Salient Features for Multilingual Speech Recognition in Indian Scenario

Thesis submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

in

Electronics and Communications Engineering

by

Hari Krishna Vydana
201433663
hari.vydana@research.iiit.ac.in

International Institute of Information Technology
Hyderabad - 500 032, INDIA
January 2019
Acknowledgments

This thesis would not have been possible without the support of many individuals. I would like to extend my gratitude to all of them.
Abstract

**Keywords:** Automatic Speech Recognition, Indian languages, Indian scenarios, Multilingual, Joint acoustic model, Common pone-set, code-mixing, Residual networks, Speaker Normalization

Automatic Speech Recognition (ASR) systems have witnessed a lot of progress in the past decade. In high resourced scenarios like English, ASR systems have shown performance compared to human parity level on specific tasks. Speech recognition systems for Indian languages are less studied compared to other high resourced languages like English. India is a developing country with large emerging markets for speech recognition technology. Developing Indian language ASR systems requires certain challenges to addressed which are innate to Indian languages. India is a multilingual society. India has 23 official languages. To penetrate deep into the Indian markets, ASR systems which can be operated in multiple languages are to be developed. Collecting data from a multilingual environment is much tedious task than to acquire data from a monolingual environment. Often Indian language ASR systems have to be developed for low-resourced scenarios. Apart from multilingual nature, bilingualism is very prevalent in the Indian population which leads frequent code-switching and word borrowing between any two languages. Operating parallel ASR systems with code-switching capabilities in Indian scenarios is a huge challenge. This motivated us to work towards multilingual ASR systems which can handle code-mixing and word borrowing efficiently.

In this thesis, we address various issues related to development of ASR systems for Indian scenarios. An integrated ASR system is developed using common phone-set which can efficiently handle multilingual code-mixed speech. Acoustic modeling approaches such as HMM-GMM, HMM-SGMM and RNN-CTC have been studied to find the most suitable acoustic model. Various acoustic modeling units such as context independent phone, context dependent phone and syllables are explored to find the most suitable unit for developing joint acoustic models. Residual connections have been explored to improve the performance of the joint acoustic models. Studies directed towards supplementing the conventional features along with articulatory features have been explored for developing multilingual ASR systems. Fricative landmarks are detected and the detected landmarks are used as the features for improving the performance of multilingual ASR system. Distinctive features from speech are modeled using a statistical approach and their relevance for improving the performance of a multilingual ASR is explored. In a low
resourced scenario when the data is pooled from multiple sources, the meta-level information about the speaker is not accessible. A speaker normalization to handle those scenarios is explored.

Some major conclusions from the work are:

- Using Common phone-set for training joint acoustic models offers an attractive solution developing ASR systems to handle multilingual and code-mixed scenarios.

- Acoustic models that can model context independent phones have performed better compared to the context dependent tri-phone units. RNN-CTC based joint-acoustic model has performed better that HMM-SGMM model.

- Using residual networks to develop the joint acoustic models have stabilized the training. They have improved the performance of the acoustic model when the model is sufficiently deep.

- Fricative landmarks when fused with the features have improved the performances of the ASR systems when the data size is small, with relatively large sized datasets the performance have not improved significantly.

- Using distinctive features predicted from DNNs along with the input features have not significantly improved the performance of multilingual ASR system.

- When the meta-information about the speaker ID is not available, Using the speaker codes derived from a speaker ID network can be used to improve the performance of a multilingual ASR system.
## Contents

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Introduction</td>
</tr>
<tr>
<td>1.1</td>
<td>Introduction</td>
</tr>
<tr>
<td>1.2</td>
<td>Scope of the Thesis</td>
</tr>
<tr>
<td>1.2.1</td>
<td>An Exploration Towards Joint Acoustic Modeling for Indian Languages</td>
</tr>
<tr>
<td>1.2.2</td>
<td>Exploring Residual Neural Networks for Developing Multilingual Speech Recognition Systems</td>
</tr>
<tr>
<td>1.2.3</td>
<td>Using Fricative Landmarks in Speech Recognition Systems</td>
</tr>
<tr>
<td>1.2.4</td>
<td>Using Distinctive Features for Improving Multilingual ASR</td>
</tr>
<tr>
<td>1.2.5</td>
<td>Speaker Normalization for Low Resourced Scenarios</td>
</tr>
<tr>
<td>1.3</td>
<td>Organization of the Thesis</td>
</tr>
<tr>
<td>2</td>
<td>Multilingual Speech Recognition Systems</td>
</tr>
<tr>
<td>2.1</td>
<td>Overview of Automatic Speech Recognition Systems</td>
</tr>
<tr>
<td>2.1.1</td>
<td>Front-End Parametrization:</td>
</tr>
<tr>
<td>2.1.2</td>
<td>Acoustic Modeling</td>
</tr>
<tr>
<td>2.1.2.1</td>
<td>Generic Feature-based Approaches for Acoustic Modeling</td>
</tr>
<tr>
<td>2.1.2.2</td>
<td>Phonetic Feature-based Approaches for Acoustic Modeling</td>
</tr>
<tr>
<td>2.1.3</td>
<td>Language Modeling</td>
</tr>
<tr>
<td>2.1.4</td>
<td>Decoding</td>
</tr>
<tr>
<td>2.1.5</td>
<td>Evaluating the Performance of an ASR System</td>
</tr>
<tr>
<td>2.2</td>
<td>Multilingual Speech Recognition Systems and the Issues Involved</td>
</tr>
<tr>
<td>2.2.1</td>
<td>Language Identification Systems</td>
</tr>
<tr>
<td>2.2.2</td>
<td>Approaches For Cross-lingual Knowledge Transfer</td>
</tr>
<tr>
<td>2.2.3</td>
<td>End-to-End Approaches for Multilingual ASR Systems</td>
</tr>
<tr>
<td>2.3</td>
<td>Multilingual ASR Systems in Indian Scenarios</td>
</tr>
<tr>
<td>3</td>
<td>An Exploration Towards Joint Acoustic Modeling for Indian Languages</td>
</tr>
<tr>
<td>3.1</td>
<td>Introduction</td>
</tr>
<tr>
<td>3.2</td>
<td>Related Work</td>
</tr>
<tr>
<td>3.3</td>
<td>Multilingual Speech Recognition Systems using Joint Acoustic Models</td>
</tr>
<tr>
<td>3.3.0.1</td>
<td>Database</td>
</tr>
<tr>
<td>3.3.0.2</td>
<td>Common phone-set</td>
</tr>
<tr>
<td>3.3.1</td>
<td>RNN-CTC based Acoustic Modeling</td>
</tr>
<tr>
<td>3.3.1.1</td>
<td>Deep-bidirectional LSTMs</td>
</tr>
<tr>
<td>3.3.1.2</td>
<td>CTC-objective function</td>
</tr>
<tr>
<td>3.3.2</td>
<td>Subspace-Gaussian Mixture Models</td>
</tr>
<tr>
<td>3.4</td>
<td>Experiments, Results &amp; Discussion</td>
</tr>
<tr>
<td>3.4.1</td>
<td>HMM-SGMM vs. RNN-CTC as a Joint Acoustic Model for Multilingual ASR</td>
</tr>
<tr>
<td>3.4.2</td>
<td>Sub-sampling the Acoustic Sequence for training RNN-CTC</td>
</tr>
<tr>
<td>CONTENTS</td>
<td>PAGE</td>
</tr>
<tr>
<td>------------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>3.4.3 Exploring the use of Language ID in a Multilingual ASR</td>
<td>42</td>
</tr>
<tr>
<td>3.4.4 Exploring Various Basic units for Joint Acoustic Modeling</td>
<td>43</td>
</tr>
<tr>
<td>3.4.5 Data Sharing in a Multilingual Joint Acoustic Model</td>
<td>45</td>
</tr>
<tr>
<td>3.4.6 Results &amp; Discussion</td>
<td>46</td>
</tr>
<tr>
<td>3.5 Summary and Conclusions</td>
<td>47</td>
</tr>
<tr>
<td>4 Residual Neural Networks as Joint Acoustic Models for Indian Languages</td>
<td>48</td>
</tr>
<tr>
<td>4.1 Introduction</td>
<td>48</td>
</tr>
<tr>
<td>4.2 Related Work</td>
<td>48</td>
</tr>
<tr>
<td>4.3 Experimental setup of HMM-Residual Neural Networks For Speech Recognition</td>
<td>50</td>
</tr>
<tr>
<td>4.3.1 Database</td>
<td>50</td>
</tr>
<tr>
<td>4.3.2 Feature Extraction</td>
<td>50</td>
</tr>
<tr>
<td>4.3.3 Experimental Details</td>
<td>51</td>
</tr>
<tr>
<td>4.3.4 HMM-Residual Neural Network Architecture</td>
<td>52</td>
</tr>
<tr>
<td>4.3.5 Results &amp; Discussion</td>
<td>53</td>
</tr>
<tr>
<td>4.4 Residual Neural Network as a Joint Acoustic Model for Multilingual Speech Recognition</td>
<td>58</td>
</tr>
<tr>
<td>4.4.1 LSTM Layers With a Linear Projection (LSTM-LP)</td>
<td>59</td>
</tr>
<tr>
<td>4.4.2 Densely Connected LSTM-LP Networks</td>
<td>60</td>
</tr>
<tr>
<td>4.5 Summary and Conclusions</td>
<td>62</td>
</tr>
<tr>
<td>5 Using Fricative Landmarks in Speech Recognition Systems</td>
<td>63</td>
</tr>
<tr>
<td>I Detection of Fricatives using S-Transform</td>
<td>65</td>
</tr>
<tr>
<td>5.1 Introduction</td>
<td>67</td>
</tr>
<tr>
<td>5.2 Related Work</td>
<td>67</td>
</tr>
<tr>
<td>5.3 Database</td>
<td>71</td>
</tr>
<tr>
<td>5.3.1 TIMIT-Database</td>
<td>71</td>
</tr>
<tr>
<td>5.4 S-Transform, Implementation and Implications on Speech</td>
<td>71</td>
</tr>
<tr>
<td>5.4.1 Relation between S-Transform and STFT</td>
<td>73</td>
</tr>
<tr>
<td>5.4.2 Implications of S-Transform on the Speech Signal</td>
<td>75</td>
</tr>
<tr>
<td>5.4.3 Applying S-Transform for Speech Signals</td>
<td>76</td>
</tr>
<tr>
<td>5.5 Proposed S-Transform based Approach for Detecting Fricatives</td>
<td>81</td>
</tr>
<tr>
<td>5.5.1 Arriving at the Threshold</td>
<td>82</td>
</tr>
<tr>
<td>5.6 Evaluation of Proposed S-Transform based Approach for Detecting Fricatives</td>
<td>84</td>
</tr>
<tr>
<td>5.6.1 STFT based Approach for Fricative Detection</td>
<td>88</td>
</tr>
<tr>
<td>5.6.2 Comparing the Performances of STFT and S-Transform based Approaches</td>
<td>88</td>
</tr>
<tr>
<td>5.6.3 Comparing the Performances of Predictability and S-Transform based Approaches</td>
<td>90</td>
</tr>
<tr>
<td>5.7 Combining S-Transform and Predictability based Approach for Detecting</td>
<td>91</td>
</tr>
<tr>
<td>5.8 Summary and Conclusions</td>
<td>96</td>
</tr>
<tr>
<td>II Detection of Fricative Landmarks using Spectral Weighting: A Temporal Approach</td>
<td>99</td>
</tr>
<tr>
<td>5.9 Introduction</td>
<td>101</td>
</tr>
<tr>
<td>5.10 Related Work</td>
<td>101</td>
</tr>
<tr>
<td>5.11 Database</td>
<td>102</td>
</tr>
<tr>
<td>5.12 Spectral Weighting Approach for Detecting Fricatives</td>
<td>103</td>
</tr>
<tr>
<td>5.12.1 Analysis of Various Scaling Functions</td>
<td>105</td>
</tr>
<tr>
<td>5.12.2 Proposed Algorithm</td>
<td>107</td>
</tr>
<tr>
<td>5.12.3 Arriving at a Threshold</td>
<td>108</td>
</tr>
<tr>
<td>5.13 Evaluation of the Spectral Weighting Approach for Detecting Fricatives</td>
<td>112</td>
</tr>
</tbody>
</table>
## CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.13.1 Comparing the Performance of the Proposed Approach with S-transform based Approach and Predictability Measure based Approach</td>
<td>112</td>
</tr>
<tr>
<td>5.13.2 Comparing the Performance S-transform based Approach and Proposed Approach</td>
<td>114</td>
</tr>
<tr>
<td>5.13.3 Comparing the Performance of Proposed Approach and Predictability Measure based Methods</td>
<td>115</td>
</tr>
<tr>
<td>5.14 Combining the Proposed and Predictability based Approaches</td>
<td>116</td>
</tr>
<tr>
<td>5.15 Summary and Conclusions</td>
<td>121</td>
</tr>
</tbody>
</table>

