus, we have an absolute reduction of 2.8% in WER just by employing sMBR sequence training. When we apply sMBR based sequence training on the best performing PCA and dictionary-based student DNNs, we observe significant performance gains of 2.9% and 3.4% absolute WER reductions respectively. This suggests that the gains in ASR performance from sequence discriminative training and low-rank or sparse enhancements are actually complementary to each other.

In Approach-2, System-0 is trained using sMBR objective and has no enhancement whereas System-1 to 3 are trained using cross entropy loss with soft targets from System-0. We observe performance improvement using non-enhanced soft targets based System-1 as compared to System-0 which shows that soft targets have the potential to capture the essence of sequence discrimination knowledge in them and transfer it to a student DNN (also confirmed in [Wong and Gales (2016)]). Next, System-2 and 3 are trained using PCA and dictionary-based enhanced soft targets respectively (with values of $\sigma = 70\%$ and $\lambda = 0.1$ which were tuned in 5.4.3 under Approach-0). These systems are unable to bring any improvements over the sequence trained baseline System-0 and the performance actually degrades. The best versions of System-2 and 3 under Approach-2 were in fact found to be operating at $\sigma = 100\%$ and $\lambda = 0.0$ respectively which trivially correspond to non-enhanced soft targets based System-1. We conclude from this observation that sequence discriminative training essentially modifies the senone subspaces and underlying senone correlations in such a way that PCA and overcomplete dictionaries are no longer capable of capturing them. Hence, it is not possible to improve the acoustic modeling by low-rank and sparse enhancements in this case. These experiments demonstrate that Approach-1 is the suitable strategy for complementary integration of sequence discriminative training with our enhanced soft targets based acoustic modeling to improve ASR performance.

5.4.5 Exploiting Untranscribed Data Using Semi-supervised Training

Given an accurate DNN acoustic model and some untranscribed input speech data, we can obtain soft targets for the new data through a simple forward pass. These additional soft targets can be used to augment our original training data. An important assumption here is that the given initial model can generalize well on the unseen data resulting in highly accurate soft targets. In this section, we propose to learn better DNN acoustic models using the augmented training set. This method is reminiscent of the *knowledge transfer* approach [Hinton et al. (2015); Chan et al. (2015)]. In our experiments, we keep the architecture of the DNN trained with augmented training set the same as the initially given DNN.

DNNs trained with low-rank and sparse soft targets are used to generate soft targets for ICSI corpus and Librispeech (LIB100) which are sources of untranscribed data. Table 5.7 shows interesting observations from various experiments using data augmentation. First, System-2 is built augmenting enhanced AMI training data with ICSI soft targets generated from System-1. We consider ICSI corpus, consisting of spontaneous speech from meeting recordings, as in-domain with AMI corpus. While PCA based DNN successfully exploits information from the additional ICSI data showing significant improvement from System-1 to System-2, the same is not observed using sparsity-based DNN. Next, System-3 is built by augmenting enhanced AMI data with Librispeech(LIB100) soft targets obtained from system 1. Read audiobook speech data from Librispeech is out-of-domain as compared to spontaneous speech in AMI. Still, System-3 achieves similar reductions in WER as observed in System-2 which was built using in-domain ICSI data.

Surprisingly, DNN soft targets obtained from sparse reconstruction are not able to exploit the unseen data in all the systems. We speculate that dictionary learning for sparse coding captures the non-linearities specific to AMI database. These nonlinear characteristics may correspond to channel and recording conditions which vary over different databases and cannot be transcended. On the other hand, the local linearity assumption of PCA leads to extraction of a highly restricted basis set that captures the most critical dynamics in the senone probability space. Such regularities mainly address the acoustic dependencies among senones which are generalizable to other acoustic conditions. Hence, the eigenposteriors are invariant to the exceptional effects due to channel and recording conditions. Sparse reconstruction can mitigate the undesired effects as long as they have been seen in the training data. We believe that sparse modeling might be powerful if some labeled data from unseen acoustic conditions are made available for dictionary learning. It may be noted that training with additional untranscribed data is not effective if non-enhanced soft targets are used. In fact, Systems 2-3 without low-rank or sparse reconstruction, perform worse than System-1 although they have seen more training data.

In literature, there have been recent improvements on ASR on AMI corpus. Notably, alignments generated using a speaker adaptively trained baseline GMM system (Kaldi *tri4a* setup) leads to a more competitive baseline hybrid DNN-HMM system that gives a WER of 29.6% on AMI IHM dataset [Renals and Swietojanski (2017)]. Further, a recent state-of-the-art lattice-free MMI training criteria over a time-delay neural network (TDNN) leads to significantly lower WER of 22.4% [Povey et al. (2016)].

5.5 Conclusions

In this chapter, we show how to explicitly model low-dimensional structures in speech using dictionary learning and sparse coding over the DNN posteriors. In spite of their power in

Table 5.7: Performance of various systems (in WER%) when additional untranscribed training data is used. System 0 is hard-targets based baseline DNN. In parantheses, SE-0 denotes supervised enhancement of DNN outputs from system 0 and FP-1 shows forward pass using System-1.

Sys#	Training Data	PCA(σ=70)	Sparsity(λ=0.1)	Non-Enhanced Soft Targets
0	AMI (Baseline WER **32.4%**)	-	-	-
1	AMI(SE-0)	31.2	31.6	32.0
2	ICSI(FP-1) + AMI(SE-0)	31.0	31.8	32.4
3	LIB100(FP-1) + AMI(SE-0)	**30.9**	31.7	32.4

representation learning, DNN based acoustic modeling still has room for improvement in 1) exploiting the union of low-dimensional subspaces structure underlying speech data and 2) acoustic modeling in noisy conditions. Using dictionary learning and sparse coding, DNN posteriors are transformed into projected posteriors which are shown to be more accurate. Sparse reconstruction moves the test posteriors closer to the subspace of the underlying senone class by exploiting the fact that the true information is embedded in a low-dimensional subspace, thus separating out the high dimensional erroneous estimates. Improvements in ASR performance are shown for both clean and noisy conditions on a small vocabulary task on Numbers database. The importance of low-dimension structures is further confirmed through RPCA analysis. The limitation of this approach is that it is non-trivial to scale its success to LVCSR tasks where the number of senone subspaces is too large and sparse recovery is not tractable.

To deal with the above limitation, we investigate the sparse and low-rank reconstruction of DNN posteriors in a "*one senone subspace at a time*" manner. Using senone-specific dictionaries and principal components, the training data DNN posteriors are transformed into projected posteriors which are shown to be more suitable targets for training better acoustic models. Improvements in ASR performance are achieved on large vocabulary ASR task on AMI meeting corpus. We also demonstrate that the performance gains from our enhancement approach and sequence discriminative training have different sources and they can be combined in a complementary way. Finally, sparse and low-rank modeling based enhancement is also found to be crucial for exploiting untranscribed data and further improving the acoustic model performance using semi-supervised training.

In the next chapter, we explore the applications of the approaches developed until this chapter to improve ASR in far-field conditions where speech is corrupted by noise, overlapping speech, and reverberation.

6 Applications of Sparse and Low-rank Modeling for Far-field Speech ASR

This chapter explores the utility of the approaches developed in this thesis in improving far-field speech ASR. Specifically, it focuses on enhancing the back-end and front-end of far-field ASR by exploiting low-dimensional subspace information learned from close-talk speech data. The chapter is organized as follows. Section 6.1 presents an approach to improve far-field DNN acoustic models by using enhanced soft targets from parallelly recorded close-talk data. Section 6.2 focuses on far-field speech enhancement using sparse and low-rank models. Section 6.3 concludes the results of this chapter.

6.1 Far-field ASR Using Enhanced Soft Targets from Parallel Data

In this section, we focus on improving the back-end of far-field speech ASR. Training accurate DNN acoustic models using far-field speech is often considered a challenging task due to the poor quality of framewise senone alignments available with it. For example, a distant microphone might pick up strong background speech or other additive noise, and align spoken words in the transcription with these unintended regions [Peddinti et al. (2016)]. These effects degrade the quality of target senone alignments which in turn results in poor DNN based acoustic modeling. A common way to tackle this problem is to parallelly record speech data using close-talk microphones and use close-talk speech to generate better quality senone alignments [Qian et al. (2016)] (as shown in Figure 6.1). DNN acoustic models trained with clean alignments from parallel data have been consistently shown to outperform models which use alignments from the far-field data [Qian et al. (2016); Himawan et al. (2015)].

In this work, we extend the parallel data approach by exploring the use of enhanced soft targets for training far-field acoustic models. We had shown in Chapter 5, that enhanced soft targets are successful in improving close-talk ASR performance. Low-rank reconstruction using *eigenposteriors* and sparse reconstruction using overcomplete dictionaries [Candès and Wakin (2008)] are principled ways of preserving the global low-dimensional structures in soft targets while discarding the random high-dimensional noise. In this chapter, we use the sparse and low-rank soft targets obtained from a close-talk ASR system to train DNN acoustic models

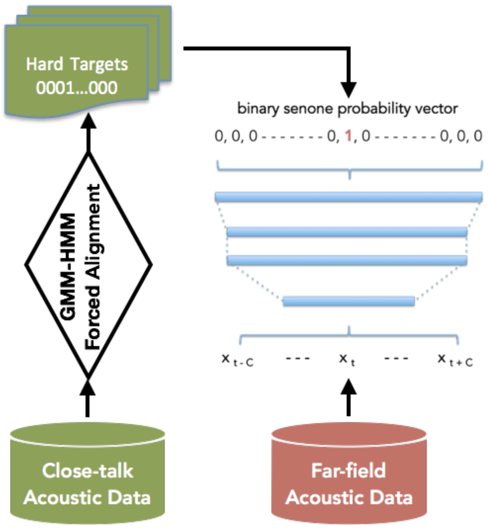

Figure 6.1: Using hard alignments from parallel close-talk speech data to train DNN acoustic models for far-field speech.

for far-field speech.

Prior research in far-field ASR using parallel data can be categorized into front-end and back-end based approaches. In front-end approaches [Qian et al. (2016); Du et al. (2014); Sim et al. (2017)], the far-field acoustic features are first enhanced by mapping them to parallel close-talk features. The enhanced acoustic features are then used for ASR. We explore this approach in Section 6.2. In contrast, the back-end approaches focus on employing stronger acoustic models like CNN [Swietojanski et al. (2014)], LSTM-RNN and their variants [Zhang et al. (2016); Kim et al. (2017)], or adapting the back-end model by knowledge sharing [Qian et al. (2016); Kim et al. (2018)] with a parallel close-talk based acoustic model. Another common approach, as discussed earlier, is to use hard alignments from clean speech data. This approach has been explored successfully in [Peddinti et al. (2016); Himawan et al. (2015); Weninger et al. (2015)].

6.1.1 Motivation and Contribution

Our motivation for using enhanced soft targets for learning far-field DNN acoustic models is twofold. Firstly, in a reverberated speech signal, the acoustic realization of a senone would be continuously smudged by the presence of neighboring senones. Hence, any acoustic feature frame of reverberated speech can possibly have evidence of multiple senones which would actually appear in a comparatively more discrete sequence if the speech was captured using a close-talk microphone. This suggests an increased amount of temporal correlation exhibited by senones in the acoustic feature space of far-field speech. As argued in Chapter 5, such temporal correlation among senones is better characterized by soft targets as they are obtained by processing a context of neighboring acoustic feature frames at the input of the close-talk DNN.

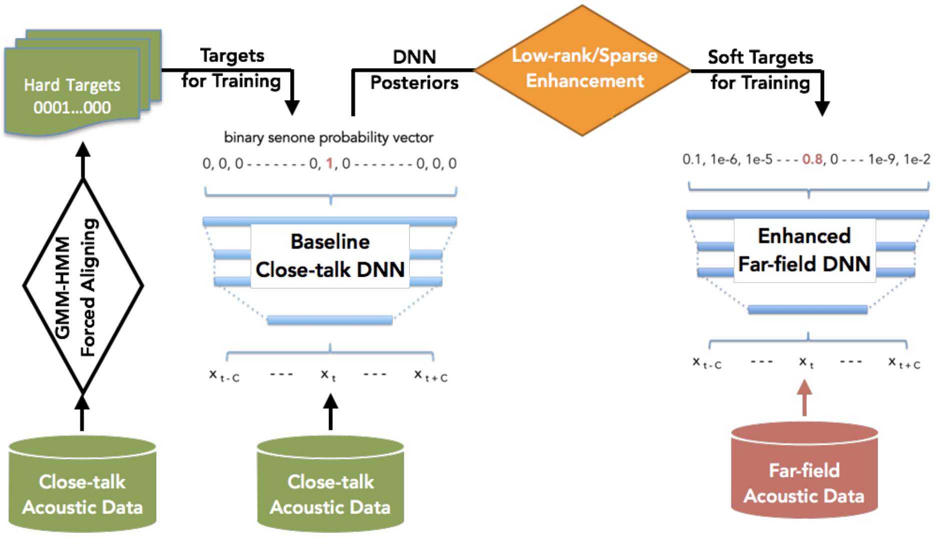

Figure 6.2: Schematics of our system which uses low-rank and sparse soft targets for training the far-field DNN acoustic models. Required soft targets are obtained by PCA or dictionary based enhancement of close-talk speech DNN posteriors.

Secondly, as shown in [Peddinti et al. (2016)], far-field acoustic features might lead to a choice of different pronunciations for the same word transcription. In such a case, it will be preferable to have soft targets as DNN outputs so as to support possibilities of multiple phonetic sequences rather than hard alignments which enforce one particular pronunciation of the underlying word sequence. Finally, we need the soft targets not to associate with unstructured local noise in the far-field acoustic features. This motivated us to work with enhanced low-rank and sparse soft targets which primarily focus on the intra-class global patterns and the inter-class correlations rather than local erroneous probability estimates present in the original DNN posteriors.

Experimental evaluations are conducted on the AMI corpus [McCowan et al. (2005)] which provides audio recordings that were parallelly recorded using close-talk and distant microphones. This provides a perfect use case for our experiments on improving far-field ASR. We show in Section 6.1.2.2 that low-rank and sparse soft targets lead to improved ASR performance using DNN, long short-term memory networks (LSTM) as well as time-delay neural networks (TDNN) acoustic models. We achieve nearly 4-5% absolute WER reduction as compared to the traditional far-field data based DNN baseline.

6.1.2 Improving Far-field ASR Using Senone Subspace Modeling

We present our system for far-field acoustic modeling in Figure 6.2. Instead of using hard targets from parallelly recorded close-talk speech data, we are using low-rank and sparse soft-targets. First, a close-talk acoustic model is trained traditionally (shown as baseline close-talk DNN in Figure 6.2) with hard targets obtained from GMM-HMM forced alignments. The close-talk DNN thus trained is used to generate soft targets from the close-talk speech features. These soft targets are then passed through a PCA or dictionary based enhancement process as explained in Section 5.4.1 to generate enhanced soft targets. The enhanced soft targets are used with the far-field speech to train more accurate DNN acoustic models (shown as enhanced far-field DNN in Figure 6.2). Below we describe the details of our ASR experiments and the subsequent analysis to evaluate the performance of our approach on far-field ASR.

6.1.2.1 Database and Features

We demonstrate our approach on the AMI corpus. Single distant microphone (SDM) data with mic-id 1 is used in our experiments for far-field speech and individual headset microphone (IHM) as the source of close-talk speech. The rest of the details for this database can be found in Section 5.4.2.

The Kaldi toolkit [Povey et al. (2011)] is used for training DNN-HMM systems. The input features to the DNN have a dimension of 1320 (40-dimensional log-Mel filterbank energies+Δ+$\Delta\Delta$ features × 11 frame context). Senone set generated using IHM data consists of 3992 senones and those generated using SDM data consist of 3932 senones. All DNNs have 6 hidden layers with 2048 neurons each. The experiments are based on the Kaldi *tri3b* system where the senone set and the subsequent GMM-HMM forced alignment are learned after LDA+MLLT transforms[1] [Rath et al. (2013)]. All DNNs are randomly initialized and trained using cross-entropy (CE) loss backpropagation followed by sequence discriminative training to minimize the sMBR objective. For sequence training using sMBR loss, the alignments and denominator lattices are generated using the CE trained DNNs. AMI pronunciation dictionary has ~47K words and a trigram model for decoding. All the results reported in this paper are reproducible using the standard AMI Corpus [McCowan et al. (2005)] setup, Kaldi toolkit [Povey et al. (2011)] and scripts provided in [Dighe (2017)].

For generating low-rank and sparse soft targets, a value of $\sigma = 95\%$ and $\lambda = 0.1$ was found to be optimal while optimizing WER on *dev* set. Setting $\sigma = 95\%$ results in a different number of principal components being retained for different senone classes. The average number of retained principal components over all classes was found to be ~40 as compared to the overall dimension of 3992 senones.

Experiments based on long short-term memory (LSTM) and time-delay neural network (TDNN) (Section 6.1.2.2) are based on standard recipes and parameter settings from Kaldi *nnet3* scripts. Both LSTM and TDNN are trained using the same input Fbank features and output senone labels as the baseline DNN acoustic model. Some architectural details of these models follow here. The LSTM and bidirectional(Bi-) LSTM use a recurrence of 20 time steps for back-propagation and have 3 hidden layers each of size 1024. Splicing is done at the input

[1] This additional feature transformation step renders Kaldi *tri3b* setup different from the *tri2* setup which was used in the experiments discussed in Chapter 5. Also, we use filterbank energies instead of MFCC features in this chapter.

Table 6.1: ASR performance on AMI SDM eval set (in WER%) when soft targets are derived from eigenposteriors and dictionaries learned using SDM senone set and corresponding baseline DNN.

System #	Training Targets	Network Type	
		CE	CE+sMBR
1.1	SDM (Hard)	58.6	54.4
1.2	SDM (PCA)	**57.9**	**53.1**
1.3	SDM (Dictionary)	60.8	55.7

to include a left and right context of 2 frames each and delta features are not appended. TDNN acoustic model is based on [Peddinti et al. (2015)] and uses layerwise splicing of {-2,2 ; -1,2 ; -3,3 ; -7,2 ; -3,3 ; 0 ; 0}. Each ';' separated pair of numbers gives the left (with '-' symbol) and right context for splicing at each successive layer of the TDNN model. Similar to LSTM models, we do not append delta features to Fbank features at the input of TDNN. Our LSTM and TDNN setup is more comparable to the previous works presented in [Swietojanski et al. (2013)] and [Peddinti et al. (2015)]. In contrast to the system in [Peddinti et al. (2016)], we do not employ speaker-adaptive training for acoustic modeling.

6.1.2.2 Experimental Analysis

We use the SDM eval set for evaluation here. Scoring is done using NIST asclite tool [Fiscus et al. (2006)] for up to 4 overlapping speakers. Our initial results, shown in Table 6.1, are entirely based on SDM data. ASR word error rates (WER %) are provided for DNN acoustic models which are first trained with CE loss and then subsequently sequence discriminatively trained with the sMBR loss. The first row depicts a traditional baseline system (System 1.1) trained using far-field acoustic features with hard alignments from SDM which works at 54.4% WER with sequence training. We enhance the soft targets generated from System 1.1 using PCA and sparse coding to train System 1.2 and 1.3 respectively. While PCA based System 1.2 gives a small (1.3% reduction in WER) performance improvement, sparse coding-based System 1.3 turns out to perform even worse than the hard target based baseline itself. We noticed this performance degradation over a range of values for σ and λ for ASR on the *dev* set as well. This experiment confirms the poor quality of SDM based senone alignments as well as the DNN in System 1.1 which generated the soft targets. We conclude that our approach is not able to learn meaningful senone subspace information with SDM senone set and SDM based DNN posteriors. Next, we do experiments with soft targets from IHM data.

The baseline system in the experiments with parallel data uses hard targets from IHM close-talk data as shown in Figure 6.1. Table 6.2 depicts this as System 2.1. In these experiments, we also provide results for ASR on IHM data to compare how ASR performance improvements on IHM data relate to those on SDM data. These results are based on Kaldi *tri3b* setup, and are different from the *tri2* setup experiments conducted in Chapter 5. As expected, System 2.1

with IHM hard targets performs better than System 1.1 which uses SDM hard targets. We use System 2.1 (CE loss-based IHM system) to generate soft targets which are enhanced and used to train System 2.2 and 2.3. The original soft targets didn't bring any significant improvements on IHM data in Chapter 5 and we do not consider them here. On IHM data, we notice that PCA soft targets based System 2.2 performs the best at 26.8% WER with sequence training. Although sparse soft targets based System 2.3 outperforms the IHM hard target baseline, the improvements are still lower than PCA based System 2.2. However, on far-field SDM data, both System 2.2 and 2.3 give significant WER reductions, and System 2.3 with sparse soft targets outperforms System 2.2 trained using PCA based soft targets. Compared to the SDM hard target based sequence trained baseline, the overall improvement by using System 2.3 is 4.4% absolute (~8% relative) and compared to IHM hard targets, it is 2.1% absolute (~4% relative).

An interesting observation here is that the sparse soft targets result in better acoustic modeling than their low-rank counterparts for SDM data, whereas we observe the contrary on IHM data. The success of sparse soft targets for SDM shows that the non-linear low-dimensional modeling of senone subspaces, enabled by dictionaries, is highly beneficial for mapping reverberated noisy speech acoustic features to underlying senone classes. We also note that the performance improvements using enhanced soft targets are observed in both CE and sMBR loss based systems, and we conclude that the benefits of enhanced soft targets are complementary to those of sequence training, as shown previously in Chapter 5.

In Table 6.3, we further evaluate our approach on state-of-the-art recurrent and time-delay neural network architectures. We observe in Table 6.3 that the enhanced soft targets are superior for training the LSTM and TDNN based acoustic models than IHM hard targets. The WER reductions are noticeably smaller for these strong baselines, but we consistently achieve ~5% absolute improvement in WER as compared to the SDM hard targets baseline, and ~1% absolute improvement as compared to IHM hard targets based systems. Bi-LSTM based System 3.2 and 3.3 with low-rank, and sparse targets perform equally well and give the best WER of 49.3%. These experiments further confirm the importance of modeling low-dimensional senone subspaces for improving ASR. Note that the low-rank and sparse soft targets from the parallel IHM data were still obtained from CE loss based IHM System 2.1

Table 6.2: ASR performance on AMI IHM and SDM eval set (in WER%) when soft targets are derived from eigenposteriors and dictionaries learned using IHM senone set and the corresponding IHM DNN acoustic model.

System #	Training Targets	Training & Evaluation Data			
		IHM		SDM	
		CE	CE+sMBR	CE	CE+sMBR
1.1	SDM (Hard)	-	-	58.6	54.4
2.1	IHM (Hard)	30.5	28.0	54.9	52.1
2.2	IHM (PCA)	**29.4**	**26.8**	52.9	51.5
2.3	IHM (Dictionary)	30.4	27.3	**52.1**	**50.0**

Table 6.3: ASR performance using recurrent and time-delay NN architectures on AMI SDM eval set (in WER%) when soft targets are derived from eigenposteriors and dictionaries learned using IHM senone set and the corresponding IHM DNN acoustic model.

		Network Type		
System #	Training Targets	LSTM	Bi-LSTM	TDNN
1.1	SDM (Hard)	54.9	54.2	55.0
3.1	IHM (Hard)	51.3	49.7	51.0
3.2	IHM (PCA)	**50.1**	**49.3**	**49.8**
3.3	IHM (Dictionary)	50.2	**49.3**	50.2

depicted in Table 6.2. A recent state-of-the-art lattice-free (LF-) MMI training criterion over a time-delay neural network (TDNN) leads to significantly lower WER of 46.1% [Povey et al. (2016)] on AMI SDM dataset without using parallel IHM dataset. Application of our low-rank and sparse modeling approach on LF-MMI training of acoustic models is an open area of research.

6.2 Far-field Speech Enhancement Using Sparse and Low-rank Modeling

In this section, we focus on improving the front end for far-field speech ASR. One of the primary reasons for the degradation of ASR performance on far-field speech is the poor quality of the captured acoustic signal. A distant microphone is prone to capturing noise, overlapping speech as well as reverberations. Hence, spectral features generated from noisy far-field speech are far inferior in quality then their counterparts generated using clean close-talk microphone speech. This is evident from Table 6.2 where ASR performance using SDM condition data gives a WER of 54.9% as compared to a WER of just 30.5% for IHM condition. Both these systems use IHM forced alignments as targets for training the acoustic models. But, the difference in the quality of input acoustic features leads to a substantial difference in the ASR performance. Figure 6.3 contrasts the spectrograms of a speech utterance when it is captured using a close-talk microphone versus using a distant microphone. The far-field spectrogram is visibly more noisy than the close-talk case as the speech information is heavily corrupted in it due to additive and convolutional noise. The front end approaches for improving far-field ASR concentrate on enhancing the quality of far-field speech features by removing this unwanted noise and effects of reverberation. These approaches collectively fall into the category of *speech enhancement* techniques.

A set of speech enhancement techniques aim to use deep neural networks to map far-field speech features to parallelly recorded time-synchronous close-talk speech features [Xu et al. (2014); Qian et al. (2016); Giri et al. (2015); Gao et al. (2015); Chen et al. (2015); Du et al. (2014); Mimura et al. (2015)]. In context of ASR, such techniques typically have a front-end speech enhancer (SE) DNN and a back-end acoustic modeling (AM) DNN. The acoustic modeling

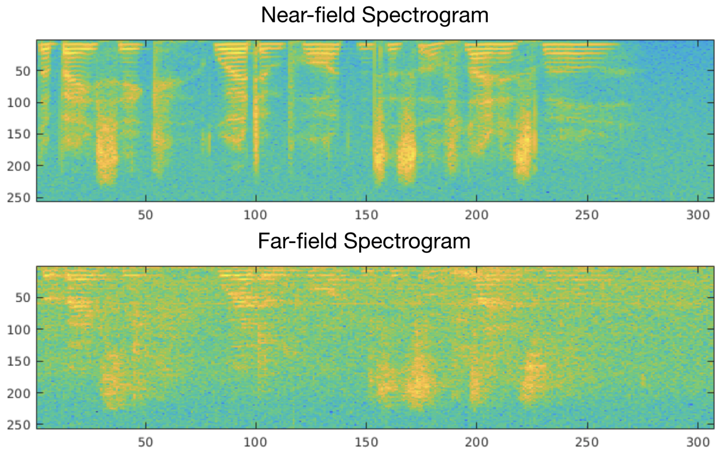

Figure 6.3: Comparing a close-talk speech spectrogram with a far-field speech spectrogram. Due to noise and reverberation, the far-field features are an erroneous and redundant version of the close-talk features - which are clean and low-rank.

process benefits from the front-end speech enhancement operation which results in improved ASR performance.

While a variety of architectures have been proposed in the previous literature [Qian et al. (2016); Gao et al. (2015)] for integrating the SE and AM DNN, the most common approach combines them in a serial order where the outputs of the SE DNN are fed as inputs to the AM DNN. Both the networks in this architecture can be either 1) trained separately or 2) stitched together and trained jointly under a multi-task learning framework. Figure 6.4 depicts a typical joint training architecture for the speech enhancement and acoustic modeling. The first few hidden layers map the far-field features to close-talk speech features, and the later layers solely focus on predicting the senone targets for subsequent ASR decoding. One of the intermediate hidden layer acts as the output layer for a regression task where a mean squared error loss $\mathcal{L}_{SE}$ is computed with respect to the clean speech features. On the other hand, the final layer of the network focuses on the senone classification task using a cross entropy loss $\mathcal{L}_{AM}$. The network is trained to jointly minimize a combination of both the losses as:

$$\mathcal{L}_{\text{Joint}} = \mathcal{L}_{SE} + \alpha \mathcal{L}_{AM} \tag{6.1}$$

where α controls the ratio of the two losses and decides the importance of each task during the training of the network. Once the network is trained, the outputs of the final layer are used for ASR, and the outputs of the intermediate layer can be ignored.

It is important to note that the speech enhancement DNN expects one-to-one mapping between the far-field features and close-talk features. Typically parallel far-field and close-talk datasets can be obtained in two ways: either by simultaneously recording speech using close-talk as well as distant microphones or by creating artificial far-field speech by corrupting close-talk clean speech data with additive and convolutional noise. Therefore, the DNN-based

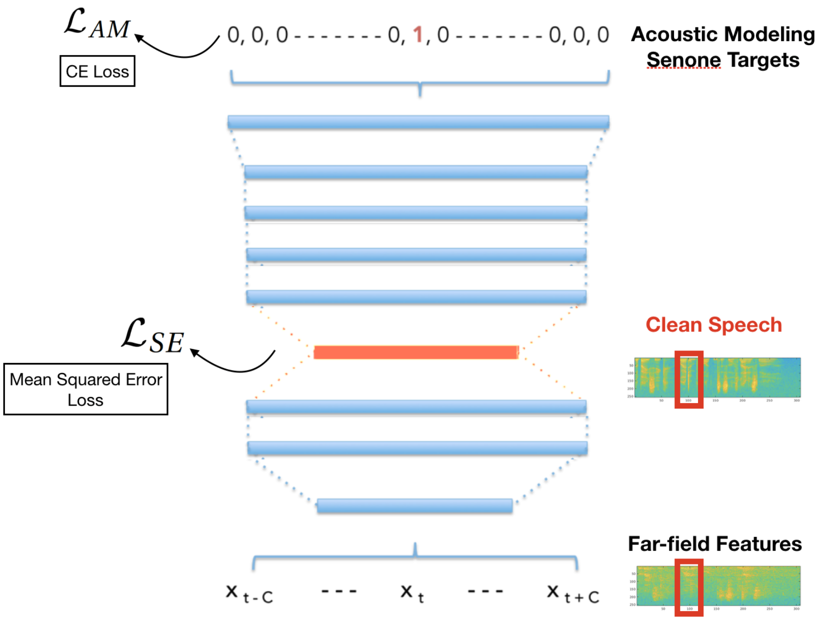

Figure 6.4: Joint speech enhancment and acoustic modeling (JSEAM) DNN architecture under a multi-task learning framework. An intermediate hidden layer performs a regression task for speech enhancement and the final hidden layer performs a classification task for acoustic modeling.

speech enhancement approach depends on the availability of an equal amount of clean speech data as the corrupted far-field data.

In this thesis, we explore the task of speech enhancement for improving ASR as a use case for our sparse and low-rank modeling approach. Specifically, we propose to enhance far-field speech features by projecting them on the low-dimensional senones subspaces learned from close-talk speech features. Enhanced features, thus obtained, are used as targets for the speech enhancement task under the architecture presented in Figure 6.4. In the following sections, we motivate our approach and provide its experimental evaluation.

6.2.1 Motivations and Our Approach

If a speech signal is simultaneously recorded using a close-talk microphone and a distant microphone, this thesis hypothesizes that- 1) the speech related acoustic information, which is crucial for ASR, lives in low-dimensional subspaces, and 2) although this information is readily accessible in close-talk features, it is densely superimposed with high-dimensional noise in the case of far-field features, thus making it obscure.