### III Using Fricative Landmarks for Improving the Performance of Speech Recognition Systems 123

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.16 Introduction</td>
<td>125</td>
</tr>
<tr>
<td>5.17 Exploring the Weighted Spectral Evidence for Improving the Performance of Speech Recognition System</td>
<td>125</td>
</tr>
<tr>
<td>5.17.0.1 fMLLR Features</td>
<td>125</td>
</tr>
<tr>
<td>5.17.0.2 HMM-DNN based Speech Recognition System</td>
<td>127</td>
</tr>
<tr>
<td>5.17.0.3 RNN-CTC based Speech Recognition System</td>
<td>127</td>
</tr>
<tr>
<td>5.18 Exploring the Weighted Spectral Evidence for Improving the Performance of Multilingual ASR</td>
<td>129</td>
</tr>
<tr>
<td>5.19 Summary and Conclusions</td>
<td>131</td>
</tr>
</tbody>
</table>

### 6 Using Distinctive Features for Improving Multilingual ASR 132

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.1 Introduction</td>
<td>132</td>
</tr>
<tr>
<td>6.2 Related work</td>
<td>132</td>
</tr>
<tr>
<td>6.3 Proposed Distinctive Feature Network</td>
<td>134</td>
</tr>
<tr>
<td>6.4 Using Distinctive Features in Developing Monolingual ASR</td>
<td>136</td>
</tr>
<tr>
<td>6.5 Using Distinctive Features in Multilingual ASR</td>
<td>137</td>
</tr>
<tr>
<td>6.6 Summary and Conclusions</td>
<td>138</td>
</tr>
</tbody>
</table>

### 7 Speaker Normalization for Low Resourced Scenarios 139

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.1 Introduction</td>
<td>139</td>
</tr>
<tr>
<td>7.1.1 Related Work</td>
<td>140</td>
</tr>
<tr>
<td>7.2 Database &amp; Experimental setup for end-to-end Speech Recognition System</td>
<td>142</td>
</tr>
<tr>
<td>7.2.1 Database</td>
<td>142</td>
</tr>
<tr>
<td>7.2.2 Architecture</td>
<td>142</td>
</tr>
<tr>
<td>7.3 Proposed Approach for Speaker Normalization</td>
<td>142</td>
</tr>
<tr>
<td>7.3.1 Approach 1</td>
<td>142</td>
</tr>
<tr>
<td>7.3.2 Approach 2</td>
<td>144</td>
</tr>
<tr>
<td>7.4 Complementarity of the Proposed Approaches with fMLLR based Speaker Normalization</td>
<td>147</td>
</tr>
<tr>
<td>7.5 Using the Proposed Speaker Normalization for Multilingual ASR System</td>
<td>148</td>
</tr>
<tr>
<td>7.6 Summary and Conclusions</td>
<td>148</td>
</tr>
</tbody>
</table>

### 8 Summary and Conclusions 150

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.1 Conclusions</td>
<td>150</td>
</tr>
<tr>
<td>8.2 Future Work</td>
<td>151</td>
</tr>
</tbody>
</table>

### Appendix A: Multilingual Speech Recognition using Attention based Sequence-to-Sequence Learning 153

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>A.1 Introduction</td>
<td>153</td>
</tr>
<tr>
<td>A.2 Experiments and Results</td>
<td>155</td>
</tr>
<tr>
<td>A.3 Summary and Conclusions</td>
<td>166</td>
</tr>
</tbody>
</table>

### Bibliography 167
List of Figures

Figure | Page
--- | ---
2.1 Working diagram of a speech recognition system | 9
2.2 Blockdiagram describing various processes involved in developing a speech recognition system (The figure has been adapted from [1]). | 10
2.3 Blockdiagram describing the process of computing MFCC features (The figure has been adapted from [1]). | 12
2.4 Blockdiagram describing encoder-decoder networks for speech recognition (The figure has been adapted from [2]). | 21
2.5 Blockdiagram describing a phonetic feature-based speech recognition system (The figure has been adapted from [3]). | 22
2.6 Blockdiagram describing probabilistic framework for landmark based speech recognition system (The figure has been adapted from [4]). | 23
2.7 Blockdiagram describing the algorithm for detecting various abrupt landmarks from speech. (The figure has been adapted from [5]). | 24
2.8 Blockdiagram describing a multilingual ASR with LID as a front-end switch. | 28
2.9 A multilingual ASR which is operated using multiple monolingual ASR systems in parallel and a back-end LID system various processes involved in developing a speech recognition system. | 29
2.10 Blockdiagram describing an integrated multilingual ASR system | 30
2.11 The utf8-text and the corresponding IT3 representation along with the phone sequence produced by the parser is presented. | 33

4.1 Residual block used in this study. | 52
4.2 Comparing the performance of speech recognition systems developed using stacked network and residual network in terms of frame error rate. Figures are generated from the subset of WSJ corpus mentioned in subsection 4.3.1 | 54
4.3 Performance of speech recognition systems developed using stacked network and wide-residual networks in terms of frame error rate. Figures are generated from the subset of WSJ corpus mentioned in subsection 4.3.1 | 57
4.4 Blockdiagram showing LSTM-LP with residual and dense connections. In the above Figure (a) is the blockdiagram of LSTM-LP with residual connections and (b) is the blockdiagram of LSTM-LP with Dense connections. | 60

5.1 Time frequency representation of alveolar voiced fricative [z] using conventional wideband spectrogram and S-Transform. (a) Speech signal (b) S-Transform based spectrogram (c) Conventional spectrogram with 2 ms frame size and 1 sample frame shift. | 77
5.2 Characteristics of alveolar voiced fricative [z]. (a) Speech signal, (b) S-Transform based spectrogram, (c) 1-D temporal curve obtained by computing the spectral energy above 5 kHz, and (d) Short time energy of contour shown in (c). | 78
5.3 Characteristics of alveolar voiced fricative [z] with a vowel context [i]. (a) Speech signal, (b) S-Transform based spectrogram, (c) 1-D temporal curve obtained by computing the spectral energy above 5 kHz, and (d) Short time energy of contour shown in (c). ........................................ 79

5.4 Characteristics of the boundary between the alveolar voiced-unvoiced fricative [z → s] (a) Speech signal, (b) S-Transform based spectrogram, (c) 1-D temporal curve obtained by computing the spectral energy above 5 kHz, and (d) Short time energy of contour shown in (c). .................................................. 80

5.5 (a) Speech signal of phonetically labeled TIMIT utterance “She had your dark suit in greasy wash water”, (b) S-Transform, (c) 1-D temporal curve obtained by computing the spectral energy below 1.5 kHz, (d) 1-D temporal curve obtained by computing the spectral energy above 1.5 kHz, (e) Frication ratio ($F_r$), (f) Friction index, fricative regions predicted by the algorithm is shown in dotted line and the ground truths are indicated using the solid line. ................................................................. 85

5.6 Figure showing the effect of various scaling functions $k, k^2, k^3$ on speech signal and the corresponding spectrogram. Speech signal in the figure is taken from the TIMIT utterance with the transcription “She had your dark suit in greasy wash water”. In the figure, fricative regions are labeled to emphasize the changes due to scaling. (a, b) Speech signal and its corresponding spectrogram with a frame size of 20 ms and frame shift of 10 ms. Spectrally weighted speech signals and the corresponding spectrograms for the weighting functions $k, k^1$ and $k^2$ are shown in (c, d), (e, f), (g, h). ................................................................. 106

5.7 Intermediate evidences computed in the proposed approach. (a) Speech signal (Speech signal in the above figure is taken from the TIMIT utterance with the transcription “She had your dark suit in greasy wash water”), (b) Energy of speech signal ($E_x[n]$), (c) Energy of spectrally weighted speech signal using a scaling function $k^2$ ($E_{x^k}[n]$), (d) H[n] computed by the ratio of $E_x[n], E_{x^k}[n]$, (e) Proposed fricative evidence ($g[n]$) and (f) Proposed fricative evidence ($g[n]$), detective fricative region ($d_{frc}$) shown in dotted line and the groundtruth in solid line. ................................................................. 109

6.1 Blockdiagram describing the proposed distinctive feature based approach. .......................... 136

7.1 Blockdiagram of the proposed speaker normalization approach described as Approach-2 in section. 7.3. ................................................................. 145

A.1 Blockdiagram showing different blocks in sequence-to-sequence with attention. ................. 154

A.2 Plots showing the attention pattern of an utterance across different epochs. The figure is obtained from LA-ATT with location based attention trained using LSTM-LP-7H. The figure is generated during the training phase with a teacher forcing of 0.6. .......................... 158

A.3 Plots showing the attention pattern of an utterance across different epochs. The figure is obtained from MO-ATT with location based attention trained using LSTM-LP-7H. The figure is generated during the training phase with a teacher forcing of 0.6. .......................... 160

A.4 Plots showing the attention pattern of an utterance across different epochs. The figure is obtained from LA-ATT with location based attention trained using LSTM-LP-7H. The figure is generated during the validation phase with a teacher forcing of 0. ................................ 162

A.5 Plots showing the attention pattern of an utterance across different epochs. The figure is obtained from MO-ATT with location based attention trained using LSTM-LP-7H. The figure is generated during the validation phase with a teacher forcing of 0. ................................ 164
## List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Performances of Multilingual joint acoustic model using HMM-SGMM and RNN-CTC.</td>
</tr>
<tr>
<td>3.2</td>
<td>Various approaches for sub-sampling the acoustic sequence.</td>
</tr>
<tr>
<td>3.3</td>
<td>Using Language Information in a multilingual ASR system.</td>
</tr>
<tr>
<td>3.4</td>
<td>Performances of Multilingual ASR using a context independent-phone, context dependent-phone and syllable as basic units.</td>
</tr>
<tr>
<td>3.5</td>
<td>A study exploring the data sharing capabilities of a joint acoustic model using Multilingual ASR.</td>
</tr>
<tr>
<td>4.1</td>
<td>Performance of HMM-DNN and HMM-Resnet(6Res, 8Res, and 10Res) speech recognition systems in terms of Word error rate (WER).</td>
</tr>
<tr>
<td>4.2</td>
<td>Performance of wide-residual networks for speech recognition.</td>
</tr>
<tr>
<td>4.3</td>
<td>Performances of residual neural networks for multilingual ASR system.</td>
</tr>
<tr>
<td>4.4</td>
<td>Performances of Densely connected LSTMs networks for multilingual ASR system.</td>
</tr>
<tr>
<td>5.1</td>
<td>Performance of the proposed algorithm at various threshold values.</td>
</tr>
<tr>
<td>5.2</td>
<td>Performance of the proposed S-Transform based approach at various overlapping criteria. (TAR- true-acceptance rate).</td>
</tr>
<tr>
<td>5.3</td>
<td>Comparing the performance of S-Transform, STFT and Predictability based approaches at various overlapping criteria.</td>
</tr>
<tr>
<td>5.4</td>
<td>A phone level analysis of falsely detected fricatives in S-Transform based approach. (FAR-False-alarm rate).</td>
</tr>
<tr>
<td>5.5</td>
<td>Performance of combined approach at various overlapping criteria.</td>
</tr>
<tr>
<td>5.6</td>
<td>Comparing the performance of proposed S-Transform based approach, Predictability based approach and the combined approach. (TAR- true-acceptance rate).</td>
</tr>
<tr>
<td>5.7</td>
<td>Comparing the performance of proposed combined approach with the state of the art manner class detectors. (TAR- true-acceptance rate).</td>
</tr>
<tr>
<td>5.8</td>
<td>A phone level analysis of falsely detected fricatives in combined approach. (FAR- false-alarm rate).</td>
</tr>
<tr>
<td>5.9</td>
<td>Comparing the performance of various scaling functions ($k_1$, $k_2$ and $k_3$) in detecting fricative regions from speech. The database comprises of 1936 fricatives and 12438 non-fricative phones.</td>
</tr>
<tr>
<td>5.10</td>
<td>Performance of the proposed algorithm at various threshold values.</td>
</tr>
<tr>
<td>5.11</td>
<td>Performance of the proposed approach at various overlapping criteria. (TAR- true-acceptance rate).</td>
</tr>
<tr>
<td>5.12</td>
<td>Comparing the performance of S-Transform, Proposed approach and Predictability based approaches at various overlapping criteria.</td>
</tr>
<tr>
<td>5.13</td>
<td>A phone level analysis of falsely detected fricatives using the proposed approach. (FAR-False-alarm rate).</td>
</tr>
<tr>
<td>5.14</td>
<td>Performance of combined approach at various overlapping criteria.</td>
</tr>
</tbody>
</table>
5.15 A phone level analysis of falsely detected fricatives in combined approach. (FAR- False-alarm rate). .......................................................... 118
5.16 Comparing the performance of proposed approach, predictability based approach and the combined approach. (TAR- true-acceptance rate). .......................................................... 119
5.17 Comparing the performance of proposed combined approach with the state of the art manner class detectors. (TAR- true-acceptance rate). .......................................................... 119
5.18 Performance of speech recognition systems developed by appending the proposed features (deve1, deve2, deve3) with fMLLR features. In the Table l2, l3, l4 corresponds to depth of hidden layers in RNN. .......................................................... 127
5.19 Performance of speech recognition systems developed by appending the proposed features (deve1, deve2, deve3) with fMLLR using WSJ corpus. .......................................................... 128
5.20 Improving the performance of multilingual ASR using the detected fricative evidences. .......................................................... 130

6.1 Distinctive feature groups used in the study. .......................................................... 135
6.2 Performance of distinctive feature network reported in terms of mean frame error rate. .......................................................... 135
6.3 Performance of ASR trained using the proposed distinctive features. .......................................................... 137
6.4 Improving the performance of Multilingual ASR using the distinctive features. .......................................................... 138

7.1 Performance of the proposed speaker normalization approach using one-hot speaker codes to represent the identity of the speaker (WER-word error rate). .......................................................... 143
7.2 Performance of the proposed speaker normalization approach using softmax vectors from SPKID-DNN as representations for the speakers identity. .......................................................... 145
7.3 Performance of the proposed speaker normalization approach using softmax vectors from SPKID-DNN and SPKID-RNN as representations for the speakers identity. .......................................................... 147
7.4 Performance of the speech recognition systems developed using fMLLR features and the proposed normalization approaches. .......................................................... 147
7.5 Improving the Performance of Multilingual-ASR using proposed speaker normalization approach. .......................................................... 148

A.1 Performances of multilingual ASR system developed using sequence-to-sequence with attention model. .......................................................... 165
List of Abbreviations

- AI - Artificial Intelligence
- ASR - Automatic Speech Recognition
- ANNs - Artificial Neural Networks
- BLSTM - Bidirectional long-short-term Memory
- CTC - Connectionist Temporal Classification
- CV - Consonant Vowel
- CMLLR - Constrained Maximum Log-likelihood Transform
- CMVN - Cepstral Mean Variance Normalization
- DCT - Discrete cosine transform
- DFT - Discrete Fourier Transform
- DNN - Deep Neural Network
- DRF - Dominant Resonant Frequency
- EM - Expectation Maximization
- ED - Edit Distance
- FAR - False-Alarm Rate
- fMLLR - Feature-Space Maximum Log-likelihood Transform
- GMM - Gaussian Mixture Model
- HMM - Hidden Markov Model
- IDFT - Inverse Discrete Fourier Transform
• LDA - Linear Discriminant Analysis
• LER - Label Error Rate
• LID - Language Identification
• LSTM - Long Short-Term Memory
• LHUC - Learning Hidden Unit Contributions
• LP - Linear Prediction
• LPC - Linear Prediction coefficients
• LAS - Listen Attend and Spell
• LVCSR - Large Vocabulary Continuous Speech Recognition
• NGD - Numerator of Group Delay
• PER - Phone Error Rate
• PLP - Perceptually weighted Linear Prediction
• PPRLM - Parallel phone recognizers followed by language model
• MFCC - Mel-Frequency Cepstral Coefficients
• MLPs - Multi-Layer Perceptron
• MVN - Mean Variance Normalization
• RNN - Recurrent Neural Network
• RNNLM - Recurrent Neural Network Language Modeling
• ReLU - Rectified Linear Units
• PReLU - Parametrized Rectified Linear Units
• SRILM - SRI Language Modeling
• SGMM - Subspace Gaussian Mixture Model
• SVM - Support Vector Machines
• STFT - Short Time Fourier Transform
• SFTT - Short frequency Time Transform
- SGD - Stochastic Gradient Decent
- TAR - True-Acceptance Rate
- UBM - Universal Background Model
- VTLN - Vocal Tract Length Normalization Scheme
- WER - Word Error Rate
- WFST - Weighted Finite State Transducer
- WSJ - Wall Street Journal
- ZFF - Zero Frequency Filtering
- ZTL - Zero Time-Liftering
Chapter 1

Introduction

1.1 Introduction

India is a multilingual country. Multilingual automatic speech recognition (ASR) systems are widely appreciated in Indian languages. Apart from being multilingual country, most of the Indian population are bi-lingual and the phenomena like code-mixing, word-borrowing are prevalent. An ASR system that can operate on multiple languages and handle code-mixed speech can attain wider usability. This motivated us to develop a multilingual ASR system for Indian languages.

Multilingual ASR systems can be developed by operating the existing mono-lingual ASR systems in parallel with a language ID system as a switch or by developing an integrated multilingual ASR system. In the first approach, the advancements in mono-lingual ASR systems can be directly extended to multilingual ASR systems. But these systems can not handle code-mixing in an efficient way. An Integrated ASR can handle multiple languages in a single system. The efficiency of an integrated ASR system depends on the properties of the languages that are involved in developing the system. This thesis focuses on developing an integrated ASR system for Indian languages.

Indian languages share the same phonetic space but differ in phonotactics. Most of the languages have overlapping phone-set. Phones from different languages are combined and similar phones form different languages are merged to form a common phone-set for Indian languages. As Indian languages are phonetic in nature rule-based parsers can be used to generate lexicons which can map words from different orthographies to a sequence of phones from common phone-set. This common phone-set and lexicon can be used to develop an integrated ASR system for Indian languages. The efficiency of these ASR systems depends on the suitable acoustic modeling approach and the target acoustic modeling unit. The target acoustic model unit i.e., mono-phone or tri-phone or syllable will be efficient when target units share the same acoustic space across different languages. The acoustic model that can efficiently learn to produce the target sequence from the acoustic sequence in presence of language variability is required. This thesis explores a suitable acoustic model and suitable acoustic modeling unit (mono-
phone or tri-phone or syllable) for developing an integrated ASR for Indian languages using the common phone-set. This thesis focuses on improving the performance of this integrated ASR system by exploring residual and dense networks as acoustic models and by extracting more informative features form speech signal such as fricative and distinctive features. The performance of these systems is improved by using speaker embeddings to normalize the speaker variabilities. The scope and the organization of this thesis are briefly presented in the below sections.

1.2 Scope of the Thesis

1.2.1 An Exploration Towards Joint Acoustic Modeling for Indian Languages

This chapter develops an integrated ASR for Indian languages using a common phone-set. A joint acoustic model for multiple languages can address various challenges involved in developing a multilingual ASR system. This chapter studies the amenability of two different acoustic modeling approaches for training a joint acoustic model for Indian languages. HMM-SGMM and RNN-CTC based acoustic models are explored for developing joint acoustic models. HMM-SGMM acoustic models use context-dependent tri-phones as target units while RNN-CTC based acoustic models use context independent mono-phones as target units. Three Indian languages namely Telugu, Tamil and, Gujarati are considered for the study. A common phone-set has been used for developing a joint acoustic model. From the experimental results, it can be observed that the joint acoustic models trained with RNN-CTC have performed better than the SGMM system. A degradation in the performance is observed in HMM-SGMM joint acoustic model compared to HMM-SGMM monolingual models, which is not the case with RNN-CTC based joint models. The work has been reported in [6]. This chapter explores to study the suitable acoustic modeling unit among context-independent phone, context-dependent phone, syllable for developing an integrated ASR. The data-sharing properties of these systems and the use of language ID information are explored in this chapter.

1.2.2 Exploring Residual Neural Networks for Developing Multilingual Speech Recognition Systems

This Chapter, explores the effectiveness of residual neural networks for developing ASR systems. Optimizing a deeper network is more complex task than to optimize a less deeper network, but recently residual network has exhibited a capability to train a very deep neural network architecture and are not prone to vanishing/exploding gradient problems. Along with the depth of the residual network, we also
study the criticality of the width of the residual network. It has been observed that at a higher depth, the width of the networks is also a crucial parameter for attaining significant improvements.

Mono-lingual ASR systems are developed using a 14-hour subset of WSJ corpus. It has been observed that the residual networks have performed better than deep feed-forward neural networks. The work has been reported in [7]. Later the residual architectures have been explored for developing a multilingual ASR using a joint acoustic model. Stacked LSTM networks are used to train joint acoustic models. Stacking Bi-directional LSTMs along the depth has increased the parameters in all the hidden layers. In this work, a linear projection layer is used after every layer to reduce the output dimension of the LSTM layer and thus reducing the number of parameters. LSTMs with linear projection (LSTM-LP) has performed better compared to typical LSTM layers. Residual connections in LSTM-LP networks have improved the performance of multilingual ASR. A residual networks has a skip connection from the previous layer providing an extra path for gradient. This chapter also explores densely connected networks which are a variant of residual networks. In a densely connected LSTM-LP, a skip connection exists from the present layer to all the layers below it. The densely connected LSTM-LP networks have further improved the performance of multilingual ASR systems.