The reason behind the above hypothesis is as follows. A far-field signal gets reverberated due to its reflections from the surrounding environment. As discussed in Section 6.1.1, the acoustic realization of speech in such a reverberated signal would be continuously smeared by the

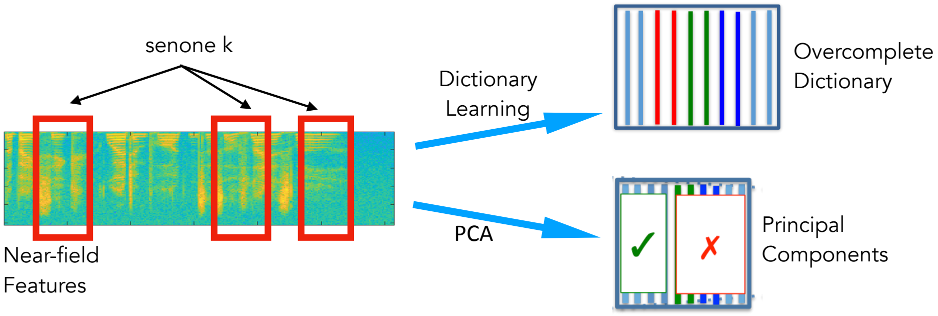

Figure 6.5: Learning low-dimensional senone-specifc subspaces from the clean close-talk speech using dictionary learning and PCA.

presence of incoming reflections and other sources of additive noise. This effect conceals the low-dimensional speech information by smudging it across time and frequency dimension. As a result, the captured signal will be noisy and high-dimensional, whereas the actual speech information would be in low-dimensional subspaces. In other words, our hypothesis considers the close-talk signal as a low-rank representation of the parallelly recorded far-field signal. Therefore, we refer to the near-field spectrogram depicted in Figure 6.3 as a low-rank enhanced copy of the noisy far-field spectrogram.

Our hypothesis naturally leads us to seek ways to project the high-dimensional far-field speech data onto the underlying low-dimensional subspaces which contain the acoustic information relevant for ASR. Close-talk speech features provide direct access to these low-dimensional subspaces as they are relatively much cleaner than the far-field speech features. Since the end goal is to improve ASR, we propose to learn these low-dimensional subspaces at the hierarchy of senones (as done in Chapter 5 in the context of posterior features). Our approach is depicted in Figure 6.5 and Figure 6.6.

We learn senone-wise dictionaries and principal components from close-talk speech spectral

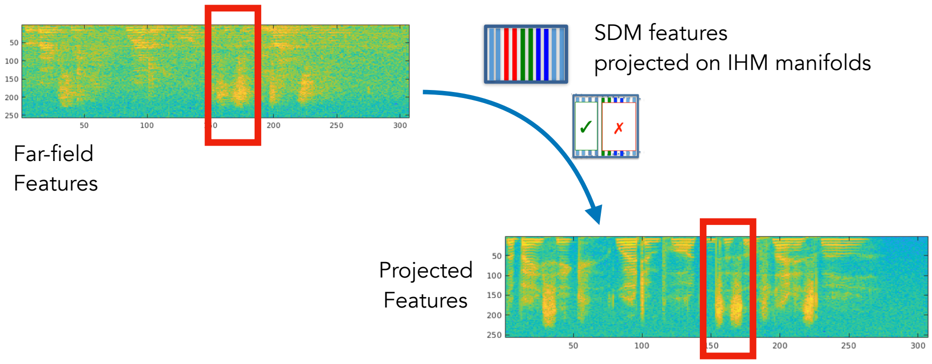

Figure 6.6: Projecting noisy and high-dimensional far-field speech on the low-dimensional manifolds learned from the close-talk speech data.

features in a similar way as done for posterior features in Chapter 5. These methods model the frequent regularities in the data as the senone-specific acoustic information, whereas local errors and random noise is discarded. Close-talk speech features act as a clean and reliable source of data to learn the sparse and low-rank models. Also, we need the GMM-HMM based ground truth alignments for segregating the acoustic features before applying PCA or dictionary learning algorithm.

In the next step, we project the far-field acoustic features on the low-dimensional subspaces modeled by the close-talk speech dictionaries and principal component matrices. This step again requires the senone labels from the forced alignment for supervised enhancement of the far-field acoustic features. Therefore, we enhance only the labeled training data using this approach. Since dictionary and PC projection reduce the rank of the data by only preserving the class-specific information, we expect the projected features to be enhanced in quality. It is important to note that the projection process relies solely on the senone subspace information modeled by the dictionaries and the principal components; it does not try to map the far-field feature to the parallelly measured close-talk features. In the final step, we use the projected features in the joint speech enhancement acoustic modeling network described before in Figure 6.4. We expect the projected features to act similarly as the parallelly recorded clean speech features.

6.2.2 Experimental Evaluation

Parallel data from the AMI corpus as described in Section 6.1.2.1 is used in the experiments here. For speech enhancement, we use 40-dimensional log-Mel filterbank energy (Fbank) features. The close-talk features are spliced with a context of 5 neighbors frames for left and right resulting in a 440-dimensional feature vector. Dictionaries and PC matrix are learned on these context-appended vectors. Online dictionary learning algorithm (Section 3.2) and LARS Lasso algorithm (Section 3.3) are used for learning dictionaries and sparse recovery respectively. A value of $\lambda = 0.1$ and $\sigma = 80$ was found optimal for sparse and low-rank reconstruction of far-field acoustic features. After reconstruction, we retain only the center frame of the context-appended projected acoustic feature. Using the GMM-HMM forced alignments and close-talk speech dictionaries/PC matrices, we generate the projected acoustic features for all the far-field training data.

DNN architecture used for joint speech enhancement and acoustic modeling (JSEAM) is as follows. The input layer has a dimension of 1320 (=40 Fbank+Δ+$\Delta\Delta$ features × 11) due to a context of 5 frames from left and right. The first two hidden layers map the input layer to an intermediate hidden layer which has the same dimension as the input layer. This intermediate layer backpropagates a mean squared error loss with respect to the projected features. The output of the intermediate layer is processed by 4 more hidden layers which are connected to the final output layer of dimension 3992. The final output layer predicts senone classes. We employ IHM hard targets as well as low-rank and sparse soft targets for training the acoustic

Table 6.4: ASR performance using the joint speech enhancement-acoustic modeling (JSEAM) approach on AMI SDM eval set (in WER%). Results are shown for different combinations of speech enhancment and acoustic modeling training targets which are used during the training of the JSEAM network.

	AM Targets		
SE Targets	IHM(Hard)	PCA ($\sigma = 95$)	Dictionary ($\lambda = 0.1$)
No enhancement	54.9	52.9	52.1
IHM	53.2	-	-
SDM PCA Proj. ($\sigma = 80$)	53.4	52.3	-
SDM Dictionary Proj. ($\lambda = 0.1$)	53.2	-	**51.7**

modeling part of the network. In all experiments, cross entropy loss is used at the final output layer. All hidden layers have a dimension of 2048 nodes. The intermediate output layer is connected linearly to the next hidden layer. The speech enhancement part of the network is first trained independently followed by joint training of both the networks.

Table 6.4 provides the results of speech enhancement based ASR experiments. We train a variety of JSEAM networks based on different training targets for SE and AM tasks. For SE task, close-talk IHM features are compared with the PCA and dictionary-based projected far-field features. For AM task, IHM forced alignments are compared with sparse and low-rank enhanced soft targets. As expected, speech enhancement helps improve the ASR performance in all of the cases. Sparse modeling based network which uses dictionary reconstruction for both SE and AM performs the best at 51.7% WER, thus giving a 1.5% absolute reduction over an equivalent IHM-based JSEAM network which performs at 53.2% WER. We conclude that the projected acoustic features act as a suitable replacement for close-talk features as we get similar or even better WER reductions using the dictionary and PC-based projected data. This observation suggests that the speech enhancement procedure using DNNs does not rely on the actual close-talk speech features. Projected features can result in equally good speech enhancement. In fact, the true acoustic information in close-talk features is embedded in low-dimensional subspaces which can be efficiently modeled using dictionary learning or PCA.

A unique feature of our approach is that we do not require the whole parallelly recorded close-talk dataset to generate the projected features. Dictionaries and PCs are computed using a limited amount of acoustic features for each senone class, after which the rest of the clean speech data can be discarded. Nevertheless, the alignments generated using the close-talk speech features are still needed for the supervised enhancement of the far-field speech.

6.3 Conclusions

In this work, we presented an application of our sparse and low-rank modeling approach to improve far-field speech ASR. To improve the back-end, we use low-rank and sparse soft targets from parallelly recorded close-talk speech to enhance the DNN acoustic modeling for far-field speech. PCA and dictionary learning encode the low-dimensional senone subspaces present in DNN posteriors of a close-talk ASR system. Enhanced soft targets prove to be better than hard targets from close-talk speech. Gains in ASR performance using sparse soft targets are particularly promising and suggest a potential for exploring sparse modeling based techniques to improve far-field ASR.

We also explore the application of low-dimensional senone subspace modeling in speech enhancement. Under this approach, far-field acoustic features are projected on low-dimensional senone specific subspaces towards the goals of enriching the acoustic information in them and discarding undesired noise. Through this process, far-field features are brought close to the manifold where close-talk features live. Using a joint speech enhancement acoustic modeling network, we exploit the projected far-field acoustic features to improve the performance of far-field speech ASR. It is shown that the clean acoustic information present in close-talk speech can be effectively compressed in the form of senone-specific dictionaries and principal components.

In the next section, we develop an information theoretic analysis technique to quantify the quality of any acoustic model under an HMM-based ASR framework. We use this technique to examine the reasons why sparse and low-rank modeling of speech is indeed vital for improving ASR.

7 On Quantifying the Quality of Acoustic Models in Hybrid DNN-HMM ASR

This chapter studies the quality of acoustic models in terms of how well they comply with HMM assumptions. We quantify the acoustic model quality in information theoretic terms and relate it with their performance on speech recognition tasks. The information theory based analysis technique developed in this chapter is also used to understand why sparse and low-rank modeling of speech helps in improving HMM-based ASR.

This chapter is organized as follows. Section 7.1 motivates this work and briefly introduces our approach. Section 7.2 provides background on HMM-based speech recognition and discusses the relevant previous research in detail. Section 7.3 introduces a speaking-listening perspective to the process of ASR using a novel z-HMM formulation. Section 7.4 introduces the information theoretic analysis technique using a novel z-HMM formulation. Section 7.5 describes the experimental results and analysis of our findings. Finally, Section 7.6 concludes our work in this chapter.

7.1 Motivations and Our Approach

Over the last 40 years, hidden Markov models (HMMs, and more recently DNN-HMM) have served as the backbone of virtually all large-scale ASR systems [Jelinek (1976); Rabiner (1989); Jelinek (1997)]. However, HMMs are built upon several major assumptions which are well known and understood, yet often shattered in the speech community [Bourlard and Morgan (1994); Bilmes (2006); Gillick et al. (2011)].

More specifically, the HMM theory relies on the following assumptions. Firstly, the probability distribution associated with a hidden state depends only on that state. Therefore, the acoustic observation is conditionally independent of all the rest given the underlying hidden state. Secondly, HMMs used for modeling speech are typically first order, i.e., the probability of the Markov chain to be in a particular state at a time step depends solely on the previously visited state and nothing else. Although the first order Markovian assumption makes HMM computations tractable, it limits the scope of capturing temporal dependencies. Furthermore, traditional HMM-based sequence modeling requires an a-priori definition of the state conditional probability distributions, which were often considered as mixtures of multivariate Gaussians in GMM based HMM systems.

In modern hybrid DNN-HMM architectures, the DNNs make no such assumption about the statistical distribution of the observations and directly estimate the state-specific posterior probabilities[1] conditioned on some limited temporal context. Replacing GMM acoustic models by DNNs under the tandem [Hermansky et al. (2000)] or hybrid ASR approach [Bourlard and Morgan (1994)] has been the single largest source of ASR performance improvement in the last few years [Hinton et al. (2012)]. This leap in performance achieved by using DNN acoustic models necessitates the study of the following fundamental questions:

> *Does DNN based acoustic modeling specifically fulfill the HMM assumptions better than GMMs? And if so, can we formally identify some properties desired in an acoustic model for improving ASR performance?*

The work presented in this chapter is an attempt towards answering the above question by deriving an information theoretic analysis framework for analyzing DNN-HMM speech recognition. The core analysis depends upon conceptualizing the traditional HMM formulation in a different manner. Instead of treating a continuous multi-dimensional acoustic feature (e.g., an MFCC vector) as an observation, we do the following. We use the emission probabilities predicted by the acoustic model to compute the gamma posterior probabilities of HMM hidden states using the forward-backward algorithm [Rabiner (1989)]. Based on the gamma posterior vectors, we predict categorical values for the hidden states at each time step. These categorical predictions are treated as discrete observable features emitted by the actual underlying hidden HMM states. We term this modified HMM framework as z-HMM (further details in Section 7.3). Proposed z-HMM formulation facilitates the computation of some useful information theoretic terms which are otherwise highly non-trivial to compute using acoustic features. These information theoretic measurements allow us to analyze the quality of acoustic models without computing full-fledged ASR-related measurements like frame classification accuracy or word error rate (WER). We talk in detail about the contributions of this chapter in Section 7.2.2. The important notations for the mathematical expressions are listed in Table 7.1.

7.2 Background and Prior Research

Speech is a complex time-varying signal which is usually assumed as resulting from a piecewise stationary stochastic process so that the observed signal can be modeled by a hidden Markov model. HMM is a probabilistic finite state model in which the state-specific emission

[1] DNN based state posterior probabilities, conditioned on a local context, simply provide an ad-hoc way of computing emission probabilities of HMM states (or data likelihoods) as they are divided by the state prior probabilities before decoding. We note that these state posterior probabilities are different than the gamma posterior or state occupancy probabilities for hidden states which can be computed by full forward-backward algorithm [Rabiner (1989)].

Table 7.1: Notations and their associated meaning.

Notation	Indication
$\mathbb{Q} = \{q_1, \ldots, q_k, \ldots, q_K\}$	Set of discrete values that a HMM hidden state can take, could refer to set of senones which are physical HMM states obtained after tying of logical states.
Q_t	Hidden state random variable at time index t which takes values from the set $\mathbb{Q}$.
$\mathcal{Q} =< Q_1, \ldots, Q_t, \ldots, Q_T >$	Sequence of HMM hidden states underlying an utterance.
X_t and $\mathbf{x}_t$	Acoustic feature random variable and its value $\in \mathbb{R}^n$ denoting the acoustic observation at time t.
$\mathcal{X} =< X_1, \ldots, X_t, \ldots, X_T >$	Sequence of acoustic features observed for an utterance.
Z_t	Random variable denoting the state predicted from the gamma posterior vector computed using the acoustic model at time t; it takes values from the set $\mathbb{Q}$.
$\mathbf{z}_t = [P(Z_t = q_1\|\mathcal{X}) \ldots P(Z_t = q_K\|\mathcal{X})]^\top$	K-dimensional gamma posterior probability vector given by the acoustic model; superscript $\top$ denotes transpose.
$\mathcal{Z} =< Z_1, \ldots, Z_t, \ldots, Z_T >$	Sequence of states predicted framewise from the gamma posteriors.

probability distribution is assumed to be independent of previous states and previous observations, and the transition probabilities follow a first-order Markovian structure. More details on HMM and its application for ASR is explained in Section 2.2.1.

In addition to the terminology defined in Chapter 2, we introduce another HMM related quantity here - the “gamma posterior probability” - which is relevant to the work presented in this chapter. The gamma posterior probability is defined as:

$$\gamma_t(q_k) = P(Q_t = q_k|\mathcal{X}) \tag{7.1}$$

hence representing the probability of hidden state Q_t at time t taking the value q_k given the whole observation sequence $\mathcal{X}$. The gamma posterior probability is computed using forward-backward algorithm, details of which can be found in [Rabiner (1989)]. We term the vector $\gamma_t = [\gamma_t(q_1), \ldots, \gamma_t(q_k), \ldots, \gamma_t(q_K)]^\top$ as the gamma posterior vector at time t for the concerned utterance. If $\gamma_t(q_k)$ is summed over all time steps of all utterances, the quantity computed can be treated as the posterior probability of the HMM hidden state being q_k at time step t conditioned on the whole utterance.

As discussed before, the hidden states usually correspond to senones in modern large vocabulary ASR. While the total number of logical states in a speech modeling HMM could be very large, the number of senones obtained after state tying is typically in the order of few thousands. Since senones are extensively used as physical HMM states in modeling large vocabulary ASR systems, we assume here onwards that the possible values for HMM hidden states come from the set of senones.