1.2.3 Using Fricative Landmarks in Speech Recognition Systems

This chapter explores the use of landmark-based evidences for improving the performance of multilingual ASR system. This work is mainly focused on detecting the fricative landmarks and using the detected evidences for improving the performance of a speech recognition system.

Two prime acoustic characteristics of fricatives are the concentration of spectral energy above 3 kHz and having noisy nature. Spectral domain approaches for detecting fricatives rely on capturing the information from spectral energy distribution. In this work, Stockwell-Transform (S-Transform) based time-frequency representation is explored for detecting fricatives from continuous speech. S-Transform based time-frequency representation exhibits a progressive resolution which is tailored for localizing the high-frequency events (i.e. onset and offset of fricative regions) with time. Spectral evidence computed from S-Transform based time-frequency representation is observed to perform better compared to the spectral evidence computed from short time Fourier transform (STFT). The existing predictability measure based approach relies on capturing the noisy nature of fricatives. A phone level comparative analysis is carried out between S-Transform and predictability measure based approaches and the phone distribution of the detected fricatives is observed to be complimentary. A combination of S-Transform and predictability based approaches is put forth for detecting fricatives from continuous speech. Apart from detecting the presence of a fricative, the proposed S-Transform based approach and combined approach exhibit better accuracy in detecting the boundaries of fricatives i.e. extracting the durational...
The work presented in [8] uses a time-frequency representation (S-Transform) for detecting fricative regions in the speech signal. The detection accuracy of these approaches depends on the efficiency of the employed time-frequency representation. An approach that would not require any time-frequency representation for detecting fricatives from speech has been explored in [9]. A time-domain operation is proposed which emphasizes the high-frequency spectral characteristics of fricatives implicitly. The proposed approach aims to scale the spectrum of the speech signal using a scaling function $k^2$, where $k$ is the discrete frequency. The spectral weighting function used in the proposed approach can be approximated as a cascaded temporal difference operation over speech signal. The emphasized regions in spectrally weighted speech signal are quantified to detect fricative regions. Contrasting the spectral domain approaches, the predictability measure based approach in literature relies on capturing the noisy nature of fricatives. Proposed approach and the predictability measure based approaches rely on two complementary properties for detecting fricatives, a combination of these approaches is put-forth in this work. The proposed approach has performed better than the state of the art fricative detectors. To study the significance of the proposed evidence, an early fusion between the proposed evidence and the feature-space maximum log-likelihood transform features (fMLLR) is explored for developing speech recognition systems.

1.2.4 Using Distinctive Features for Improving Multilingual ASR

This chapter explores to incorporate cues from speech production mechanism into an ASR system. There have been attempts to use the cues from speech production mechanism in a statistical ASR. This chapter explores a statistical framework for obtaining distinctive features form the speech signal. Features that can specify phonemic contrast between words of a language are termed as distinctive features. The task of obtaining the distinctive features form speech signal is treated as a multi-label classification problem. The distinctive feature vector used in this study comprises of a 28-dimensional binary vector. Neural networks are trained to produce the distinctive feature sequence form the acoustic sequence. The distinctive features predicted by the networks are further combined with the conventional features through early fusion and ASR systems are developed.

1.2.5 Speaker Normalization for Low Resourced Scenarios

This work explores a speaker normalization approach, which is efficient when the meta-level information about the speaker ID is not available while training ASR systems. The efficiency of these approaches has been explored for developing multilingual ASR systems. The present study explores speaker normal-
ization methods for end-to-end speech recognition systems that could efficiently be performed even when a single utterance from the unseen speaker is available. It is hypothesized that by suitably providing information about the speaker’s identity while training an end-to-end neural network, the capability to normalize the speaker variability could be incorporated into an ASR system. The efficiency of these normalization methods depends on the representation used for unseen speakers. The identity of the training speaker is represented in two different ways viz. i) by using a one-hot speaker code, ii) a weighted combination of all the training speakers identities. The unseen speakers from the test set are represented using a weighted combination of training speakers representations. The work has been presented in [10].

1.3 Organization of the Thesis

The remaining part of the thesis is organized as follows:
Chapter 1 presents an overview of ASR system and the briefly describes the tasks performed by each of its components.

Chapter 2 presents an overview of multilingual ASR systems. This chapter describes the types of multilingual ASR systems. This chapter presents the literature about various studies directed towards developing multilingual ASR systems.

In chapter 3, multilingual ASR systems have been developed. A joint acoustic model has been trained using common a phone-set. Different acoustic models have been explored for developing joint acoustic models. In this chapter, we have explored the use of various stable units for training a joint acoustic model such as context independent phones, context-dependent phones, and syllables etc.

In chapter 4, we explore the residual neural networks for developing acoustic models. We have initially studied the efficiency of the residual networks for developing monolingual-ASR systems and later the work has been extended for developing multilingual ASR systems using a common phone-set.

In chapter 5, fricative landmarks are detected from continuous speech. Part 1 we propose an approach to detect fricative landmarks using S-transform based approach and, in part 2 we propose spectral weighting approach which would not require any time-frequency representation for detecting fricative regions. The detected fricative regions combined with conventional features by an early fusion and combined features are used for developing mono-lingual and multilingual ASR systems.
In chapter 6, we explore to use distinctive features from speech signal along with conventional features for improving the performance of a speech recognition system. To detect the distinctive features from the speech signal, we train deep neural networks to learn the mapping the between the acoustic features and distinctive features. The task of detecting the distinctive features from the speech signal is posed as a multi-label classification problem.

In chapter 7, we explore a speaker normalization for low resourced scenarios. We explore a speaker normalization approach where the metadata about the speaker is not available. A speaker normalization approach that would not require any information about the speaker in prior. The study has been initially studied in developing monolingual ASR systems and the results are later extended to multilingual scenarios.
Chapter 2

Multilingual Speech Recognition Systems

2.1 Overview of Automatic Speech Recognition Systems

Automatic speech recognition refers to the task of converting spoken acoustic sequence to text sequence. The task of converting the acoustic sequence to a discrete symbol sequence is referred as acoustic modeling, the detected discrete symbol sequence might not be the text sequence. Automatic speech recognition (ASR) systems can be used as an interface of communication in human-computer interaction. The working diagram of an ASR system is shown in Figure 2.1. The initial speech recognition systems were a mere word recognition systems where the acoustic model is a system that is obtained by minimizing the acoustic distance. Developing a large vocabulary ASR system and operating it in an unconstrained environment requires information from various knowledge sources (KS) to be incorporated in an ASR system [11, 12, 13]. Knowledge sources for developing an ASR system are as follows:

- Acoustics - Knowledge about the variability in speech
- Phonetics - Knowledge about the characteristics of speech sounds
- Phonology - Knowledge about the variability of speech sounds
- Prosodics - Knowledge about stress and the intonation patterns
- Lexical - Knowledge about patterns of language
- Syntax - Knowledge about the grammatical structure of language
- Semantics - Knowledge about the meaning of the words
- Pragmatics - Knowledge about the context of conversion.
Blockdiagram describing the components of a typical ASR system is presented in Figure 2.2. Speech signal is converted to digital signal by a transducer. A front-end processing module is employed to convert the digital signal into an efficient and compact representation. The feature vectors derived from the front-end are used by the acoustic model for learning the mapping function between the acoustic sequence and the corresponding token sequence. The token sequence might not always be the required text sequence. The decoder module produces the most probable text sequence form the token sequence which corresponds to the acoustic sequence. The tasks performed by each of the blocks in Figure 2.2 are briefly described in the following sections.
The basic formulation of a speech recognition systems can be given by the following equations [1]

\[
\hat{W} = \arg \max_{W} P(W|Y) = \arg \max_{W} \frac{P(W)P(Y|W)}{P(Y)}
\] (2.1)

Here \( Y \) is the sequence of acoustic vectors \((Y = y_1, y_2, y_3, y_4, \ldots, y_T)\) and \( W \) is the sequence of words \((W = w_1, w_2, w_3, w_4, \ldots, w_n)\). From equation 2.1, it can be observed that \( \hat{W} \) is the most likely word sequence obtained by maximizing the product of \( P(W) \) and \( P(Y|W) \). \( P(W) \) is the probability of occurrence of the word sequence independent of the observed acoustic signal and it is determined from the language model. \( P(Y|W) \) is the probability of observing the acoustic signal \( Y \) given the word \( W \), which is derived from the acoustic model.

### 2.1.1 Front-End Parametrization:

A quasi-stationarity is imposed on the speech signal, i.e. the speech signal is considered to be stationary for an interval of a few milliseconds. In those milliseconds the spectral properties are relatively constant. The front-end parametrization block divides the speech signal into blocks and a smoothed spectral estimate is computed for each block. The typical analysis window is considered to be of 20 ms and the window shift is considered to be of 10 ms. A tapering window like Hanning or Hamming is typically used to divide the signal to blocks. The speech signal is pre-emphasized to reduce the effect of lip-radiation. Typically Fourier analysis or linear prediction analysis is performed to obtain the smooth spectral envelope. The obtained smooth spectral envelop is further processed to obtain the required acoustic vectors.

The most widely used acoustic feature representation is Mel-frequency cepstral coefficients (MFCC). The frequency scale of the spectral envelope is Mel-warped and integrated using triangular bins. The Mel-scale is designed to replicate the frequency response of human ear and has shown to perform well. Typically for 16 kHz speech signal, 24 bins are used for computing the feature vectors. Spectral statistics are made approximately Gaussian by computing a logarithmic transform on the power spectrum of the speech signal. A Discrete cosine transform (DCT) is applied on the log filterbank coefficients. The DCT compresses the spectral information to the lower order coefficients and de-correlates them for subsequent modules. Typically the top 12 coefficient along with the energy value are retained to form a 13-dimensional acoustic feature vector. The first and second order differentials of these coefficients are computed to represent dynamic information of speech. The static coefficients appended to the first (Delta) and second order (Delta-Delta) differential coefficients make a 39-dimensional feature vector. Cepstral coefficients can also be computed using the spectral estimate obtained from Linear Prediction (LP) analysis. The cepstral coefficients obtained by perceptually weighting the LPC spectrum have performed better and these coefficients are termed as Perceptually weighted Linear Prediction coefficients.
(PLP coefficients). Blockdiagram describing the process of computing MFCC features is presented in Figure 2.3.

![Blockdiagram describing the process of computing MFCC features](image)

Figure 2.3: Blockdiagram describing the process of computing MFCC features (The figure has been adapted from [1]).

### 2.1.2 Acoustic Modeling

An acoustic model converts acoustic sequence to a discrete symbol sequence. Acoustic models can be majorly grouped into two types. They are:

- Generic feature-based approaches
- Phonetic feature-based approaches

Both the approaches are briefly described in the following subsections

#### 2.1.2.1 Generic Feature-based Approaches for Acoustic Modeling

In a generic feature-based acoustic model, the speech signal is parametrized to one of the features like MFCC or PLP and a statistical model is employed to learn the mapping between the acoustic sequence and the symbol sequence. These approaches are popularly known as statistical approaches. Some of the most successful and widely used statistical acoustic models are:

1. HMM-GMM based acoustic models
2. HMM-DNN based acoustic models
3. RNN-CTC based acoustic models

4. Encoder-Decoder acoustic models

These models are described briefly in the following subsections.

**HMM-GMM based Acoustic Modeling**  Hidden Markov model-Gaussian mixture model (HMM-GMM) based speech recognition is one of the widely used acoustic models. In an HMM-GMM based acoustic model, HMMs are used to model the temporal variability of speech and GMMs model how well each state fits a frame or a short window of frames [14, 15]. The richness of GMMs to model relations between the acoustic input and state of HMM are exploited for developing speech recognition systems. The HMM-GMM framework is trained using the expectation maximization (EM) algorithm. GMMs have multiple advantages that make them suitable to model the states of HMMs. With enough number of components, GMMs can fit any probability distribution at the desired level of accuracy using the EM algorithm. There has been a significant amount of research that has been done in training GMMs faster and accurate [16]. The basic formulation of HMM can be expressed in the following steps.

Spoken word can be decomposed in to sequence of $k_w$ basic phones, and the sequence is called the pronunciation sequence $q_1^{k_w}$. The likelihood $P(Y|w)$ of multiple pronunciations is computed by

$$P(Y|w) = \sum_Q P(Y|Q)P(Q|w) \quad (2.2)$$

and $Q$ is a particular sequence of pronunciations

$$P(Q|w) = \prod_{l=1}^{L} P(q_l^{w_l}|w_l) \quad (2.3)$$

where $q_l^{w_l}$ is the valid pronunciation sequence of word $w_l$.

In practice each base-phone is represented by continuous density HMM. Traditionally HMMs with left-to-right topology and three hidden states are employed for speech recognition. Each phone is parametrized by transition probabilities $\{a_{ij}\}$ and a output observation distributions $\{b_i()\}$. During the operation of HMM, it makes transition from the current state to one of its connected states and the probability of making a transition from $s_i$ and $s_j$ is termed as transition probability $a_{ij}$. On entering to state $j$ probability of producing a feature vector is obtained from the distributions of $b_j()$. This form of HMMs operate with two basic assumptions.
• States are conditionally independent of all the other states given the previous state (Present state depends only on the previous state).

• Observations are conditionally independent of all the other observations given the state that has generated it.

The most common choice for modeling the distribution \( \{b_j()\} \) is a multivariate Gaussian distribution

\[
b_j(y_t) = \sum_{m=1}^{M} c_{jm} \mathcal{N}(y_t; \mu_{jm}, \sigma_{jm}^2)
\]  

(2.4)

where \( c_{jm} \) is the weight of mixture component \( m \) in state \( j \) and \( \mathcal{N}(y; \mu, \sigma^2) \) denotes a multi-variate Gaussian of mean \( \mu \) and covariance \( \sigma^2 \).

For the given word all the constituent base phones \( q^{(w_1)}, \ldots, q^{(w_l)} \) are concatenated to form a composite HMM then the acoustic likelihood is given by:

\[
P(Y|Q) = \sum_{\theta} P(\theta, Y|Q)
\]  

(2.5)

where \( \theta = \theta_1, \ldots, \theta_{T+1} \) state sequence of the composite model and

\[
P(\theta, Y|Q) = a_{\theta_0}\theta_1 \prod_{t=1}^{T} b_{\theta_t}(y_t)a_{\theta_t}\theta_{t+1}
\]  

(2.6)

where \( \theta_0, \theta_{T+1} \) are the non-emitting and entry and exit states. The model parameters of HMM i.e. \( \lambda = [a_{ij}, b_j()] \) and can be efficiently estimated using forward backward algorithm which is an example of expectation maximization. With a flat start i.e. using all means and covariances as global mean and covariances and all transition probabilities equal. GMM-HMM based acoustic model is trained by_iteratively learning the parameters (\( \lambda \)) using EM algorithm and are guaranteed to improve till they reach a local maximum.

The effect of co-articulation is high when phones are modeled using HMMs, to achieve good phone discrimination different HMMs are to be trained in different contexts. This problem is addressed by training the separate HMMs for tri-phones. Due to the use of tri-phone HMM models, the number of parameters to be trained is huge but the data is too less. To attain trainability with the limited amount of data, all the Gaussian components are tied based on the similarity and the tied Gaussian components are used as states of HMM. Significant improvements in the performance can be attained using tied states in an HMM-GMM based acoustic models [17].

Limitations of HMM-GMM Acoustic Models
• HMMs can only model the first order temporal dependencies of states.

• GMMs learn the mapping between acoustic vector and state of HMM, but GMM cannot use a window of a sequence of acoustic features.

• Every state of an HMM should be represented only by a single GMM but not multiple.

• GMMs with a large number of components use their parameters inefficiently as only a subset of data is used for training each parameter.

• GMMs require a de-correlated input as they use diagonal covariance and cannot benefit from the multiple input frames from temporal context.

**HMM-DNN based Acoustic Models** Despite the advantageous properties of GMM, they have certain limitations as mentioned in the above subsection. The states of the HMM can be modeled using a single deep neural network where each tied-state is a separate class. Traditionally shallow neural networks have been explored for this task and performances produced by the shallow networks could not challenge well explored GMMs. Availability of larger sized datasets, infrastructure to train in affordable time and development of algorithms to train neural networks with higher depths and larger sizes have improved the performance of ASR systems. Using new learning methods several groups have shown that deep neural networks (DNNs) have outperformed GMMs when used as acoustic models for large vocabulary speech recognition [18, 19, 20, 21, 22, 23, 24, 25].

A DNN is a basic feed-forward artificial neural network that has more than one layer of hidden units between its inputs and outputs. Each hidden unit $j$, typically uses the nonlinear activation function to map its total input from a layer below i.e., $x_j$ to scalar state $y_j$ and it sends to the layer above.

$$y_j = activation(x_j)$$  \hspace{1cm} (2.7)

$$x_j = b_j + \sum_i (y_i w_{ij})$$  \hspace{1cm} (2.8)

where $b_j$ is the bias for unit $j$, $i$ is an index describing the units below the layer and $w_{ij}$ is the weights on a connection to unit $j$ from unit $i$ in the layer below. In a multi-class classification the output unit $j$ converts $x_j$, into a class probability $p_j$ by a softmax nonlinearity given by

$$p_j = \frac{e^{x_j}}{\sum_k e^{x_k}}$$  \hspace{1cm} (2.9)

where $k$ is the index over all classes.

A single DNN is trained for all the states and it can be discriminatively trained. DNNs are efficient when there is a large amount of data. HMM-DNN based acoustic model can be trained with a window
of input frames to exploit the temporal contextual information. The alignment between the target and
the label for training the DNN is obtained using the HMM-GMM system. Apart from these advantages
some of the limitations of HMM-DNN based acoustic models are given below

Limitations of HMM-DNN Acoustic Models

- HMMs can only model the first order temporal dependencies of states, but higher order temporal
dependencies might also be useful for speech recognition.

- In an HMM-DNN framework, DNN is trained to classify the targets at the frame level but the
error measure is related to a sequence level transcription.

- In an HMM-DNN framework, while training DNNs target alignments for every frame are obtained
using HMM-GMM based system. DNNs can be trained better when there is a better alignment
between the frame level targets. This leads to an iterative procedure where the performance cannot
be improved over a certain level.

- An HMM-DNN model requires a pronunciation dictionary to map the words to phone sequences.

- In an HMM-DNN system, phones cannot be modeled directly, and tri-phones have to be modeled
due to a large number of tri-phones states have to be tied using a decision tree making the decoder
more complex.

- Expertise is required to build good decision trees.

RNN-CTC based Acoustic Models  The acoustic model is expected to predict the sequence of labels
from unsegmented data. Recurrent neural networks (RNNs) appear to be powerful learners for such tasks
but, using RNN to learn the mapping between the features and targets from the pre-segmented data is
limiting its efficiency. Many of the above-mentioned problems can be solved by directly training networks
to produce the label sequences. The connectionist temporal classification (CTC) proposed in [26] is used
to train an acoustic model that would not require a pre-segmentation. The basic formulation of CTC
based acoustic model is presented below:

The input and target sequences are \( x = (x_1, x_2, \ldots, x_T) \), \( z = (z_1, z_2, \ldots, z_L) \) and \( L \leq T \), \( L^* \) is the set
of all possible label sequences and \( L \) is the set of labels. \( S \) be the dataset used to develop the acoustic
model and \( \hat{S} \) is the test set. The aim of every example of \( S \) is to train a temporal classifier \( h : X \rightarrow Z \)
to minimize the error measure called a label error rate (LER).

\[
LER(h, \hat{S}) = \frac{1}{|\hat{S}|} \sum_{(x,z) \in \hat{S}} \frac{ED(h(x), z)}{|z|}
\]  (2.10)
where \( ED(p, q) \) is the edit distance between \( p \) and \( q \).

CTC network has a softmax output layer with one label more than the labels in \( L \) given by \( \hat{L} \). The initial \( L \) labels are the probabilities of observing the corresponding labels and extra label is for a blank symbol. Let \( y^t_k \) is the activation of output unit of \( k \) at time \( t \), which defines a distribution over the set \( \hat{L}^T \) of length \( T \).

\[
P(\pi|X) = \prod_{t=1}^{T} y^t_{\pi_t}, \quad \forall \pi \in \hat{L}^T
\]

(2.11)

A many-to-one mapping is defined by a map \( \beta \) by removing the blanks and repeated labels.

\[
\beta(ab-a) = \beta(aa-barb) = aab
\]

(2.12)

Intuitively, it is outputting a new label when the output switches from predicting no label. The Label sequence is obtained by the below expression

\[
P(l|x) = \sum_{\pi \in \beta^{-1}(l)} P(\pi|x)
\]

(2.13)

The output of the classifier is expected to produce the most probable label sequence.

\[
h(x) = \arg \max_{l \in \hat{L}^T} P(l|x)
\]

(2.14)\]

(2.15)

For training a network with CTC, we require an efficient way of computing the conditional probability \( P(l|x) \). The conditional probability of \( P(l|x) \) is computed as the sum over all the paths and it is computationally very complex. But this problem is solved by a dynamic programming algorithm called forward-backward algorithm.

\[
\alpha_t(s) = \sum_{\beta(\pi_1:s) = l_1:s} \prod_{p=1}^{t} y^p_{\pi_p}
\]

(2.16)

where \( \alpha_t \) can be calculated recursively from \( \alpha_{t-1}(s) \alpha_{t-1}(s-1) \)

\[
\alpha_t(s) = \alpha_{t-1}(s) + \alpha_{t-1}(s-1)
\]

(2.17)

To allow the blanks in between the label sequence the utterance of length is \( l \) is added with blanks at the beginning, end and between every pair of labels making the length of \( \hat{l} = 2|l| + 1 \). All the label sequences can start with blank or non-blank labels. Initialization is given by the rules:
\[ \alpha_1(1) = y_b^1 \] (2.18)
\[ \alpha_1(1) = y_l^1 \] (2.19)
\[ \alpha_1(s) = 0; \forall s > 2 \] (2.20)

The recursion is given by:
\[
\alpha_t(s) = \begin{cases} 
\alpha_t(s)y_l^s & \text{if } l_s = b \text{ or } l_{s-2} = l_s \\
(\alpha_t(s) + \alpha_{t-1}(s-2))y_l^s & \text{otherwise}
\end{cases}
\] (2.21)

The CTC loss is given by:
\[
P(l|x) = \alpha_T(\hat{l}) + \alpha_T(\hat{l} - 1)
\] (2.22)

where \( \alpha_T(\hat{l}) \) is the label sequence and \( \alpha_T(\hat{l} - 1) \) is the label sequence without final blank.

\[
\alpha_t(s) = 0; \; \forall |l| - 2(T - t) - 1
\] (2.23)

which indicates the lack of enough number of times steps to complete sequence.