7.2.1 Prior Research

Building upon work initiated in the early 90's [Bourlard and Morgan (1994)] and exploiting the availability of larger amounts of training data and processing power, DNNs are now recognized to outperform GMMs in HMM-based ASR [Hinton et al. (2012)]. Several studies have been conducted to better understand the reasons behind the superior performance of DNN acoustic models. We review the previous research findings to enlist some of these properties desired in an acoustic model for improving HMM-based ASR.

7.2.1.1 Towards Better State Conditional Probabilities

A detailed discussion in [Bilmes (2006, 2004)] argues that accurate sequence decoding using HMM requires the acoustic model to be structurally discriminative of the underlying classes. Furthermore, the acoustic model should be designed to preserve maximum mutual information between the input features and the underlying HMM states. Investigations in [Hinton et al. (2012); Bengio (2009)] confirmed that deep neural networks indeed fulfill the above requirements. One of the key factors contributing to the success of DNNs is their invariant representation learning power for class discrimination. Unlike the generative GMM models, DNNs derive discriminative representations of the data by applying non-linear transforms through multiple hidden layers. They alleviate the need for an explicit probability distribution function to model the data because the data-driven discriminative approach leads to more accurate modeling of the underlying class (HMM states) distribution. In [Nagamine et al. (2015)], it was found that the individual neurons in DNN hidden layers learn to be selectively active in different ways towards distinct phone patterns. At the same time, information irrelevant to phonetic discrimination such as gender are discarded by the deeper (closer to the output) layers. It was also confirmed analytically in [Huang et al. (2014)] that DNNs are significantly better at phone classification compared to GMMs and, although robustness against unseen noise and data/channel mismatch is a challenge, they still outperform GMMs in these conditions.

In summary, the success of DNN based acoustic modeling in ASR is partly owed to the accurate estimation of the state-specific probabilities and better discrimination of the boundaries resulting in superior HMM state-level classification results.

7.2.1.2 Acoustic Modeling Using Markovian Structure

Prior research has also investigated the effects of HMM structural hypotheses on ASR failures. In particular, the conditional independence assumption of HMM is often acknowledged as the number one limiting factor resulting in poor ASR performance and lack of robustness [Ravuri and Wegmann (2016)].

Natural speech exhibits strong temporal correlations and contextual dependencies. This correlation is partly present in the acoustic features. HMM conditional independence assumption requires that the acoustic features associated with a specific sub-word (senone) state are independent of the past and future states and acoustic features. To test this hypothesis, earlier works [McAllaster et al. (1998); Gillick et al. (2011)] replaced real speech data with synthetic data which strictly follow HMM assumptions. When the synthetic data strictly follow conditional independence assumptions, the ASR performance was found to be nearly perfect even using GMM-HMM architecture. Along this line, [Ravuri and Wegmann (2016); Gillick et al. (2012)] have studied how DNNs cope with violation of HMM assumptions. It was shown that using many hidden layers in DNNs yields acoustic models less sensitive to the contextual dependencies. Each hidden layer successively makes the system more robust towards the contextual dependencies existing in the real speech data, hence resulting in better ASR. From the prior research summarized above, we conclude that for improving HMM-based ASR, the acoustic model should be robust against the violation of Markovian conditional independence assumption.

In contrast, a different approach to tackle the limitation of the conditional independence assumption is to rely on alternative architectures that can capture longer temporal dependencies. For instance, the segmental models proposed in [Ostendorf et al. (1996)] model a speech segment of long duration as a unit instead of the traditional frame-wise modeling procedure. In segmental models, the HMM conditional independence assumptions are not enforced within a segment, thus reducing the data-model mismatch. More recently, recurrent and convolutional neural network architectures, e.g. long short-term memory (LSTM) RNNs [Hochreiter and Schmidhuber (1997); Sak et al. (2014); Lu et al. (2016)] and time-delay neural networks (TDNN) [Waibel et al. (1990); Peddinti et al. (2015)] based acoustic models respectively, have also shown consistent improvements over the state-of-the-art DNN based ASR performance. Acoustic models based on these architectures access a longer temporal context of acoustic features to make the framewise prediction of HMM state posterior probabilities. Another active area of research is towards developing end-to-end ASR systems based on connectionist temporal classification [Graves et al. (2006)] and attention-based mechanisms [Bahdanau et al. (2016); Chan et al. (2016)]. End-to-end systems typically do away with the traditional HMM backend for ASR by directly focussing on neural network based sequence-to-sequence modeling.

7.2.2 Contributions of this Chapter

The contributions of the work presented in this chapter can be summarized as follows:

- We develop an analytical framework that does not rely on empirical evidence such as phone classification errors and ASR accuracies. The proposed method quantifies the desired acoustic model properties without performing ASR, and the deficiencies in the system can be measured in disjoint aspects.
- Previous mutual information estimation method [Bilmes (1998)] based on GMMs uses the expectation-maximization algorithm to learn a joint probability distribution which quantifies the quality of GMM-based acoustic modeling. We propose a novel z-HMM formulation that facilitates expressing the qualitative aspects of acoustic models in information theoretic terms by directly using state posterior probabilities
- We theoretically quantify the contribution of DNNs in addressing the limitations imposed by HMM assumptions. We show that DNNs are not only more accurate in computing state conditional probabilities, but they are also more robust against the contextual dependencies existing in the data which violate the HMM conditional independence requirement. In addition, we provide a quantitative analysis of GMM, RNN, and TDNN acoustic models. We also evaluate DNNs with different training criteria and model architectures.
- We apply the proposed analysis technique to measure the effect of sparse and low-rank modeling in improving DNN based ASR.

The next section introduces a speaking-listening perspective to the process of ASR, which forms the basis of our z-HMM formulation.

7.3 Speaking-Listening z-HMM Perspective

The graphical model for a traditional speech recognition HMM (in Figure 7.1(a)) as discussed in Section 7.2 consists of a sequence of hidden states which emit observable acoustic features at each time step. In terms of random variables, the hidden state Q_t underlies the generation of feature X_t. Acoustic feature X_t can take a value $\mathbf{x}_t \in \mathbb{R}^n$ and state Q_t can take a value $q_t \in \mathbb{Q}$ from the set of senones.

For the sake of the acoustic modeling analysis framework proposed in this paper, we introduce a novel z-HMM formulation. This formulation is described here as follows: a DNN based acoustic model estimates the posterior probabilities of senone classes based on the observed acoustic feature[2] X_t. The state posterior probabilities are then converted to pseudo-likelihoods by dividing them with the state prior probabilities as shown in (2.12). In the case of GMMs, the data likelihoods are computed directly from the Gaussian probability distribution function.

[2] While the acoustic feature at time t is usually appended with a small context c to make predictions using a DNN acoustic model, we assume in our definition of z-HMM that the context appended input feature vector as a whole is the observed feature belonging to the frame at time t.

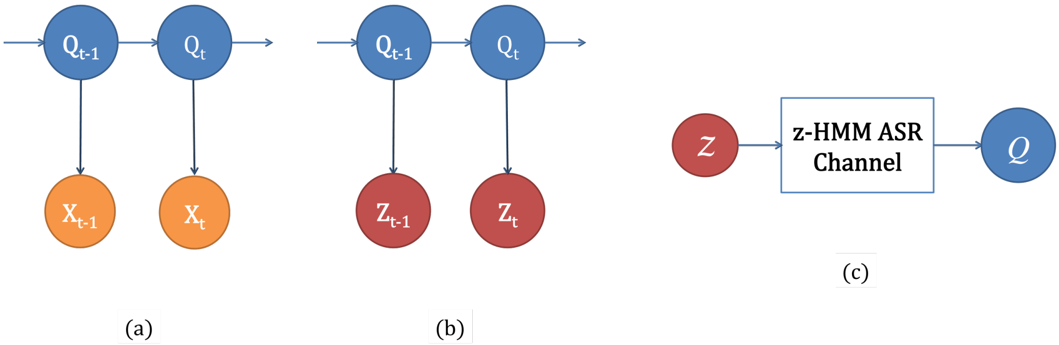

Figure 7.1: (a) Classical HMM graphical model where hidden state Q_t emits acoustic feature observation X_t; (b) z-HMM graphical model: a hidden states $Q_t \in \mathbb{Q}$ emits a random variable $Z_t \in \mathbb{Q}$. From this new perspective, the sequence of states $\mathcal{Q}$ governs the speaking process (or speech generation) in the domain of acoustic features $\mathcal{X}$, whereas the sequence of acoustic model predictions $\mathcal{Z}$ directs the listening process (or speech recognition) which is inferred from $\mathcal{X}$. (c) ASR is analyzed as a communication channel which transmits the sequence of acoustic model predictions $\mathcal{Z}$ for decoding the hidden state sequence $\mathcal{Q}$.

Using the data likelihood probabilities and ignoring any language model constraints, we run the forward-backward algorithm with a controlled beam size to compute the gamma posterior probabilities (as defined in (7.1)) for each state at each time step. Although the fixed beam size provides us only with an approximation of the true gamma posteriors, it ensures that the forward-backward algorithm is tractable to compute. At this stage, gamma posteriors are the probabilistic predictions made by the acoustic model about the value taken by the HMM hidden states Q_t's at different time steps based on the complete observed sequence $\mathcal{X}$. For each time step t, the gamma posterior vector (γ_t) can be seen as a categorical probability distribution over the predicted value of Q_t. In the proposed z-HMM, this categorical prediction about the value taken by the hidden state random variable Q_t is defined as a new random variable Z_t and is considered as an observable feature. The graphical model shown in Figure 7.1(b) depicts this process. This model here is referred to as z-HMM since the emission from the hidden state Q_t is denoted as the discrete random variable Z_t.

The z-HMM formulation as defined above leads to following important observations:

- Since the observed feature Z_t in z-HMM is simply a categorical prediction about Q_t, both the random variables Q_t and Z_t take values from the same domain of possible states, i.e., $\mathbb{Q}$.
- The categorical probability distribution over Z_t is conditioned over the whole acoustic feature sequence $\mathcal{X}$ and is given by the forward-backward algorithm as the gamma posterior vector $\mathbf{z}_t = [P(Z_t = q_1|\mathcal{X})\ldots P(Z_t = q_K|\mathcal{X})]^\top$ at time t. We denote this probability distribution vector as $\mathbf{z}_t$ instead of γ_t to emphasize that we are considering the prediction about hidden state Q_t as a different random variable Z_t in the z-HMM formulation.

- Although random variable Z_t is defined as an observed feature in z-HMM, we do not actually have access to the exact values taken by any particular Z_t. We only have access to the acoustic model's probabilistic prediction about the random variable Z_t taking values from the set $\mathbb{Q}$ conditioned on the observed feature sequence $\mathcal{X}$.
- Since the language model constraints are ignored and the HMM transition probabilities are kept fixed once the acoustic model is trained, the accuracy of gamma posteriors $\mathbf{z}_t$'s as computed by the forward-backward algorithm is directly dependent on the data likelihoods (or state posteriors in case of DNN) generated by the acoustic model.

Under this framework, we can now introduce distinct interpretations for z-HMM's hidden state Q_t and the observed feature Z_t as separate random variables corresponding to

- Q_t: Speaking random variable, and
- Z_t: Listening random variable.

The speaker intends the production of speech under HMM state $Q_t = q_k$ (where q_k could be representing a physical HMM state, e.g., senone) at time t. The acoustic features $\mathcal{X}$ serve as the information bearing medium through which the listener infers the observed feature Z_t in a probabilistic manner in terms of the gamma posterior vector $\mathbf{z}_t$. Thus, the task of speech recognition is for the listener to find the most likely speaker state sequence $\mathcal{Q}$ given that the listener state sequence $\mathcal{Z}$ was observed. Therefore, we consider ASR as a communication channel (Figure 7.1(c)) where the input is the sequence of listening random variable $\mathcal{Z}$, obtained from the acoustic model (e.g., DNN), and the output is the sequence of speaker random variable $\mathcal{Q}$. From this perspective, z-HMM can be interpreted as a joint speaking-listening HMM where the speaking process is represented by the underlying hidden sequence $\mathcal{Q}$ leading to the acoustic features $\mathcal{X}$ and the listening process, inferred from $\mathcal{X}$ by the acoustic model, is represented by the observed feature sequence $\mathcal{Z}$.

Note that the conception of z-HMM is only for the purpose of ensuing information theoretic analysis, and it should not be thought of as a new approach towards ASR. In the next section, we present the information theoretical analysis of the acoustic models based on z-HMM formulation described above.

7.4 Information Theoretic Analysis of Acoustic Models

In the context of z-HMM described above, we consider the following two factors as *critical* in judging the quality of an acoustic model- (1) how accurate are the data likelihoods or state conditional posterior probabilities and (2) how well the acoustic model complies with the HMM assumptions. These factors can be quantified using the following two mutual information terms involving the random variables Q_t and Z_t as

(i) $I(Z_t; Q_t)$: the mutual information between the observed feature Z_t and the underlying hidden state Q_t, and

(ii) $I(Z_t; Q_{t-1}|Q_t)$: the mutual information between the feature Z_t and the former state Q_{t-1} if the current underlying hidden state Q_t is known.

The notion of mutual information is defined in Section 7.4.1. We explain the relation between the above quantities and the desired properties of acoustic models in Section 7.4.2.

7.4.1 Mutual Information

In information theory, the mutual information of two random variables quantifies the information conveyed about one random variable by the other random variable. This concept is defined through the notion of entropy which measures the quantity of information held in a random variable [Cover and Thomas (1991)].

For a discrete random variable A which takes values $a \in \mathcal{A}$, the entropy $H(A)$ is defined as

$$H(A) = -\sum_{a\in\mathcal{A}} P(A = a)\log(P(A = a)) \tag{7.2}$$

Accordingly, the conditional entropy $H(A|B)$ of a random variable A given another random variable B, which takes values $b \in \mathcal{B}$, is defined as

$$H(A|B) = \sum_{b\in\mathcal{B}} P(B = b)H(A|B = b) \tag{7.3}$$

where

$$H(A|B = b) = -\sum_{a\in\mathcal{A}} P(A = a|B = b)\log(P(A = a|B = b)) \tag{7.4}$$

Entropy quantifies the uncertainty of a random variable; thereby, it increases as the uncertainty about the underlying values grows, i.e., $p(A)$ and $p(B)$ tend to a uniform distribution.

Mutual information $I(A; B)$ between two random variables A and B is the measure of mutual dependence between the two variables. It quantifies the reduction in uncertainty of A due to the knowledge of B, and vice versa. It is defined as

$$I(A; B) = H(A) - H(A|B) = H(B) - H(B|A) \tag{7.5}$$

Accordingly, conditional mutual information is defined as

$$I(A; B|C) = H(A|C) - H(A|B, C) \tag{7.6}$$

7.4.2 Desired Acoustic Model Properties

The z-HMM ASR communication channel discussed in Section 7.3 is most efficient when 1) the observed feature Z_t at the input and the underlying hidden state Q_t which is to be inferred have the highest mutual information $I(Z_t; Q_t)$, and 2) the HMM conditional independence assumption is well satisfied so that the mutual information between the feature and the former hidden state is minimized if the current hidden state is known, i.e., $I(Z_t; Q_{t-1}|Q_t)$ approaches zero.

7.4.2.1 Property 1: High Information Transmission Capacity (P$_1$)

To maximize the amount of information transmitted by the acoustic model through the ASR channel, it is desired to have a high mutual information between the feature Z_t and the underlying state Q_t expressed as

$$I(Z_t; Q_t) = H(Z_t) - H(Z_t|Q_t) \quad \forall t \in \{1, \ldots, T\} \tag{7.7}$$

In an ideal scenario, no uncertainty should be left in the acoustic model's prediction Z_t by revealing the underlying hidden state Q_t. Therefore, $H(Z_t|Q_t)$ should be zero because of the one-to-one logical mapping between random variables Q_t and Z_t. But, due to variability in the intermediate acoustic features $\mathcal{X}$ and possible correlations between states, the acoustic model's prediction Z_t is not deterministic and the probabilities $P(Z_t|Q_t)$'s are not binary, leading to a non-zero value of $H(Z_t|Q_t)$. We rely on $I(Z_t; Q_t)$ as the measure of the information transmitted by the acoustic model in the ASR communication channel.

7.4.2.2 Property 2: First-order Markovian HMM Structure (P$_2$)

It is desired that the acoustic model yields robustness to the HMM conditional independence assumption that is often violated by the speech acoustic features. In other words, the feature Z_t emitted by an underlying state Q_t should be independent of the past and future states as well as observations if the current hidden state Q_t is known. This property can be expressed as

$$Z_t \perp\!\!\!\perp \left\{Q_{\neg t}, Z_{\neg t}\right\} | Q_t \tag{7.8}$$

In this work, we limit our analysis of conditional independence only to the preceding hidden state Q_{t-1} although our algorithmic approach is generally extendable to any order of dependency computation. We quantify the following condition:

$$Z_t \perp\!\!\!\perp Q_{t-1} | Q_t \tag{7.9}$$

To measure the amount of mutual dependence between Z_t and Q_{t-1}, we deploy conditional mutual information as

$$I(Z_t;Q_{t-1}|Q_t) = H(Z_t|Q_t) - H(Z_t|Q_{t-1},Q_t) \tag{7.10}$$

In an ideal scenario, $I(Z_t;Q_{t-1}|Q_t) = 0$ indicates that the acoustic model fulfills the first-order Markovian requirement for HMM based decoding.

7.4.3 Computational Procedure Using Gamma Posterior Features

In this section, we develop the procedure to quantify the desired properties P_1 and P_2 by using the gamma posteriors generated using the acoustic model. This procedure requires the ground truth-based forced state alignments to compute the mutual information terms in P_1 and P_2.

To measure P_1 and P_2, the following entropy terms must be computed for calculation of the mutual information measures in (7.7) and (7.10):

$$\{H(Z_t),\ H(Z_t|Q_t),\ H(Z_t|Q_t,Q_{t-1})\} \tag{7.11}$$

The entropy terms, in turn, require computation of the following probabilities:

$$\{P(Z_t),\ P(Z_t|Q_t),\ P(Z_t|Q_t,Q_{t-1})\} \tag{7.12}$$

Using forward-backward algorithm on the likelihoods obtained from the acoustic model, we get the gamma posterior vector $\mathbf{z}_t$ which is the categorical probability distribution over the random variable Z_t conditioned on the acoustic feature sequence $\mathcal{X}$. Along with $\mathbf{z}_t$'s, we also generate the ground truth transcription based forced alignments for all the utterances in the dataset. The forced state alignments (senones alignments in our experiments) are considered as the underlying true hidden state sequence $\mathcal{Q}$ in the context of z-HMM. The acoustic models are trained using a transcribed training data. We perform the analysis on a separate development dataset which also has ground truth transcription available. While gamma posterior vectors $\mathbf{z}_t$'s are probabilistic, the forced alignments are one-hot vectors, i.e., they are simply framewise labels from the set of possible states $\mathbb{Q}$.

The probability $P(Z_t)$ required in (7.12) is approximated as the average gamma posterior probability by marginalization over all frames in the dataset across all the utterances as:

$$\begin{aligned} P(Z_t) &= [P(Z_t = q_1),\dots,P(Z_t = q_K)]^\top \\ &\approx \frac{1}{N}\sum_{t=1}^{N}\mathbf{z}_t \end{aligned} \tag{7.13}$$

where N is the total number of frames in the dataset summed over all utterances in the data. This is an approximation because the marginalization considers a uniform probability distribution for all the observations sequences $\mathcal{X}$ in the dataset. Given a large sample size in the analysis, this ensemble averaging can be assumed to yield a reliable estimate.

To obtain the state conditional gamma posterior probabilities $P(Z_t|Q_t = q_k)$, we consider only those frames for marginalization which are aligned to the state $Q_t = q_k$ in the forced alignments. Thus, we have

$$\begin{aligned} P(Z_t|Q_t = q_k) &= [P(Z_t = q_1|Q_t = q_k), \dots, P(Z_t = q_K|Q_t = q_k)]^\top \\ &\approx \frac{1}{N_{q_k}} \sum_{\substack{t \text{ s.t.} \\ Q_t = q_k}} \mathbf{z}_t \end{aligned} \tag{7.14}$$

where N_{q_k} is the number of frames in the dataset aligned to senone q_k. Similarly, we compute $P(Z_t|Q_t = q_k, Q_{t-1} = q_{k'})$ by averaging the gamma posteriors only over those frames that are aligned to state $Q_t = q_k$ and the preceding frame is aligned to the state $Q_{t-1} = q_{k'}$ in the forced alignment as

$$\begin{aligned} &P(Z_t|Q_t = q_k, Q_{t-1} = q_{k'}) \\ &\qquad = [P(Z_t = q_1|Q_t = q_k, Q_{t-1} = q_{k'}), \dots, P(Z_t = q_K|Q_t = q_k, Q_{t-1} = q_{k'})]^\top \\ &\qquad \approx \frac{1}{N_{q_k, q_{k'}}} \sum_{\substack{t \text{ s.t. } Q_t = q_k \\ Q_{t-1} = q_{k'}}} \mathbf{z}_t \end{aligned} \tag{7.15}$$

where $N_{q_k, q_{k'}}$ is the number of frames aligned to senone q_k such that preceding frame is aligned to senone $q_{k'}$.

The steps to compute the entropies in (7.11) and the calculation of the required mutual information quantities from probabilities computed in (7.13), (7.14) and (7.15) are listed in Algorithm 3. The state prior probabilities $P(Q_t = q_k)$ and state transition probabilities $P(Q_t = q_k, Q_{t-1} = q'_k)$ involved in Algorithm 3 are obtained by the frequency count approach using the ground truth forced state alignment. Quality of the acoustic model can now be measured based on a high value of $I(Z_t; Q_t)$ (property P_1) and a low value of $I(Z_t; Q_{t-1}|Q_t)$ (property P_2).

Algorithm 3 Computing $\mathbf{I(Z_t;Q_t)}$ and $\mathbf{I(Z_t;Q_{t-1}|Q_t)}$ using gamma posterior probabilities

```
Require: : Gamma posterior vectors $\mathbf{z}_t$'s and forced state alignments $\mathcal{Q}$ for the dataset.
1: $P(Q_t = q_k)$ and $P(Q_t = q_k, Q_{t-1} = q'_k)$ are estimated by the frequency count approach using the forced state alignment.
2: $P(Z_t)$ is estimated by averaging all the gamma posteriors $\mathbf{z}_t$ from the data.
3: $H(Z_t)$ is calculated using (7.2).
4: $P(Z_t|Q_t = q_k)$ is estimated through averaging all posteriors $\mathbf{z}_t$ aligned to state $q_k$.
5: $H(Z_t|Q_t = q_k)$ is calculated using (7.4).
6: $P(Z_t|Q_t = q_k, Q_{t-1} = q_{k'})$ is estimated by averaging all posteriors $\mathbf{z}_t$ aligned to state $q_k$ preceded by state $q_{k'}$.
7: $H(Z_t|Q_t = q_k, Q_{t-1} = q_{k'})$ is calculated using (7.4).
8: $H(Z_t|Q_t)$ is calculated using (7.3) on the state specific entropies estimated in Step 5 and probabilities $P(Q_t = q_k)$ from Step 1.
9: $H(Z_t|Q_{t-1}, Q_t)$ is calculated using (7.3) on the state transition specific entropy terms estimated in Step 7 and probabilities $P(Q_t = q_k, Q_{t-1} = q'_k)$ from Step 1.
10: $I(Z_t;Q_t)$ is calculated using (7.7) and the entropies estimated in Steps 3 and 8.
11: $I(Z_t;Q_{t-1}|Q_t)$ is calculated using (7.10) and the entropies estimated in Steps 8 and 9.
```

7.5 Numerical Evaluation and Analysis

7.5.1 Experimental Setup

For our analysis, we work with AMI IHM dataset. The details of the setup are similar as in Section 5.4.2. Note that a baseline GMM-HMM system is used to obtain the ground truth based forced senone alignments over the train set and dev set. These alignments serve as the hidden state sequences $\mathcal{Q}$ for our analysis.

Our baseline DNN acoustic model has 4 hidden layers with 1200 nodes each, and it is trained using hard targets from the forced senone alignments. Details of LSTM based and TDNN acoustic models are provided in Section 6.1.2.1. Details of the sparse and low-rank enhancements based student DNN models are discussed in Section 5.4.3. For our analysis, the gamma posterior probabilities are obtained by employing the forward-backward algorithm on emission probabilities generated by the concerned acoustic models. To approximate the forward-backward algorithm, we generate very deep lattices by choosing a high value for `-lattice-beam=100` and setting `-determinize-lattice=false` in Kaldi lattice generation scripts. We ignore the contribution of language model by setting `-acoustic-scale=100` during decoding which essentially diminishes the language model scores. Finally, we also set `-lm-scale=0` and use a large value `-beam=100` for beam search in Kaldi decoding scripts.

The information theoretic analysis of GMM versus DNN acoustic models is shown in Table 7.2. The last row shows all the quantities by treating forced senone alignments as binary posterior features. Hence, the entropy $H(Z_t)$ here refers to the entropy of the prior probabilities of senone classes and this row essentially depicts the most ideal values we could hope to achieve from an acoustic model.

7.5.2 Comparing DNN v/s GMM Acoustic Modeling

We compare the first two rows of Table 7.2 here. Our primary observation is that the DNN acoustic model exhibits higher mutual information $I(Z_t;Q_t)$ (property P_1) as compared to GMM. This indicates that the capacity of the ASR communication channel (Figure 7.1(c)) using DNNs is higher than GMMs and the DNN based posteriors are more accurate in discrimination of the underlying senone classes. This observation confirms the well known better modeling capability of DNN as compared to GMM.

The low values of state conditional entropies ($H(Z_t|Q_t)$ and $H(Z_t|Q_t,Q_{t-1})$) in the case of DNN acoustic model suggests that DNN are able to learn a sharper distribution over senones as compared to GMMs. When the DNN based sharp state conditional posterior vectors $P(Z_t|Q_t = q_k)$ are averaged across all the states, i.e., $\forall q_k \in \mathbb{Q}$, the resulting posterior $P(Z_t)$ has comparatively more evenly spread out probability distribution than in the case of GMM. This leads to the DNN having a higher entropy $H(Z_t)$ than the GMM.

Next, we observe that the HMM conditional independence criterion (property P_2) is better satisfied by the DNN acoustic model than by the GMM acoustic model as the mutual information $I(Z_t;Q_{t-1}|Q_t)$ is lower in the former case. In terms of classical HMM, the acoustic features $\mathcal{X}$ violate the HMM conditional independence assumption to the same extent for both DNN and GMM acoustic models. But in terms of z-HMM, the DNN acoustic model transforms the acoustic features $\mathcal{X}$ to generate the observed feature sequence $\mathcal{Z}$ which is in better compliance with property P_2 than $\mathcal{X}$. Hence, we conclude that DNNs are more robust than GMMs against the violation of HMM conditional independence assumption.

When these models are used to perform ASR on test data, DNN performs significantly better than GMM as expected. Since the information theoretic criteria are computed before actual full-fledged decoding for ASR, this study essentially disentangles the contribution of the acoustic modeling from the language modeling. The additional information conveyed by the language model can be quantified nevertheless by re-estimation of the gamma posteriors after a full-fledged ASR decoding.

7.5.3 Effect of Increasing Depth in DNN Acoustic Models

An interesting evaluation is to measure the contribution of increasing the number of DNN hidden layers. The results are listed in Tables 7.2. We compare DNN architectures in a manner similar to the study in [Ravuri and Wegmann (2016)] where the number of hidden layers is increased with or without keeping the total number of network parameters equal to the baseline DNN. DNN with 0 hidden layers is merely a logistic regression model with input layer connected directly to the output layer.

Table 7.2: Information theoretic analysis of different acoustic models to evaluate properties P_1 and P_2: Higher mutual information $I(Z_t;Q_t)$ (7.7) indicates higher accuracy of the acoustic model and lower conditional mutual information $I(Z_t;Q_{t-1}|Q_t)$ (7.10) shows compliance with the HMM conditional independence assumption. Information theoretic analysis is done on AMI-dev set and AMI-eval set is used for evaluating ASR performance; last column shows word error rate (WER, in %). xHL denotes number of hidden layers and EP denotes equal parameters as the baseline DNN with 4 hidden layers.

Acoustic Model	$H(Z_t)$	$H(Z_t\|Q_t)$	$H(Z_t\|Q_t,Q_{t-1})$	$I(Z_t;Q_t)$	$I(Z_t;Q_{t-1}\|Q_t)$	AMI *eval* (WER%)
GMM	9.698	3.416	3.018	6.281	0.399	42.9
DNN(-4HL)	9.890	2.915	2.527	6.975	0.388	32.4
DNN-3HL	9.922	2.964	2.568	6.957	0.396	32.8
DNN-2HL	9.959	3.018	2.618	6.942	0.400	34.5
DNN-1HL	10.026	3.120	2.710	6.906	0.410	36.9
LogReg(-0HL)	10.302	3.719	3.252	6.583	0.466	52.0
DNN-3HL-EP	9.934	2.953	2.560	6.981	0.393	33.0
DNN-2HL-EP	9.943	3.013	2.613	6.930	0.399	34.0
DNN-1HL-EP	10.007	3.098	2.690	6.909	0.408	36.1
Forced Aligned	10.057	0	0	10.057	0	-

As expected, we observe that the mutual information $I(Z_t;Q_t)$ increases with the depth of the DNN leading to higher acoustic model accuracy. This trend is observed in both the cases-when the number of model parameters is equal and when they are not. At the same time, the mutual information $I(Z_t;Q_{t-1}|Q_t)$ gradually decreases with increasing depth which implies that the property P_2 is better satisfied. DNN with more hidden layer has increased robustness against the violation of HMM conditional independence assumption by acoustic features.

The zero hidden layer logistic regression network (*LogReg*) has the highest correlation between the features and the past state, and subsequently performs the poorest in the ASR task. Compared to this model, the GMM acoustic model performs significantly better in ASR. While GMM has a low mutual information $I(Z_t;Q_t)$ than *LogReg* model under property P_1, the former shows a much better compliance in case of property P_2. This observation highlights the importance of mutual information term $I(Z_t;Q_{t-1}|Q_t)$ being low for better ASR performance under HMM-based frameworks. The comparison and subsequent equivalence of generative GMM and discriminative log-linear modeling approaches has been studied extensively in [Heigold et al. (2011)].

7.5.4 Relations to Cross Entropy Training of DNN

As done for the random variable Z_t, we can define another prediction random variable $\widetilde{Z}_t$ based on the frame level DNN posteriors. $\widetilde{Z}_t$ is conditioned only on the acoustic feature X_t at current time step whereas Z_t is conditioned on the complete sequence $\mathcal{X}$. The distribution of random variable $\widetilde{Z}_t$ is given by the DNN output as $P(\widetilde{Z}_t = q_k|X_t)$, $\forall q_k \in \mathbb{Q}$. The frame level cross entropy (CE) training of DNNs can be viewed as minimizing the Kullback-Leibler divergence between frame level DNN predictions $P(\widetilde{Z}_t = q_k|X_t)$ and ground truth senone alignment $P(Q_t = q_k|\mathcal{X})$ (which is a binary one-hot vector in case of Viterbi training).