CTC training allows the network to be trained in end-to-end fashion without any pre-segmentation [27]. It also allows the acoustic model to predict the character sequences rather than phone sequences [28]. It has shown reasonably good performance without the use of language model and lexicon. The information present from the lexicon and language model are learned directly from the large data by deep recurrent neural networks [29]. WFST based decoders are integrated to work with CTC-objective function and this system can utilize the information from lexicon and language model [30]. As RNN-CTC trains on the phones or characters directly, it would not require any decision trees and decoder is not complex. The model can be trained to produce the character sequences from the acoustic sequences and such systems would not require a phonetic lexicon.

**Limitations of RNN-CTC Acoustic Models**  CTC-simplifies the sequence level error function as a product of frame level error functions (i.e., independence assumption), which essentially mean that it still does frame level classification. CTC deals with this problem by replicating the same label or blank label so that consecutive frames may correspond to the same output label or blank token.

**Encoder-Decoder Acoustic Models**  The typical encoder-decoder model comprises of an RNN to encode the input acoustic sequence to the encoded sequence. The decoder RNN is an autoregressive model which generates the sequence of labels. At each decoder time-step the attention mechanism helps
the decoder to focus on certain parts of encoded acoustic sequence [31]. These models do not have the
conditional independence assumption while producing the label. The input sequence \( x = (x_1, x_2, \ldots, x_L) \)
and the output of an encoder RNN is given by \( h = (h_1, h_2, \ldots, h_L) \) and output \( y \) is the sequence of
phonemes. Label output \( y_i \) is generated by the following process:

\[
s_i = Recurrence(s_{i-1}, y_{i-1}, g_{i-1})
\] (2.24)

\[
\alpha_i = Attend(s_i, \alpha_{i-1}, h)
\] (2.25)

\[
g_i = \sum_{j=1}^{L} (\alpha_{i,j}, h_j)
\] (2.26)

\[
y_i = generate(s_i, g_i)
\] (2.27)

Though these networks offer a richness in their design, they require large amount of data to train [32].
Further studies incorporated content aware attention along with location-based attention mechanisms
to work for utterances of different length from the training utterances [32, 2, 33]. Networks with a
combination of content and location-based attention mechanisms have performed superior. To reduce the
training time of these networks a pyramidal encoder architecture is used. Recently attention mechanism
is windowed to reduce the training time [33]. Integrating the language model while optimizing the
networks has been explored in [2].
Limitations of Encoder-Decoder Frameworks  Though these models offer richness in their design and operational dynamics are largely understudied. These models need large data and computational infrastructure for training the models.

2.1.2.2 Phonetic Feature-based Approaches for Acoustic Modeling

Phonetic feature-based models contain a lexicon which maps words to the sequence of segments which are described in terms of distinctive features. A feature that can discriminate two different sound units is termed as a distinctive feature. The speech signal is described in terms of discrete phonological segments, i.e., in terms of a bundle of distinctive features [3]. The approach is proposed to be more robust to the variabilities in the pronunciation of the speech signal. The instants of peaks, valleys and rapid changes in the acoustic signal are termed as landmarks. The distinctive features which can be extracted at landmarks are referred as articulatory-free features. The distinctive features that are extracted with articulatory bound provide contextual information. Both this features are explored to map bundle of features with the word. These approaches robustly model the pronunciation variations produced by
the speaker [34]. Generic feature-based approaches require a uniform parametrization, use of multiple parameterizations have proven to be more advantageous in detecting different sound units [35, 36].

Figure 2.5: Blockdiagram describing a phonetic feature-based speech recognition system (The figure has been adapted from [3]).

Though these approaches address certain pronunciation variabilities, preparing the lexicons as a mapping between words and distinctive features requires a lot of effort and expertise. In a landmark-based ASR system, the words are mapped to a bundle of features which are binary. Use of binary features makes the miss-classifications very costly. A probabilistic framework for mapping the distinctive features to the phones has been explored in [4]. Support vector machines (SVM) classifiers are explored to compute distinctive features and the outputs of SVMs are converted to a probabilistic scale. The distinctive features are arranged into a probabilistic graph and the leaf nodes of the graphs are phones. The phone identity is computed probabilistically and the detected phone identities are mapped to the lexicon to obtain words.

A landmark speech recognition system requires front-end landmark detection algorithm for detecting and extracting various distinctive features. The work in [5] presents a landmark detection algorithm for detecting various abrupt landmarks. Energy differences in six frequency bands considered at two different time resolutions 26, 50 ms are used to detect abrupt changes in the signal. These detected
landmarks are used to compute articulatory free distinctive features. The detected features are used to detect the broad phonetic classes. The proposed approach has produced landmarks with high temporal accuracy.

Figure 2.6: Blockdiagram describing probabilistic framework for landmark based speech recognition system (The figure has been adapted from [4]).

Though the landmark-based speech recognition systems have not fully evolved to challenge statistical speech recognition systems, landmark detection algorithms have provided complementary evidence about the characteristics of the speech signal. A bottom-up speech recognition system is explored in [37]. This approach uses evidences detected at multiple levels of a speech recognition system. Various parallel attribute detectors are employed to perform phone classification. The detected phones are composed with a duration model to give a phone sequence. The detected phone sequence is composed with a lexicon, a language model transducers, but before every composition the lattices are pruned to eliminate the weak hypothesis. The study attempts to integrate knowledge acquired from various levels of decoding to prune the lattices.
The studies have explored various ways to use the complementary evidences for characterizing different sound units as a features to statistical speech recognition systems [38, 39, 40, 41, 42, 36, 43, 44]. The work presented in [45], proposed an algorithm to detect the burst onset point, the detected landmarks appended with the feature representation has improved the performance of a speech recognition system.

2.1.3 Language Modeling

Language model provides a way for estimating the probability of occurrence of $W_k$ in a sentence such that $W_{k-1}, W_{k-2}, ..., W_1$ are its preceding words. This can be effectively done using N-grams, i.e. $W_k$ depends only on the previous N-1 words.

$$P(W_k|W_{K-1}, W_{K-2}, ..., W_1) = P(W_k|W_{K-1}, W_{K-2}, ..., W_{K-N})$$

N-grams efficiently concentrate on local dependencies to model the syntax and semantics. The probability of each N-gram can be efficiently estimated from text based on the co-occurrences without any explicit linguistic rules and the grammar.
Typically speech recognition systems have shown better performances using trigrams. For a vocabulary of \( K \) words the trigrams computed are around \( K^3 \), which is very high number even for a small value of \( K \). Most of the trigrams will not appear in the data, and some will appear a few times. To address high data sparsity in computing trigrams, discounting and back-off strategies are employed. Discounting refers to reducing the trigram counts that are more frequently occurring in the data such that the probability mass is redistributed among the less frequently occurring trigrams. Back-off is done when the trigram occurrences are too few, the trigram probabilities are replaced by scaled bi-gram probabilities.

SRILM and IRSTLM are the widely used language models and recently neural network based language models such as RNNLM have shown significant improvements.

2.1.4 Decoding

For performing the recognition, the sequence of words \( \hat{W} \) must be found such that the eq. 2.1 is maximized. This can be seen as a search problem, and two major approaches can be used to obtain the best path i.e. breadth-first search and depth-first search. In a depth-first search, the most promising hypothesis is traced until the end of the speech utterance, wherein the breadth-first search all possible hypothesis are traced in parallel. The breadth-first search at bellman’s optimality is termed as viterbi decoding, viterbi decoding is the widely used in speech recognition systems.

To perform decoding, a decoding graph is constructed such that there is a branch to every possible starting word and a branch from the starting word to every possible second word and so on. When fully populated this graph has all possible word sequences. Each word in the graph is replaced by the pronunciation sequences and multiple pronunciations are joined in parallel. All common phone models are merged with identical contexts and have common entry points. Finally, the decoding graph consists of nodes (phones, characters, HMM-states) connected by node transitions and the word end nodes connected by word transitions.

Any path from the start node can be reached, and the score is evaluated by adding all log state transition probabilities in the path. Such paths can be obtained by a movable token placed at the end of each node and at the end of each word. The token has the score and the history of the words it has traversed. Any path can be traversed in the graph by moving the node from the current node to the adjoining node. Search problem now can be seen as a token passing algorithm. Initially, a single token is placed at the start node of the tree, and they are copied to all the connecting nodes. If more than one token reaches the node, then the best token is retained. After processing all the acoustic vectors, the end words are processed and the words token with the highest score is the best path and the history in the token reveals the best possible word sequence. Though the token passing algorithm is guaranteed to find all the possible best paths, it requires a lot of space and time to compute. To make it viable at every frame a token with the best score is taken and the tokens with the score lying a below the beam
width from the best score are not retained. Computing the paths which lie within the beam-width from
the best path becomes efficient and tractable.

### 2.1.5 Evaluating the Performance of an ASR System

The performance of an ASR system is measured in terms of Word Error Rate (WER). The criticality
of this metric lies when it has to be computed when the reference and hypothesized utterance are of
different length. In computing the WER, initially the hypothesized and the reference utterance is aligned,
and WER is computed by using the expression shown in

\[
\%WER = \frac{S + I + D}{N}
\]  

(2.29)

Here \( S, I, D, C \) are the number of substitutions, insertions, deletions and correct words respectively. \( N \)
is the total number of words present in reference utterance i.e., \( N = (S + D + C) \).

### 2.2 Multilingual Speech Recognition Systems and the Issues Involved

ASR systems have achieved a significant progress in the past decade. Mono-lingual ASR systems
have reached to a performance comparable to human level on certain tasks. Multilingual ASR systems
are not on par with the mono-lingual ASR systems. Recently there has been significant interest in
developing multilingual ASR systems. An ideal multilingual ASR should:

- Seamlessly handle multilingual speech
- Allow cross lingual knowledge-transfer

In a multilingual environment, operating multilingual ASR widely appreciated and it would improve
the scope of usage of technology. Collecting data from a multilingual environment is much tedious
task than to acquire data from a monolingual environment. The development of a multilingual ASR
is always associated with the data scarcity. There have been many studies to improve the efficiency
of cross-lingual knowledge transfer such that the burdensome of collecting the data can be reduced.
Some major approaches for cross-lingual knowledge transfer have been briefly described in the following
subsection 2.2.2.

Approaches for developing multilingual ASR systems can be majorly grouped into three different
scenarios i.e.

1. Switching between the monolingual models with a language identification system (LID) as a front-
end module
2. Operating all the mono-lingual ASR systems in parallel and selecting the most probable hypothesis with a back-end LID

3. An Integrated multilingual ASR

In the first approach, an initial LID system is employed to detect the language of the spoken utterance. The language predicted from the LID systems is used to switch between the monolingual ASR systems. The blockdiagram of this approach is presented in 2.8. The approach is a straightforward extension of mono-lingual ASR system and the advancements in mono-lingual ASR systems can be translated for developing multilingual ASR systems. The performance of this systems is highly dependent on the front-end LID systems. It has been observed that the performance of LID system degrades when the length of the utterances decreases. Performance of LID is much lower when a 1 sec utterance is used compared to the performance for a 3 or 5 sec utterance. In this approach, monolingual ASR systems have to be developed for all the languages and, the developed systems have to be operated in parallel. Certain parameters of the systems could be shared to reduce the data requirements. In this approach, the ASR system has to wait till the decision of LID is obtained. Handling code-mixed speech with these systems would be a complex task.

![Blockdiagram describing a multilingual ASR with LID as a front-end switch.](image-url)

In the second approach, all the mono-lingual ASR systems are to be operated in parallel. The language ID of the spoken utterance can be obtained by using the acoustic cues and phonotactics obtained from the parallel monolingual ASR systems. The blockdiagram of this approach is presented in 2.9. This approach would not require any front-end LID system, but all the ASR systems have to be operated in parallel. LID systems developed using the posterior probabilities of an acoustic model is used to select the most probable hypothesis from the ASR. Operating this model is computationally very intensive. In
this scenario, the prediction about the language of the spoken utterance would be more accurate as the LID system uses information from multiple-ASR systems and the acoustic sequence of the utterance. This approach requires multiple ASR systems to be operated in parallel which increases the computational load and complexity of ASR while operating. Handling code-mixed speech with these systems would be a complex task.