The mutual information term $I(Z_t;Q_t)$ in our analysis is, in fact, indirectly connected to the CE loss based training of DNN. This is because minimizing the CE loss function between $P(\widetilde{Z}_t = q_k|X_t)$ and $P(Q_t = q_k|\mathcal{X})$ leads to more accurate emission probability estimations for the forward-backward algorithm which in turn gives the probability distribution for Z_t. For the qualitative analysis of the acoustic models, we consider the mutual information $I(Z_t;Q_t)$ $(= H(Z_t) - H(Z_t|Q_t))$ as a more interpretable quantity than the frame level CE loss. In the context of z-HMM, it directly quantifies the decrease in uncertainty about the observation Z_t when the correct emitting state Q_t is revealed. Hence, a more accurate model for the state emission distribution leads to a bigger decrease in the uncertainty about the random variable Z_t.

7.5.5 Sparse and Low-rank Acoustic Model Enhancement

In Chapter 5, we modify the forward pass outputs of the baseline DNN using (1) principal component analysis (PCA) based low-rank reconstruction and (2) overcomplete dictionary based sparse reconstruction. We illustrated this process again in Figure 7.2(a). This process essentially computes the projection of DNN posterior features on the correct senone subspaces. In terms of information theory, the acoustic modeling component (Figure 7.2(b)) now consists of DNN acoustic model followed by an additional block of principal component transform or dictionary based sparse coding.

Table 7.4 summarizes the results for this section. Analysis of the first three rows shows that when the DNN posteriors are enhanced using sparse or low-rank reconstruction, the subsequent gamma posteriors exhibit much lower entropies $H(Z_t)$ and $H(Z_t|Q_t)$ than the baseline DNN. Also, the mutual information $I(Z_t;Q_t)$, between hidden senone state Q_t and the observation Z_t, increases significantly using low-rank and sparse enhancements. These observations show that low-rank and sparse reconstructions project the DNN posteriors to their class-specific manifolds and enforce them to capture more information related to the underlying senone subspaces. This increase in mutual information $I(Z_t;Q_t)$ explicitly quantifies the additional information that is contributed by the enhancement process. Note that PCA and dictionary-based reconstruction can exploit the global information about the senone subspaces and thus, they ought to bring additional information to the DNN local estimates. This information is available in terms of global patterns within each senone's posterior features, but

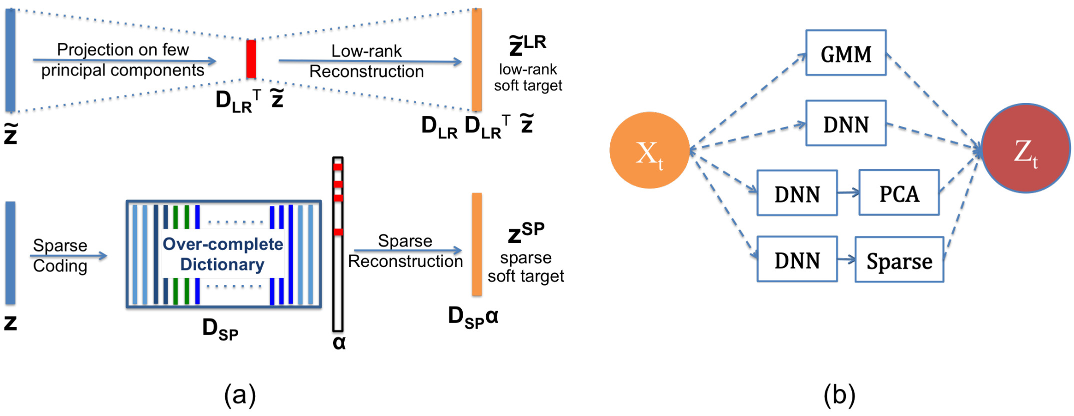

Figure 7.2: (a) Supervised enhancement of DNN based posterior features is illustrated using low-rank and sparsity based reconstruction as proposed in Chapter 5. (b) PCA and sparse coding are shown as a distinct and additional level of processing that captures global structures for better acoustic modeling.

Table 7.3: Information theoretic analysis of different acoustic models to evaluate properties P_1 and P_2 using AMI dev set. Rows with sparse and PCA projection results refer to supervised enhancement of AMI dev set using dictionaries or principal component matrices. Rows with results on sparse and PCA student refer to DNN acoustic models trained with sparse and low-rank soft targets obtained from supervised enhancement of AMI train set DNN posteriors. ASR evaluation on AMI *eval set* is shown only for the enhanced student models.

Acoustic Model	$H(Z_t)$	$H(Z_t\|Q_t)$	$H(Z_t\|Q_t, Q_{t-1})$	$I(Z_t;Q_t)$	$I(Z_t;Q_{t-1}\|Q_t)$	AMI *eval* (WER%)
DNN	9.890	2.915	2.527	6.975	0.388	32.4
Sparse Proj.	9.499	1.341	1.142	8.158	0.199	-
PCA Proj.	7.721	0.059	0.051	7.663	0.008	-
Sparse Student	9.852	2.865	2.479	6.987	0.386	31.6
PCA Student	9.902	2.903	2.512	6.999	0.391	31.2

it is not accessible to the baseline DNN during a local forward pass of the acoustic features. It is only through the supervised enhancement using principal components or an overcomplete dictionary that we are able to augment this global information in the local framewise posterior features.

Another interesting observation is that the conditional independence assumption is also better satisfied (low values of $I(Z_t;Q_{t-1}|Q_t)$) in case of low-rank and sparsity-based projections. We explain it using Figure 7.3 as follows. The frames aligned with senone q_k in the forced alignment can appear in the neighborhood of different senones in different parts of the speech utterances. These frames exhibit different contextual information in DNN posteriors due to different neighboring senones. This contextual information which is always present in the real data violates the conditional independence assumption of HMM and leads to compromise in ASR performance. When posterior features are reconstructed using PCA or an overcomplete dictionary, all the frames of senone q_k are forced to lie on a common subspace which defines q_k. By controlling the parameters of PCA and sparse reconstruction, we ensure that only the most important dynamics of the senone subspace are preserved during enhancement of DNN posteriors. Context dependent information, which is local to an individual frame and does not appear in the global patterns of the subspace, is reduced after reconstruction. Thus, the enhanced posterior features fulfill the HMM conditional independence assumption better than the posteriors before reconstruction.

A caveat here is that the sparse and low-rank enhancements are done in a supervised fashion on the dev set by using the knowledge of the forced alignments. Such ground truth text-based alignments cannot be assumed to be available for the test data, and therefore, we can not perform supervised enhancements in a similar way. To alleviate this issue, we used the reconstructed posteriors from the training data as *enhanced* soft targets to train improved student acoustic models in Chapter 5. The enhanced student models can now be used to evaluate the ASR performance of our approach on test data. These student DNNs trained using enhanced soft targets are expected to learn- 1) the acoustic modeling function learned by the baseline DNN as well as 2) the reconstruction transformation performed by PCA or sparse coding. Last two rows in Table 7.4 provide the results of our information theoretic analysis on enhanced student models. While both the sparse and PCA based student models show an increase in the mutual information $I(Z_t; Q_t)$, the decrease in mutual information term $I(Z_t; Q_{t-1}|Q_t)$ is only observed in case of the sparse student model. Nevertheless, we get ASR performance gains from both the models as compared to the baseline DNN suggesting that the better satisfaction of property P_1 outweighs property P_2 here. The student models learn to estimate the posterior probabilities in class-specific subspaces, thus leading to better acoustic modeling.

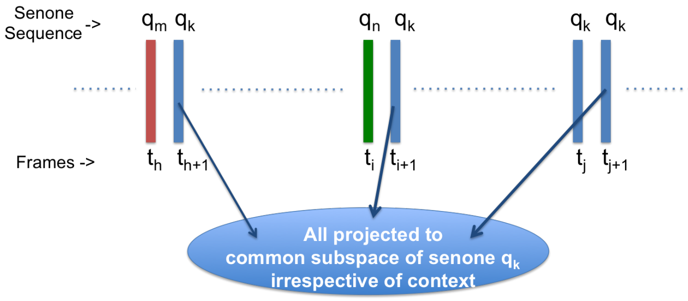

Figure 7.3: Sparse and low-rank reconstruction enforces different posterior features of a senone class to lie on a common low-dimensional subspace. Reconstructed posterior features have reduced local contextual correlations and satisfy HMM's conditional independence criteria better.

Table 7.4: Information theoretic analysis of LSTM and TDNN acoustic models to evaluate properties P_1 and P_2 using AMI dev set. ASR evaluation is done on AMI eval set.

Acoustic Model	$H(Z_t)$	$H(Z_t\|Q_t)$	$H(Z_t\|Q_t, Q_{t-1})$	$I(Z_t; Q_t)$	$I(Z_t; Q_{t-1}\|Q_t)$	AMI *eval* (WER%)
DNN (CE)	9.890	2.915	2.527	6.975	0.388	32.4
DNN (CE+sMBR)	9.744	2.745	2.418	6.999	0.327	29.6
LSTM	9.948	2.958	2.562	6.991	0.396	30.9
Bi-LSTM	9.912	2.590	2.231	7.322	0.359	29.4
TDNN	9.837	2.655	2.271	7.182	0.384	29.7

7.5.6 Analysing Various State-of-the-art Acoustic Models

The first two rows in Table 7.4 compare the results of our analysis on the baseline CE loss based DNN acoustic model with a sMBR loss based sequence discriminatively trained DNN acoustic model. The sMBR objective for sequence discrimination directly optimizes the DNN parameters to minimize the Bayes risk in the state-level alignment [Veselý et al. (2013)]. In our experiments, the sequence trained model significantly outperforms the baseline model in ASR performance on AMI *eval* set. This observation is strongly supported by our analysis on the *dev* set which shows favorable results in terms of better satisfaction of both property P_1 and P_2 for the sequence trained model. Specifically, we notice a big decrease in the mutual information term $I(Z_t; Q_{t-1}|Q_t)$ as compared to the increase in the term $I(Z_t; Q_t)$. This observation hints that the superiority of sMBR loss based models originates more from the increased robustness against the violation of HMM assumptions as compared to the increased accuracy in the modeling of state emission probabilities.

Table 7.4 also compares the results of our analysis on LSTM and TDNN acoustic models. These models are essentially different from conventional feed-forward DNNs as they use recurrent or convolutional connections to exploit longer contexts of acoustic features as compared to the simplistic splicing based context-appending done in the case of DNNs. The architecture of these models is same as in the setup described in Section 6.1.2.1.

In our analysis, the Bi-LSTM model gives the best ASR performance followed by TDNN and then the LSTM acoustic model. The strength of Bi-LSTM model is corroborated by the increased value of mutual information $I(Z_t; Q_t)$ and decreased value of $I(Z_t; Q_{t-1}|Q_t)$. This confirms that Bi-LSTM model not only produces highly accurate state posterior probabilities (property P_1) but also satisfies the z-HMM conditional independence assumption better than the baseline DNN (property P_2). We notice similar results in the case of TDNN where the improved ASR performance is in accordance with the expected changes in the mutual information terms. In the case of LSTM acoustic model, while there is an improvement in ASR performance, we notice a small unexpected increase in information term $I(Z_t; Q_{t-1}|Q_t)$ indicating an increased violation of HMM assumptions. However, we conjecture that this is compensated by the increase in information $I(Z_t; Q_t)$ leading to stronger acoustic modeling than the baseline DNN. This observation can also be partially attributed to the fact that property P_2 only considers the previous hidden state in analyzing the data-model mismatch. A comprehensive verification of satisfaction of HMM conditional independence assumption would require considering all the future and past hidden states as well as observations which we regard out of the scope of the current work.

We also note here that it is non-trivial to infer how the recurrent and convolutional models bring robustness against the data-model mismatch and violation of HMM assumptions. We hypothesize that the longer temporal contexts allow the acoustic models to decorrelate the correlated acoustic features better and generate gamma posteriors which better comply with the HMM conditional independence assumption.

7.6 Conclusions

In this work, an information theoretic approach is presented to compare the quality of different acoustic models in HMM-based ASR. The proposed analysis is based on a novel speaking-listening z-HMM perspective to HMM-based ASR that exploits the state occupancy probability distributions as categorical observations emitted by the HMM hidden states. The information transferred through the ASR communication channel is quantified in our analysis using random variables associated with the proposed z-HMM formulation. A higher amount of information transferred through this channel indicates better modeling capability of the acoustic model and results in improved ASR performance. The conditional independence assumption of HMM is also evaluated in terms of the conditional mutual information between the current observation and the previous state if the current hidden state is given. A lower value of the conditional mutual information shows better compliance with the HMM structure.

Our experimental analysis yields quantitative measurements for different aspects of the superiority of DNN based acoustic modeling over GMMs. The contribution of the incremental increase in the depth of DNN is also measured to study its role in improving the quality of the acoustic modeling. The proposed analysis can be evaluated before using the acoustic model for ASR on test data. As a use case, our analysis framework is applied on enhanced student models trained using PCA and dictionary-based soft targets. These models are shown to improve upon DNN acoustic modeling by bringing in additional global information about the senone subspaces to the DNN local estimations. In another application, sMBR loss based DNN acoustic models as well as LSTM and TDNN based acoustic models are evaluated. State-of-the-art ASR performance by these models is explained well by the increased amount of information transferred through the ASR communication channel as compared to baseline DNN acoustic model. Furthermore, these models are also shown to have increased robustness against violation of HMM conditional independence assumptions.

8 Conclusions and Directions for Future Work

In this chapter, Section 8.1 summarizes the conclusions of this thesis and Section 8.2 discusses the directions of future research.

8.1 Conclusions

In this thesis, we addressed the problem of automatic speech recognition using sparse and low-rank modeling techniques. We hypothesize that speech data lives in class-specific low-dimensional subspaces whereas random unstructured noise is scattered in high dimensions. In this regard, we identified DNN based posterior features as excellent representations for modeling the class-specific low-dimensional subspaces of speech.

We exploited posterior features as exemplars for ASR under the exemplar-based sparse representation approach. We demonstrated how posterior features provide an elegant compressive sensing based interpretation to the HMM-based ASR formulation where word probabilities can be directly obtained from phone probabilities using an overcomplete dictionary. We introduced the use of dictionary learning algorithms and collaborative hierarchical sparse recovery to develop a sparse modeling based solution for ASR. We successfully evaluated this approach on isolated word and connected digit recognition tasks. As a limitation of our approach, we identified that it is currently non-trivial to extend dictionary learning and sparse recovery to LVCSR tasks due to lack of data availability and computational resources.

Next, we investigated explicit modeling of low-dimensional structures in speech using dictionary learning and principal component analysis. We showed that albeit their power in representation learning, DNN based acoustic modeling still has room for improvement in exploiting the union of low-dimensional subspaces structure underlying speech data. Using dictionaries and principal components, we transform DNN posteriors into projected posteriors which act as high-quality soft-targets for training improved acoustic models. We motivated this approach by performing a rank analysis of senone specific subspaces which revealed that speech data indeed lies in low-dimensional subspaces which are corrupted by undesired high-dimensional noise. Using our approach, we showed improvements in ASR performance on a difficult LVCSR task in both close-talk and far-field conditions. For far-field speech, we used our approach to perform speech enhancement and improve acoustic modeling for ASR simultaneously.

Lastly, we developed an analytical framework based on concepts of information theory to understand why sparse and low-rank modeling of speech data leads to improvements in ASR. Using a novel z-HMM formulation, this analysis quantified the exact gains brought in by our approach in 1) increasing the accuracy of acoustic modeling, and 2) satisfying the assumptions imposed by HMM. We also demonstrated the application of the proposed analysis framework on a variety of conventional and recent state-of-the-art acoustic models.

To conclude, we conducted a comprehensive set of experiments to provide an empirical justification for the hypotheses that this thesis is based on, and rigorous theoretical arguments and analysis further consolidated the experimental validation.

8.2 Directions for Future Research

The research presented in this thesis can be further extended along the following lines:

- The sparse modeling approach developed in Chapter 4 can be formulated into a Bayesian dictionary-based HMM where state-specific overcomplete dictionaries govern the emission distribution of hidden states. EM algorithm based training of dictionary atoms could be devised for this formulation which integrates the strength of exemplar-based modeling with those of HMMs.
- In Chapter 5, we proposed dictionary and principal component analysis based enhancement of DNN posteriors. In future, this specific operation could be integrated into the neural network architecture such that the network promotes the global class-specific patterns and suppresses local misinformation. Class-specific sparse autoencoders and bottleneck NN architectures are promising tools in this direction. In general, we advocate the inclusion of sparsity and low-rankness constraints in designing robust and generalizable models for speech recognition.
- We proposed far-field speech enhancement using dictionary learning and sparse modeling of acoustic features in Chapter 6. The promising results achieved in the context of ASR suggest that this approach can be used in a supervised fashion to improve the quality of far-field speech.
- Projection of speech features on low-dimensional class-specific subspaces resulted in noise robust ASR in Chapter 5 and 6. Further applications of the proposed approach can be sought on more challenging speech recognition tasks like tackling accented speech, multi-channel reverberant speech, multi-lingual databases, and ASR under speech disorder condition, etc. An example of an application would be as follows. Accented speech features could be projected on sparse modeling dictionaries that are learned from native speakers' speech. Transformation of the accented phonetic space to native

phonetic space may lead to improvements in accented speech recognition task.

- In future, the information theoretic analysis framework proposed in Chapter 7 could be extended to evaluate the quality of the acoustic modeling component along with the language modeling component, or to just compare various language modeling techniques against each other.

Bibliography

M. Aharon, M. Elad, and A. Bruckstein. KSVD: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Transactions on Signal Processing*, 54(11): 4311–4322, 2006.

J. B. Allen. How do humans process and recognize speech? *IEEE Transactions on Speech and Audio Processing*, 2(4):567–577, 1994.

G. Aradilla. *Acoustic models for posterior features in speech recognition*. PhD thesis, École Polytechnique Fédéral de Lausanne (EPFL), 2008.

G. Aradilla and H. Bourlard. Posterior features applied to speech recognition tasks with user-defined vocabulary. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 3809–3812. IEEE, 2009.

G. Aradilla, J. Vepa, and H. Bourlard. An acoustic model based on Kullback-Leibler divergence for posterior features. In *2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07*, volume 4, pages 657–660, April 2007.

G. Aradilla, H. Bourlard, and M. Magimai-Doss. Using KL-based acoustic models in a large vocabulary recognition task. In *Interspeech*, 2008.

S. Bahaadini, A. Asaei, D. Imseng, and H. Bourlard. Posterior-based sparse representation for automatic speech recognition. In *Interspeech*, 2014.

D. Bahdanau, J. Chorowski, D. Serdyuk, P. Brakel, and Y. Bengio. End-to-end attention-based large vocabulary speech recognition. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4945–4949, March 2016.

M. Banko and E. Brill. Scaling to very very large corpora for natural language disambiguation. In *Proceedings of the 39th annual meeting on association for computational linguistics*, pages 26–33. Association for Computational Linguistics, 2001.

Y. Bengio. Learning deep architectures for AI. *Foundations and trends® in Machine Learning*, 2(1):1–127, 2009.

J. A. Bilmes. Maximum mutual information based reduction strategies for cross-correlation based joint distributional modeling. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, volume 1, pages 469–472, 1998.

J. A. Bilmes. What HMMs can't do. In *ATR Workshop, Invited paper and lecture*, 2004.

J. A. Bilmes. What HMMs can do. *IEICE Transactions on Information and Systems*, 89(3): 869–891, 2006.

H. A. Bourlard and N. Morgan. *Connectionist speech recognition: a hybrid approach*. Springer Science & Business Media, 1994.

E. J. Candes and T. Tao. Decoding by linear programming. *IEEE transactions on information theory*, 51(12):4203–4215, 2005.

E. J. Candès and M. B. Wakin. An introduction to compressive sampling. *IEEE signal processing magazine*, 25(2):21–30, 2008.

E. J. Candès, X. Li, Y. Ma, and J. Wright. Robust principal component analysis? *Journal of the ACM (JACM)*, 58(3):11, 2011.

W. Chan, N. R. Ke, and I. Lane. Transferring knowledge from a RNN to a DNN. In *Interspeech*, 2015.

W. Chan, N. Jaitly, Q. Le, and O. Vinyals. Listen, attend and spell: A neural network for large vocabulary conversational speech recognition. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4960–4964, March 2016.

Z. Chen, S. Watanabe, H. Erdogan, and J. R. Hershey. Speech enhancement and recognition using multi-task learning of long short-term memory recurrent neural networks. In *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

R. A. Cole, M. Noel, T. Lander, and T. Durham. New telephone speech corpora at CSLU. In *Fourth European Conference on Speech Communication and Technology*, 1995.

T. M. Cover and J. A. Thomas. Entropy, relative entropy and mutual information. 1991.

G. Davis, S. Mallat, and M. Avellaneda. Adaptive greedy approximations. *Constructive approximation*, 13(1):57–98, 1997.

M. De Wachter, M. Matton, K. Demuynck, P. Wambacq, R. Cools, and D. Van Compernolle. Template-based continuous speech recognition. *IEEE Transactions on Audio, Speech, and Language Processing*, 15(4):1377–1390, 2007.

L. Deng. Switching dynamic system models for speech articulation and acoustics. In *Mathematical Foundations of Speech and Language Processing*, pages 115–133. Springer New York, 2004.

P. A. Devijver and J. Kittler. *Pattern recognition: A statistical approach*, volume 761. Prentice-Hall London, 1982.

P. Dighe. Eigenposteriors and sparse dictionary codes: https://github.com/idiap/eigenposterior. 2017. URL https://github.com/idiap/eigenposterior.

P. Dighe, A. Asaei, and H. Bourlard. Sparse modeling of neural network posterior probabilities for exemplar-based speech recognition. *Speech Communication*, 2015.

P. Dighe, G. Luyet, A. Asaei, and H. Bourlard. Exploiting low-dimensional structures to enhance dnn based acoustic modeling in speech recognition. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 5690–5694. IEEE, Mar. 2016.

P. Dighe, A. Asaei, and H. Bourlard. Low-rank and sparse soft targets to learn better dnn acoustic models. In *Proceedings of 2017 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2017)*, 2017a.

P. Dighe, A. Asaei, and H. Bourlard. Exploiting eigenposteriors for semi-supervised training of dnn acoustic models with sequence discrimination. In *Proceedings of Interspeech*, 2017b.

P. Dighe, A. Asaei, and H. Bourlard. Far-field asr using low-rank and sparse soft targets from parallel data. *Submitted to IEEE Workshop on Spoken Language Technology*, 2018a.

P. Dighe, A. Asaei, and H. Bourlard. On quantifying the quality of acoustic models in hybrid DNN-HMM ASR. *Submitted to Speech Communication*, 2018b.

P. Dighe, A. Asaei, and H. Bourlard. Low-rank and sparse subspace modeling of speech for dnn based acoustic modeling. *Submitted to Speech Communication*, 2018c.

D. L. Donoho. For most large underdetermined systems of linear equations the minimal l1-norm solution is also the sparsest solution. *Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences*, 59(6): 797–829, 2006.

D. L. Donoho and M. Elad. Optimally sparse representation in general (nonorthogonal) dictionaries via l1 minimization. *Proceedings of the National Academy of Sciences*, 100(5): 2197–2202, 2003.

J. Du, Q. Wang, T. Gao, Y. Xu, L.-R. Dai, and C.-H. Lee. Robust speech recognition with speech enhanced deep neural networks. In *Fifteenth Annual Conference of the International Speech Communication Association*, 2014.

B. Efron, T. Hastie, I. Johnstone, R. Tibshirani, et al. Least angle regression. *The Annals of statistics*, 32(2):407–499, 2004.

M. Elad. *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*. Springer Publishing Company, Incorporated, 1st edition, 2010. ISBN 144197010X, 9781441970107.

E. Elhamifar and R. Vidal. Sparse subspace clustering: Algorithm, theory, and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(11):2765–2781, 2013.

M. H. Firooz and S. Roy. Network tomography via compressed sensing. In *Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE*, pages 1–5. IEEE, 2010.

J. Fiscus, J. Ajot, N. Radde, and C. Laprun. Multiple dimension levenshtein edit distance calculations for evaluating automatic speech recognition systems during simultaneous speech. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*, 2006.

S. Foucart and H. Rauhut. *A mathematical introduction to compressive sensing*, volume 1. Springer, 2013.

J. Friedman, T. Hastie, and R. Tibshirani. A note on the group lasso and a sparse group lasso. *arXiv preprint arXiv:1001.0736*, 2010.

T. Gao, J. Du, L.-R. Dai, and C.-H. Lee. Joint training of front-end and back-end deep neural networks for robust speech recognition. In *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*, pages 4375–4379. IEEE, 2015.

J. Gemmeke, L. Ten Bosch, L. Boves, and B. Cranen. Using sparse representations for exemplar based continuous digit recognition. In *Proc. EUSIPCO*, pages 24–28. Citeseer, 2009.

J. F. Gemmeke, T. Virtanen, and A. Hurmalainen. Exemplar-based sparse representations for noise robust automatic speech recognition. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(7):2067–2080, 2011.

D. Gillick, L. Gillick, and S. Wegmann. Don't multiply lightly: Quantifying problems with the acoustic model assumptions in speech recognition. In *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pages 71–76. IEEE, 2011.

D. Gillick, S. Wegmann, and L. Gillick. Discriminative training for speech recognition is compensating for statistical dependence in the HMM framework. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4745–4748. IEEE, 2012.

R. Giri, M. L. Seltzer, J. Droppo, and D. Yu. Improving speech recognition in reverberation using a room-aware deep neural network and multi-task learning. In *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*, pages 5014–5018. IEEE, 2015.

V. Goel and W. J. Byrne. Minimum bayes-risk automatic speech recognition. *Computer Speech & Language*, 14(2):115–135, 2000.

A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd international conference on Machine learning*, pages 369–376. ACM, 2006.

G. Heigold, H. Ney, P. Lehnen, T. Gass, and R. Schluter. Equivalence of generative and log-linear models. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(5):1138–1148, 2011.

H. Hermansky, D. P. Ellis, and S. Sharma. Tandem connectionist feature extraction for conventional hmm systems. In *icassp*, pages 1635–1638. IEEE, 2000.

G. Hickok. Computational neuroanatomy of speech production. *Nature Reviews Neuroscience*, 13(2):135, 2012.

I. Himawan, P. Motlicek, D. Imseng, B. Potard, N. Kim, and J. Lee. Learning feature mapping using deep neural network bottleneck features for distant large vocabulary speech recognition. In *IEEE ICASSP*, pages 4540–4544, 2015.

G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, et al. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Processing Magazine*, 29 (6):82–97, 2012.

G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

H.-G. Hirsch and D. Pearce. The aurora experimental framework for the performance evaluation of speech recognition systems under noisy conditions. In *ASR2000-Automatic Speech Recognition: Challenges for the new Millenium ISCA Tutorial and Research Workshop (ITRW)*, 2000.

S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8): 1735–1780, 1997.

X. Huang, A. Acero, and H.-W. Hon. *Spoken Language Processing: A Guide to Theory, Algorithm, and System Development*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 1st edition, 2001. ISBN 0130226165.

Y. Huang, D. Yu, C. Liu, and Y. Gong. A Comparative Analytic Study on the Gaussian Mixture and Context Dependent Deep Neural Network Hidden Markov Models. In *Interspeech 2014*, 2014.

B. Hutchinson, M. Ostendorf, and M. Fazel. A sparse plus low-rank exponential language model for limited resource scenarios. *IEEE Transactions on Audio, Speech, and Language Processing*, 23(3):494–504, 2015.

M.-Y. Hwang, X. Huang, and F. A. Alleva. Predicting unseen triphones with senones. *IEEE Transactions on speech and audio processing*, 4(6):412–419, 1996.

M. G. Jafari and M. D. Plumbley. Speech denoising based on a greedy adaptive dictionary algorithm. In *2009 17th European Signal Processing Conference*, pages 1423–1426, Aug 2009.

A. Janin, D. Baron, J. Edwards, D. Ellis, D. Gelbart, N. Morgan, B. Peskin, T. Pfau, E. Shriberg, A. Stolcke, et al. The ICSI meeting corpus. In *IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003.*

A. Jansen and P. Niyogi. Intrinsic fourier analysis on the manifold of speech sounds. In *IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP)*, 2006.

F. Jelinek. Continuous speech recognition by statistical methods. *Proceedings of the IEEE*, 64 (4):532–556, 1976.

F. Jelinek. *Statistical methods for speech recognition.* MIT press, 1997.

R. Z. J.-T. H. Y. G. Jinyu Li. Learning Small-Size DNN with Output-Distribution-Based Criteria. In *Interspeech*, 2014.

J. Kang, C. Lu, M. Cai, W.-Q. Zhang, and J. Liu. Neuron sparseness versus connection sparseness in deep neural network for large vocabulary speech recognition. In *ICASSP*, pages 4954–4958, April 2015.

P. Kenny, M. Lennig, and P. Mermelstein. A linear predictive HMM for vector valued observation with application to speech recognition. *IEEE Transactions on Acoustic Speech and Signal Processing*, 38(1), 1990.

J. Kim, M. El-Khamy, and J. Lee. Residual LSTM: Design of a deep recurrent architecture for distant speech recognition. *arXiv preprint arXiv:1701.03360*, 2017.

J. Kim, M. El-Khamy, and J. Lee. Bridgenets: Student-teacher transfer learning based on recursive neural networks and its application to distant speech recognition. *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 5755–5759, 2018.

S. King, J. Frankel, K. Livescu, E. McDermott, K. Richmond, and M. Wester. Speech production knowledge in automatic speech recognition. *The Journal of the Acoustical Society of America*, 121(2):723–742, 2007.

Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436, 2015.

L. J. Lee, P. Fieguth, and L. Deng. A functional articulatory dynamic model for speech production. In *ICASSP*, volume 2, pages 797–800. IEEE, 2001.

M. S. Lewicki and T. J. Sejnowski. Learning overcomplete representations. *Neural computation*, 12(2):337–365, 2000.

G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma. Robust recovery of subspace structures by low-rank representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (99):1–1, 2013.

L. Lu, L. Kong, C. Dyer, N. A. Smith, and S. Renals. Segmental recurrent neural networks for end-to-end speech recognition. In *Interspeech 2016*, pages 385–389, 2016. doi: 10.21437/Interspeech.2016-40. URL http://dx.doi.org/10.21437/Interspeech.2016-40.

B. Mailhé, R. Gribonval, F. Bimbot, M. Lemay, P. Vandergheynst, and J.-M. Vesin. Dictionary learning for the sparse modelling of atrial fibrillation in ecg signals. In *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*, pages 465–468. IEEE, 2009.

J. Mairal, F. Bach, J. Ponce, and G. Sapiro. Online learning for matrix factorization and sparse coding. *Journal of Machine Learning Research (JMLR)*, 11:19–60, 2010.

J. Mairal, F. Bach, and J. Ponce. Sparse modeling for image and vision processing. *arXiv preprint arXiv:1411.3230*, 2014.

D. McAllaster, L. Gillick, F. Scattone, and M. Newman. Studies with fabricated switchboard data: Exploring sources of model-data mismatch. In *Proceedings of DARPA Broadcast News Transcription and Understanding Workshop*, 1998.

I. McCowan, J. Carletta, W. Kraaij, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaiskos, et al. The AMI meeting corpus. In *Proceedings of the 5th International Conference on Methods and Techniques in Behavioral Research*, volume 88, 2005.

L. Meier, S. Van De Geer, and P. Bühlmann. The group lasso for logistic regression. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70(1):53–71, 2008.

M. Mimura, S. Sakai, and T. Kawahara. Speech dereverberation using long short-term memory. In *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

T. Nagamine, M. L. Seltzer, and N. Mesgarani. Exploring how deep neural networks form phonemic categories. In *INTERSPEECH*, pages 1912–1916, 2015.

B. A. Olshausen and D. J. Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381(6583):607, 1996.

B. A. Olshausen and D. J. Field. Sparse coding with an overcomplete basis set: A strategy employed by v1? *Vision research*, 37(23):3311–3325, 1997.