![Diagram of multilingual ASR system](image)

Figure 2.9: A multilingual ASR which is operated using multiple monolingual ASR systems in parallel and a back-end LID system various processes involved in developing a speech recognition system.

Integrated ASR systems can also be explored for developing multilingual ASR systems. An integrated ASR system has a language independent acoustic model, pronunciation model, and language model. The data across the languages could be shared in developing these systems. The blockdiagram of this approach is presented in 2.10. Having global phone-set is one of the efficient ways of developing a multilingual acoustic model. A superset of phones for all the languages are pooled to form a global phone-set. The lexicons from different languages are merged to form a single lexicon to develop a single ASR system. The phones with similar characteristics are merged to improve the efficiency of the global phone-set. Such systems can efficiently be a solution for developing multilingual ASR systems. But the set of rules or the approaches for combining phones form different languages and merging the similar phones have to be developed. The expertise in phonetics and multiple languages is required for developing global-phone-set [46, 47, 48]. The languages should share the common phonetic space exhibit higher efficiency in data sharing while using global phone-set. The acoustic model used should be able to utilize the properties efficiently. The suitable acoustic model that can handle variability of various languages is required for developing a single joint acoustic model for multiple languages. Lack of proper rules to form a global phone-set or lack of a suitable acoustic modeling approach in an integrated ASR.
would under-perform compared to multiple-monolingual ASR systems.

Figure 2.10: Blockdiagram describing an integrated multilingual ASR system

2.2.1 Language Identification Systems

Growing interest in multilingual ASR systems had brought a lot of scientific attention towards the development of language identification systems (LID). Language identification system refers to a module that can tag the input speech with its language identity [49]. LID system has multiple applications in multilingual dialog systems and information querying systems. Multilingual ASR systems demand a LID system as a front-end switch and operate between multiple languages [50, 51, 52, 53, 54, 55].

LID systems can be broadly categorized to two types:

- Explicit LID systems
- Implicit LID systems

Explicit systems convert the acoustic sequences to an intermediate representation such as phones, senones or tokens and temporal relations among these sequences are exploited for developing LID systems. Implicit approaches directly use the acoustic level information to predict the language identity (language ID) [55, 53]. The LID systems using high-level information such as phonotactics, phone frequency, syntax are highly reliable and robust [56, 57, 58, 59, 60]. Parallel phone recognizers followed by language model (PPRLM) systems have been demonstrated to work as robust LID system, but this systems need multiple acoustic models to be operated in parallel [50, 52]. An independent phone recognized followed by language model has been explored for developing an LID system for discriminating 4 languages in [50, 52]. A language independent acoustic model is employed to convert the acoustic sequence
to token sequence and language models such as SRI Language Modeling (SRILM) and Recurrent Neural Network Language Modeling (RNNLM) have been explored to model the temporal relations among the tokens for developing large-scale LID systems [58]. An acoustic model is used to convert the acoustic sequence to phones, and the bottleneck representation from the acoustic model is used to train an RNN for developing LID systems [61].

2.2.2 Approaches For Cross-lingual Knowledge Transfer

The approaches directed towards cross-lingual knowledge transfer are widely explored. Cross lingual knowledge transfer reduces the burdensome data requirements for a multilingual ASR system. The representations from the hidden layers of a neural network trained with multilingual data are considered as language independent bottleneck features. The language independent bottle neck features can be used to train monolingual acoustic models. As the bottleneck is trained with data pooled form various languages, these bottleneck features can be used to train ASR systems in low-resourced scenarios [62, 63]. A hat-swap architecture has been proposed where all the layers of a neural networks are shared between the languages except the last layer. The last layer is trained with monolingual data and the systems have shown better performances due to cross-lingual knowledge transfer [64]. Suitable approaches for adapting a trained network to a new language has been explored in [65]. The study proposed a system with two network architecture where the initial network is a multilingual bottleneck feature extractor and adaptation networks trained with the in-domain data. These systems have shown improvements when the in-domain data is limited [65]. The neural networks trained using multilingual data are used for initializing the monolingual acoustic models, this approach of transfer learning have lead to faster and better convergence of acoustic models [66].

2.2.3 End-to-End Approaches for Multilingual ASR Systems

Approaches for cross-lingual knowledge transfer has been widely studied in the context of hybrid acoustic models i.e., HMM-DNN. Advancement in neural networks have lead to integrated multilingual ASR systems. RNN-CTC based acoustic model can learn to produce the character sequence from the acoustic sequence without any pre-segmentation. This model is known to work even when the label sequence is the corresponding character sequence, thus overcoming the need for the phonetic dictionary [67]. The set of characters from different languages are combined to form a combines character-set and RNN-CTC based acoustic model is trained to learn the mapping between the acoustic sequence to the multilingual character set. Augmented character-set is explored for developing multilingual ASR systems [67, 68, 69]. A language specific gating mechanism is learned internally to modulate the internal representations of a neural network in a language specific way [68]. Mono-lingual RNN-CTC acoustic
model is cross lingually adapted using learning hidden unit contribution (LHUC) based adaptation method in [70]. The language feature vectors derived from an LID network are used to scale the hidden unit representations for adapting the trained network cross lingually [71, 69, 72]. Along with CTC models attention models have also been studied for multilingual speech recognition. Encoder-Decoder with attention have been explored to learn the mapping between the acoustic sequence and the orthographic symbol sequence in [73, 74, 75] for developing multilingual speech recognition systems.

2.3 Multilingual ASR Systems in Indian Scenarios:

Apart for the challenges in developing ASR technology, developing ASR in Indian scenarios have some specific challenges. Some of the challenges that an ASR systems has to address specific to Indian scenarios are described below:

- Low resourced
- Multilingual nature
- Code-mixing
- Handling Text
- Handling language
- Suitable acoustic model

Resources for developing an ASR can be majorly grouped into three types i.e.

- Lexical resources
- Transcribed data
- Meta level Information

As Indian languages are phonetic in nature, pronunciation models can be generated from a simple rule-based parsers [76, 77, 78, 79, 76]. Low resource in Indian languages majorly reflects back to the lack of transcribed data. The meta-level information about the data environment and speaker etc. can be helpful in improving the performance of the system. The pronunciation sequence generated by the rule-based parser is displayed in the Figure.2.11. A common phone set has been developed for Indian languages and the phone sequence corresponding to the common phone-set is obtained [76, 77].
India is a multilingual society. There are around 23 official languages, and each language is spoken by a sufficiently similar number of people. Most of the people in India are bilingual in nature. A generic ASR with wider usage should be able to handle multiple languages. Apart from official languages, India has many dialects. A dialect is a form of a language spoken in a particular part of the country or by a particular group of people. A Dialect may contain some new words, a different grammar and different pronunciations which are different from the standard dialects of the language. Apart from Indian languages, English is also widely spoken in India. Due to the existence of multilingual society switching the codes can often happen between any two languages, handling code-switching is one of the crucial challenges for Indian ASR. Code-switching happens in bilingual societies where languages is altered during the conversation.

There have been some studies for developing multilingual acoustic models, but multilingual language models have not been widely studied. The efficient approaches for multilingual language modeling would improve the efficiency of the systems greatly. Handling language information from the spoken utterance is one of the crucial aspects of developing a multilingual speech recognition system. Detecting the language from the spoken utterance and using it efficiently in a multilingual ASR system would improve the performances. An acoustic modeling approach that is suitable for training multilingual models using common phone-set is a central issue for developing a multilingual ASR system.

There have been some initial attempts for developing multilingual speech recognition systems in Indian scenarios. Data from 12 different languages is collected as a part of a consortium project and the details of the work is reported in the report [80]. The collected data is transcribed using the common phone-set and the phone classification systems have been developed.
Chapter 3

An Exploration Towards Joint Acoustic Modeling for Indian Languages

3.1 Introduction

In chapter 3 we have explored a suitable acoustic model which can act as a joint acoustic model for three different languages. The two acoustic models investigated in this study are Sub-space Gaussian mixture models (HMM-SGMM), and recurrent neural networks (RNN) trained using connectionist temporal classification (CTC) objective function. SGMM based approach models context dependent tri-phone states as a basic unit, where RNN-CTC models the context independent phone as a basic unit. This study also explores different levels where the language ID information of the utterance needs to embedded into a multilingual speech recognition system.

3.2 Related Work

Multilingual automatic speech recognition (ASR) systems are widely appreciated in India. Minimal attempts have been made due to the scarcity of resources required for developing state-of-the-art large vocabulary continuous speech recognition (LVCSR) systems [81, 82, 83, 84, 85]. Resources required for developing an ASR system can be broadly grouped to two aspects, i.e. transcribed data and pronunciation models. As Indian languages are phonetic in nature, the pronunciation models could be generated from a simple rule-based parser [77, 78, 79, 86]. Low resource in an Indian scenario majorly reflects lack of transcribed data. Indian languages have certain advantageous properties such as sharing the same phonetic space and differing in phonotactics [77]. Indian languages differ in prosody i.e., duration, intonation, and prominence associated with a syllable [77]. These properties could be beneficial for developing multilingual ASR systems, but the selection of an appropriate phone-set and the suit-
able acoustic modeling approach are crucial for achieving better performances. This study explores two different acoustic modeling approaches for training a joint acoustic model for Indian languages.

Traditional speech recognition systems use hidden Markov model-Gaussian mixture models (HMM-GMM) as acoustic models. In an HMM-GMM acoustic model, HMMs model the tri-phones and the states of these tri-phones (senones) are modeled using GMMs [18]. Despite the advantageous properties of GMMs such as faster convergence and capability to model any probability distribution, GMMs fail to model data on non-linear manifold [18]. Though hybrid acoustic models, i.e., HMM-Deep neural network (HMM-DNN) have performed better than HMM-GMM systems, the frame level senone labels required for training DNNs have to be obtained from an HMM-GMM system [26]. The hybrid systems suffer from a downside that the objective function which is optimized while training is much different from the true error measure of ASR system (Sequence level transcription accuracy) [32]. Advancements in deep neural networks have greatly influenced the performances of speech recognition systems. Recent developments such as connectionist temporal classification objective function and attention mechanism have enabled end-to-end training for developing acoustic models [32, 2, 33]. End-to-End networks have enriched acoustic models to train without any pre-trained alignments between the acoustic sequence and the label sequence. End-to-End training reduces the mismatch between the true error measure of the system and the objective function which is optimized while training. Apart from the theoretical advantages, end-to-end networks require large amounts of data to train and generalize well. Studies have shown that in the presence of larger sized datasets the performance of end-to-end systems is equivalent to hybrid systems using a pronunciation model and language model [87, 88, 89]. Recent Subspace mixture model has performed superior to traditional speech recognition systems, they have exhibited efficient parameter estimation in limited data scenarios [90, 91].