M. Ostendorf, V. V. Digalakis, and O. A. Kimball. From HMM's to segment models: a unified view of stochastic modeling for speech recognition. *IEEE Transactions on Speech and Audio Processing*, 4(5):360–378, Sep 1996. ISSN 1063-6676. doi: 10.1109/89.536930.

S. J. Pan and Q. Yang. A survey on transfer learning. *IEEE Transactions on Knowledge & Data Engineering*, (10):1345–1359, 2009.

V. Panayotov, G. Chen, D. Povey, and S. Khudanpur. Librispeech: An asr corpus based on public domain audio books. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5206–5210, April 2015. doi: 10.1109/ICASSP.2015.7178964.

V. Peddinti, D. Povey, and S. Khudanpur. A time delay neural network architecture for efficient modeling of long temporal contexts. In *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

V. Peddinti, V. Manohar, Y. Wang, D. Povey, and S. Khudanpur. Far-field asr without parallel data. *Interspeech 2016*, pages 1996–2000, 2016.

J. P. Pinto, M. Magimai.-Doss, and H. Bourlard. Mlp based hierarchical system for task adaptation in asr. In *Proceedings of the IEEE workshop on Automatic Speech Recognition and Understanding (ASRU)*, 2009.

J. Pitrelli, C. Fong, S. H. Wong, J. R. Spitz, and H. C. Leung. Phonebook: A phonetically-rich isolated-word telephone-speech database. In *ICASSP*, volume 1, pages 101–104. IEEE, 1995.

M. D. Plumbley, T. Blumensath, L. Daudet, R. Gribonval, and M. E. Davies. Sparse representations in audio and music: from coding to source separation. *Proceedings of the IEEE*, 98(6): 995–1005, 2010.

D. Povey. *Discriminative training for large vocabulary speech recognition*. PhD thesis.

D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlíček, Y. Qian, P. Schwarz, et al. The Kaldi speech recognition toolkit. 2011.

D. Povey, V. Peddinti, D. Galvez, P. Ghahremani, V. Manohar, X. Na, Y. Wang, and S. Khudanpur. Purely sequence-trained neural networks for asr based on lattice-free mmi. In *Interspeech 2016*, pages 2751–2755, 2016. doi: 10.21437/Interspeech.2016-595. URL http://dx.doi.org/10.21437/Interspeech.2016-595.

R. Price, K.-i. Iso, and K. Shinoda. Wise teachers train better DNN acoustic models. *EURASIP Journal on Audio, Speech, and Music Processing*, (1):1–19, 2016.

Y. Qian, T. Tan, and D. Yu. An investigation into using parallel data for far-field speech recognition. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5725–5729, March 2016. doi: 10.1109/ICASSP.2016.7472774.

L. R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

D. Ram, A. Asaei, P. Dighe, and H. Bourlard. Sparse modeling of posterior exemplars for keyword detection. In *Proceedings of Interspeech*, pages 3690–3694, Sept. 2015.

D. Ram, A. Asaei, and H. Bourlard. Subspace detection of DNN posterior probabilities via sparse representation for query by example spoken term detection. In *Interspeech*, 2016.

D. Ram, A. Asaei, and H. Bourlard. Sparse subspace modeling for query by example spoken term detection. *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, 26(6):1126–1139, June 2018a. ISSN 2329-9290.

D. Ram, L. Miculicich, and H. Bourlard. CNN based query by example spoken term detection. In *Proceedings of Interspeech*, 2018b.

S. P. Rath, D. Povey, and K. Veselỳ. Improved feature processing for deep neural networks. In *Interspeech*, 2013.

S. Ravuri and S. Wegmann. How neural network depth compensates for HMM conditional independence assumptions in DNN-HMM acoustic models. *Interspeech 2016*, pages 2736–2740, 2016.

S. Renals and P. Swietojanski. Distant speech recognition experiments using the ami corpus. In *New Era for Robust Speech Recognition*, pages 355–368. Springer, 2017.

I. Rish and G. Grabarnik. Sparse signal recovery with exponential-family noise. In *Compressed Sensing & Sparse Filtering*, pages 77–93. Springer, 2014.

D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning representations by back-propagating errors. *Nature*, 323(6088):533, 1986.

T. N. Sainath, B. Ramabhadran, D. Nahamoo, D. Kanevsky, and A. Sethy. Sparse representation features for speech recognition. In *Interspeech*, pages 2254–2257, 2010.

T. N. Sainath, B. Ramabhadran, M. Picheny, D. Nahamoo, and D. Kanevsky. Exemplar-based sparse representation features: From TIMIT to LVCSR. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(8):2598–2613, 2011.

T. N. Sainath, B. Ramabhadran, D. Nahamoo, D. Kanevsky, D. Van Compernolle, K. Demuynck, J. F. Gemmeke, J. R. Bellegarda, and S. Sundaram. Exemplar-based processing for speech recognition: An overview. *IEEE Signal Processing Magazine*, 29(6):98–113, 2012.

T. N. Sainath, B. Kingsbury, V. Sindhwani, E. Arisoy, and B. Ramabhadran. Low-rank matrix factorization for deep neural network training with high-dimensional output targets. In *ICASSP*, pages 6655–6659. IEEE, 2013.

H. Sak, A. W. Senior, and F. Beaufays. Long short-term memory recurrent neural network architectures for large scale acoustic modeling. 2014.

P. Schmid-Saugeon and A. Zakhor. Dictionary design for matching pursuit and application to motion-compensated video coding. *IEEE Transactions on Circuits and Systems for Video Technology*, 14(6):880–886, June 2004. ISSN 1051-8215. doi: 10.1109/TCSVT.2004.828329.

C. E. Shannon. Communication in the presence of noise. *Proceedings of the IRE*, 37(1):10–21, 1949.

P. Sharma, V. Abrol, and A. K. Sao. Deep-sparse-representation-based features for speech recognition. *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, 25 (11):2162–2175, 2017.

J. Shlens. A tutorial on principal component analysis. *arXiv preprint arXiv:1404.1100*, 2014.

K. C. Sim, Y. Qian, G. Mantena, L. Samarakoon, S. Kundu, and T. Tan. *Adaptation of Deep Neural Network Acoustic Models for Robust Automatic Speech Recognition*, pages 219–243. Springer International Publishing, 2017. ISBN 978-3-319-64680-0.

S. Soldo, M. Magimai-Doss, J. Pinto, and H. Bourlard. Posterior features for template-based ASR. In *ICASSP*, pages 4864–4867. IEEE, 2011.

P. Sprechmann, I. Ramirez, G. Sapiro, and Y. C. Eldar. C-HiLasso: A collaborative hierarchical sparse modeling framework. *Signal Processing, IEEE Transactions on*, 59(9):4183–4198, 2011.

N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

K. N. Stevens. Acoustic phonetics. 1998.

P. Swietojanski, A. Ghoshal, and S. Renals. Hybrid acoustic models for distant and multichannel large vocabulary speech recognition. In *Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop on*, pages 285–290. IEEE, 2013.

P. Swietojanski, A. Ghoshal, and S. Renals. Convolutional neural networks for distant speech recognition. *IEEE Signal Processing Letters*, 21(9):1120–1124, 2014.

R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

V. S. Tomar and R. C. Rose. Manifold regularized deep neural networks. 2014.

I. Tosic and P. Frossard. Dictionary learning. *IEEE Signal Processing Magazine*, 28(2):27–38, 2011.

I. Tošić, I. Jovanović, P. Frossard, M. Vetterli, and N. Durić. Ultrasound tomography with learned dictionaries. In *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*, pages 5502–5505. IEEE, 2010.

J. A. Tropp and S. J. Wright. Computational methods for sparse solution of linear inverse problems. *Proceedings of the IEEE*, 98(6):948–958, 2010.

B. E. Usevitch. A tutorial on modern lossy wavelet image compression: foundations of jpeg 2000. *IEEE signal processing magazine*, 18(5):22–35, 2001.

K. Veselỳ, A. Ghoshal, L. Burget, and D. Povey. Sequence-discriminative training of deep neural networks. 2013.

A. Waibel, T. Hanazawa, G. Hinton, K. Shikano, and K. J. Lang. Phoneme recognition using time-delay neural networks. In *Readings in speech recognition*, pages 393–404. Elsevier, 1990.

F. Weninger, H. Erdogan, S. Watanabe, E. Vincent, J. Le Roux, J. R. Hershey, and B. Schuller. Speech enhancement with LSTM recurrent neural networks and its application to noise-robust asr. In *International Conference on Latent Variable Analysis and Signal Separation*, pages 91–99. Springer, 2015.

J. H. Wong and M. J. Gales. Sequence student-teacher training of deep neural networks. In *Interspeech 2016*, pages 2761–2765, 2016. doi: 10.21437/Interspeech.2016-911. URL http://dx.doi.org/10.21437/Interspeech.2016-911.

J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma. Robust face recognition via sparse representation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 31(2): 210–227, 2009.

Y. Xu, J. Du, L.-R. Dai, and C.-H. Lee. An experimental study on speech enhancement based on deep neural networks. *IEEE Signal processing letters*, 21(1):65–68, 2014.

J. Xue, J. Li, and Y. Gong. Restructuring of deep neural network acoustic models with singular value decomposition. In *INTERSPEECH*, 2013.

S. J. Young, J. J. Odell, and P. C. Woodland. Tree-based state tying for high accuracy acoustic modelling. In *Proceedings of the workshop on Human Language Technology*. Association for Computational Linguistics, 1994.

D. Yu, F. Seide, G. Li, and L. Deng. Exploiting sparseness in deep neural networks for large vocabulary speech recognition. In *ICASSP*, pages 4409–4412, 2012.

M. Yuan and Y. Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):49–67, 2006.

Y. Zhang, G. Chen, D. Yu, K. Yaco, S. Khudanpur, and J. Glass. Highway long short-term memory RNNs for distant speech recognition. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 5755–5759. IEEE, 2016.

Pranay Dighe

✉ ppd.1989@gmail.com • 🌐 http://www.pranaydighe.com
in https://www.linkedin.com/in/pranaydighe

Education

École Polytechnique Fédérale de Lausanne(EPFL) **Switzerland**
Doctoral Student in Electrical Engineering *2014 – 2018(expected)*

Indian Institute of Technology Kanpur **India**
Masters in Technology - Computer Science, CGPA: 9.0/10.0 *2008 – 2013*

Indian Institute of Technology Kanpur **India**
Bachelors in Technology - Computer Science, CGPA: 8.5/10.0 *2008 – 2013*

Work Experience

Siri at Apple Inc. **Cupertino, USA**
Speech Research Intern *Mar'17 – Jun'17*

- Acoustic modeling for robust automatic speech recognition

École Polytechnique Fédérale de Lausanne(EPFL)/Idiap **Switzerland**
Research Assistant in Speech & Audio Processing Group *Aug'14 – Present*
Supervisors: Prof Herve Bourlard, Dr. Afsaneh Asaei

- Improving ASR using low-rank and sparse structures in deep neural network (DNN) based posterior probabilities, in context of multi-party meeting scenarios in AMI corpus
- Modeling acoustic space of DNN outputs as a union of low-dimensional subspaces resulting in application of dictionary learning for ASR
- **Github:** `https://github.com/idiap/eigenposterior`

Idiap Research Institute **Switzerland**
Intern in Speech & Audio Processing Group *Jul'13 – Jun'14*
Supervisors: Prof Herve Bourlard, Dr. Marc Ferras Font

- Modeled overlapping speech as vector taylor series approximation of corruption of a primary speaker's speech by a background speaker's speech
- Detection and labelling of overlapping speech in context of speaker diarization systems for AMI corpus

Carnegie Mellon University **Pittsburgh, USA**
Summer Intern, Language Technologies Institute *May'11 – Jul'11*
Supervisor: Prof Bhiksha Raj (LTI, CMU)

- **Multimedia Content Analysis:** Developed GMM-HMM based novel *Acoustic Unit Descriptor(AUD)* features for event detection and context recognition in audio recordings. Evaluated approach on TRECVID Multimedia Event Detection 2011 corpus
- **Language Identification (LID) using Spectro-Temporal Patch Features:** Library of randomly selected patches of spectro-temporal patterns from spectrograms of spoken examples used as novel features for LID. Evaluation done on VoxForge and CallFriend corpus

Masters Thesis

Title: *"Automated Analysis of Indian Classical Music"*
Description: **Supervisor:** Prof Harish Karnick, IIT Kanpur
Created a framework for robust automated analysis of Indian classical music. Implemented novel scale-independent raga (Indian classical melodies) identification using chromagram patterns and swara

(Indian equivalent of solfeges) based features with state-of-the-art results.

Computer skills

Programming:: C/C++, MATLAB/Octave, Java, Python, Bash

Tools:: Visual Basic, LaTeX, git **OS:**: Linux, Windows, Macintosh

Awards and Academic Achievements

- SAMSUNG Innovation Award 2012 for developing a smartphone application "CLASAT- Computationally Light Audio based Semantic Analysis Tool" ***Media link: The Hindu***
- Best Research Project Award at IPTSE-CMU Winter School 2010 for work on emotion recognition from speech using AdaBoost algorithm
- Academic Excellence Award at IIT Kanpur for 2008-09 (top 5% students)
- Rank 310 (99.92 %ile among 400,000 candidates) in IIT-Joint Entrance Examination'08
- Rank 258 (99.96 %ile among 800,000 candidates) in All India Engineering Entrance Examination'08

Activities and Interests

- **Student Representative**, Doctoral School of Electrical Engineering, EPFL
- **Coordinator**, Institute Counselling Service, IIT Kanpur
- Fine arts, reading, cinema, skiing, cricket

Publications

- P Dighe, A Asaei and H Bourlard *"Exploiting Eigenposteriors for Semi-supervised Training of DNN Acoustic Models with Sequence Discrimination"*, in **Interspeech 2017**, Stockholm, Sweden.
- P Dighe, A Asaei and H Bourlard *"Low-rank and Sparse Soft Targets to Learn Better DNN Acoustic Models"*, in **ICASSP 2017**, New Orleans, USA.
- P Dighe, G Luyet, A Asaei and H Bourlard *"Exploiting Low-dimensional Structures to Enhance DNN Based Acoustic Modeling in Speech Recognition"*, , in **ICASSP 2016**, Shanghai, China.
- P Dighe, A Asaei and H Bourlard *"Sparse Modeling of Neural Network Posterior Probabilities for Exemplar-based Speech Recognition"*, in **Speech Communication: Special Issue** on Advances in Sparse Modeling and Low-rank Modeling for Speech Processing, 2015.
- D Ram, A Asaei, P Dighe and H Bourlard *"Sparse Modeling of Posterior Exemplars for Keyword Detection"*, in **Interspeech, 2015**.
- P Dighe, M Ferras, H Bourlard *"Detecting and labeling speakers on overlapping speech using vector taylor series"* in **Interspeech 2014**.
- P Dighe, H Karnick, B Raj *"Swara Histogram Based Structural Analysis and Identification of Indian Classical Ragas"* in ISMIR 2013, Curitiba, Brazil.
- A Kumar, P Dighe, R Singh, S Chaudhuri, B Raj *"Audio event detection from acoustic unit occurrence patterns"* in **ICASSP 2012**, Kyoto, Japan.
- P Dighe, A Asaei, and H Bourlard *"Far-field asr using low-rank and sparse soft targets from parallel data"*, in IEEE Workshop on Spoken Language Technology, 2018 (under review).
- P Dighe, A Asaei, and H Bourlard *"On quantifying the quality of acoustic models in hybrid DNN-HMM ASR"*, in Speech Communication, 2018 (under review).
- P Dighe, A Asaei, and H Bourlard *"Low-rank and sparse subspace modeling of speech for DNN based acoustic modeling"*,Speech Communication, 2018 (under review).

Ce document a été imprimé au Centre d’impression EPFL,
imprimerie climatiquement neutre, certifiée myClimate.