Multilingual ASR using global phone-sets have been studied in [46, 47, 48]. Sharing some acoustic model parameters has been explored for training multilingual ASR system [92, 64, 93]. Multi-task architectures have been explored for training a multilingual speech recognition systems, a hat swap architecture has been mostly explored where lower layers are shared across languages, but the higher layers are specific to a language [64, 93]. Subspace Gaussian mixture models have been explored for multilingual speech recognition by sharing the subspace defining parameters shared across the languages [90]. The efficiency of grapheme, phoneme-based multilingual speech recognition systems have been studied using RNN-CTC based acoustic models in [94], language feature vectors have been employed in addition to features to condition the systems on language identity. An adaptation mechanism by learning the hidden unit contribution have been explored for multilingual and cross-lingual adaptation methods [70]. Recently multilingual ASR for 9 Indian languages comprising 1500 hrs of data has been presented in [95] using Listen attend and spell (LAS) architecture. This model uses a union of monolingual phone-sets comprising of 960 characters to train a single unified model by jointly optimizing the acoustic model,
pronunciation model, and language model. In-spite of using the union of phone-sets the joint-model has performed better than monolingual models due to the availability of large data for optimizing the model. Most of the studies that have explored multi-task architectures have used multilingual data to train monolingual systems or systems with certain parameters shared across languages. In an operating environment, either a front-end language identification (LID) system has to be used to switch to the corresponding monolingual-acoustic model or the best possible hypothesis from all the monolingual models have to be chosen. The former approach demands a front end LID to be very accurate and robust, and the latter requires all the monolingual systems to be operated in parallel. Operating these systems in code-mixed environments gets really complex and challenging. An acoustic model that can seamlessly handle multiple Indian languages without any prior information of the language is required. This study considers the use of a common phone-set as an efficient approach for handling multiple languages in a single system. This study explores acoustic modeling approaches that are more suitable to train a joint acoustic model for Indian languages using common phone-set.

3.3 Multilingual Speech Recognition Systems using Joint Acoustic Models

3.3.0.1 Database

The database is provided by Speech Ocean.com and Microsoft which is released as a part of “Low Resource Speech Recognition Challenge for Indian languages-Interspeech 2018”. The database comprises of data from three different languages i.e., Telugu Tamil and Gujarati. The dataset comprises of a 40 hour training set and 5 hour testing set.

3.3.0.2 Common phone-set

In this study, a common phone-set was used which is a shared representation across languages. A parser to convert utf8 to IT3 [96] has been used to convert the text to the IT3-format [78]. The text in IT3 is used to generate the pronunciation sequences for all the words. All the multilingual ASR systems used in this study are trained using common phone-set.

3.3.1 RNN-CTC based Acoustic Modeling

An end-to-end ASR has been developed using deep bidirectional long short-term memory networks (LSTMs) [97] using connectionist temporal classification (CTC) [26] objective function.
3.3.1.1 Deep-bidirectional LSTMs

For the input sequence $S = (s_1, s_2, s_3, \ldots, s_T)$ the sequence of hidden states computed by a bidirectional-LSTMs layer is given by $H = (h_1, h_2, h_3, \ldots, h_T)$. At each time step $t$ the forward and backward hidden outputs are concatenated and used as input to the next layer i.e. $h_t = [\overrightarrow{h_t}, \overleftarrow{h_t}]$. The hidden state sequence is computed using the following equations.

\begin{align*}
    i_t &= \sigma(W_{is}s_t + W_{ih}h_{t-1} + b_i) \quad (3.1) \\
    f_t &= \sigma(W_{fs}s_t + W_{fh}h_{t-1} + b_f) \quad (3.2) \\
    o_t &= \sigma(W_{os}s_t + W_{oh}h_{t-1} + b_o) \quad (3.3) \\
    c_t &= \tanh(W_{cs}s_t + W_{ch}h_{t-1} + b_c) \quad (3.4) \\
    h_t &= o_t \cdot \tanh(c_t) \quad (3.6)
\end{align*}

The activations of input gate, forget gate, output gate and memory memory cells are given by $i_t$, $f_t$, $o_t$, $c_t$ respectively. The weight matrices $W_s$ connect inputs with the units, whereas $W_h$ connects the previous hidden states with the units.

3.3.1.2 CTC-objective function

Connectionist temporal classification (CTC) is an objective function used to align two sequences of different length [26]. An end-to-end speech recognition system can be trained using CTC objective function and it would not require any frame level alignment between the acoustic sequence and the label sequence. An additional blank label is added to the set of target labels and the probability of not emitting any label at a particular time step is represented using a blank label. As the acoustic sequence and the label sequence are of different lengths an intermediate representation called CTC-path is used to learn the alignment between the acoustic sequence and label sequence. CTC-path gives the target label sequence at frame level and is obtained by repeating the non-blank labels, inserting a blank between two different non-blank labels. The target label sequence is represented by the set of all possible CTC-paths.

An input sequence of $X = (x_1, x_2, x_3, \ldots, x_T)$, the probability of a label sequence being $L$ is obtained by summing the conditional probability $P(l|X)$ over all possible CTC-paths.

\begin{equation}
    P(L|X) = \sum_{l \in \Omega(L)} P(l|X) = \sum_{l \in \Omega(L)} \prod_{t=1}^{T} P(l_t|x_t) \quad (3.7)
\end{equation}
Here $\Omega(y)$ is the set of all possible CTC-paths. The conditional probability of a label at each time step, $P(\hat{l}_t|x_t)$ is estimated using the network. The network is trained using gradient descent to maximize equation 3.7, and the forward-backward algorithm is employed to compute gradients [26].

### 3.3.2 Subspace-Gaussian Mixture Models

In a conventional acoustic model, states of HMM are modeled using GMMs. A high-dimensional super-vector of GMM parameters from all the states is expected to lie on a low dimensional manifold common to all the states [91]. Though SGMM uses the GMM as its underlying distribution, the parameters in an SGMM are shared across the states. These parameters describe the sub-space of the GMM parameters. The individual states can be described using relatively low-dimensional vectors which are the coordinates in the subspace. SGMM can be seen as a compact representation for HMM state distributions. SGMMs perform significantly better than HMM-GMM. In limited data scenarios, SGMMs have delivered much better performances [90].

### 3.4 Experiments, Results & Discussion

#### 3.4.1 HMM-SGMM vs. RNN-CTC as a Joint Acoustic Model for Multilingual ASR

Two different acoustic modeling approaches have been explored for training a joint acoustic model, i.e., SGMM and RNN-CTC. Recipes from Kaldi-toolkit have been used for training SGMM models. The SGMM-models trained during the study have used 8000 sub-states, and a diagonal UBM of 400 dimensions. End-to-End speech recognition system in this study has employed deep bidirectional long short-term memory networks (Bi-LSTMs) optimized using CTC objective function. A hyper-parameter search has been performed to obtain optimal architectural choices. It has been observed that Deep Bi-LSTMS layers with 3, 4 layers are optimal for training mono-lingual and joint acoustic models respectively, each layer comprised of 320 units. A learning rate of 0.0001 is used with a batch size of 1. The learning rate is reduced by a factor of 0.5 when a decrease in the validation accuracy is observed. RNN-CTC networks are optimized using Adam optimizer [98] with exponential decays on first and second order momentums are given by 0.9 and 0.99 respectively. The performances of various acoustic models are presented in Table 3.1. Monolingual acoustic models trained using SGMM model are presented in row 2. Row 3 is the performance of joint acoustic model trained using data from all three languages. Performances of monolingual acoustic models trained using RNN-CTC are presented in row 4.
<table>
<thead>
<tr>
<th>Acoustic model</th>
<th>Telugu</th>
<th>Tamil</th>
<th>Gujarati</th>
</tr>
</thead>
<tbody>
<tr>
<td>Monolingual-SGMM</td>
<td>21.69</td>
<td>19.63</td>
<td>14.51</td>
</tr>
<tr>
<td>Joint-SGMM</td>
<td>26.53</td>
<td>24.65</td>
<td>17.41</td>
</tr>
<tr>
<td>Monolingual-CTC</td>
<td>21.68</td>
<td>21.10</td>
<td>15.07</td>
</tr>
<tr>
<td>Joint-CTC</td>
<td>21.28</td>
<td>21.12</td>
<td>14.86</td>
</tr>
</tbody>
</table>

Apart from handling the acoustic variabilities due to the language, multilingual speech recognition should handle different orthographies of various languages. As Indian languages share the same phonetic space, there can be words with the same pronunciation in different languages. When different orthographies are used in the system with a common phone set, this word-phone sequence pairs stand as different entities in the pronunciation model. Such words could be erroneously decoded even when the acoustic model has produced the correct phone sequence. This could be efficiently avoided by using text in IT3 format [96]. IT3-format are any other language independent mapping which could map the words in different languages with the same phone sequence as a single entity would be more beneficial in training a multilingual ASR. We have considered IT3 as the language independent phone sequence based representation. In the present database, out of 140K words, there are 2K words with same phone sequences but different orthographies due to different languages. The performance of joint acoustic models trained using RNN-CTC has been presented in row 5 of Table 3.1. The transcriptions from training utterances in IT3-format have been used to train a trigram language model. The pronunciation model contains unique words from all the three languages in IT3-format and the corresponding phone sequences.

3.4.2 Sub-sampling the Acoustic Sequence for training RNN-CTC

RNN-CTC based acoustic model is trained to align two sequences of different lengths without any alignment from a pre-trained model. RNNs being sequential in nature reducing the sequence length has reduced the training time, this has been achieved by using pyramidal architectures [31, 33, 89]. Sub-sampling the acoustic sequence by a factor of 2 or 3 has not shown any degradation in the performance of RNN-CTC based acoustic models. Features from successive acoustic frames are concatenated reducing the sequence length by a factor of 2. Various approaches for sub-sampling the acoustic sequence has been explored and the results are tabulated in Table 3.2. To subsample the acoustic sequence by a factor of 2 alternate frames can be dropped or successive frames can be appended, the performances attained...
by this sub-sampling methods are presented in row 3, 6 of Table 3.2. Using a frame shift of 20 ms and a frame size of 30 ms for computing the features would also reduce the acoustic sequence by a factor of 2 and the performance obtained by this sub-sampling has been presented in row 3. The training data can be augmented by a factor of 2 dropping even and odd frames alternatively such sampling is termed as Augmented-sub-sampling. Augmented-sub-sampling would reduce the sequence length and also augment the dataset.

Table 3.2: Various approaches for sub-sampling the acoustic sequence.

<table>
<thead>
<tr>
<th>Sub-sampling</th>
<th>Dev set</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dropping</td>
<td>Telugu</td>
</tr>
<tr>
<td>Frame-shift 20 ms</td>
<td>21.97</td>
</tr>
<tr>
<td>AUG-sub-sub-sampling</td>
<td>21.90</td>
</tr>
<tr>
<td>Appending frames</td>
<td>21.45</td>
</tr>
</tbody>
</table>

3.4.3 Exploring the use of Language ID in a Multilingual ASR

For regularizing the network, 10% of the randomly sampled training examples are chosen and white Gaussian noise ($\sigma=0.075$) is injected into these features [29]. The performance of these networks is presented in row 3 of Table 3.3. The performance of the networks trained by sub-sampling (by appending the successive frames) and using Gaussian noise regularization is presented in column 4 of Table 3.3. To condition the joint acoustic model with the language identity, one-hot language representative vector has been used in tandem with the features as in [95] and this system has reduced the word error rates. The performance of a joint acoustic model conditioned on language identity is presented in row 5 of Table 3.3. Performance of speech recognition systems using a joint acoustic model, a common pronunciation model and the language model specific to that language is presented in row 6 of Table 3.3. Rather than conditioning the acoustic model with language ID, using language ID information to select the language model was more effective in reducing the WER. The Language ID systems required for these scenarios could be explicit LID systems which have been more accurate [99].
Table 3.3: Using Language Information in a multilingual ASR system.

<table>
<thead>
<tr>
<th>Acoustic model</th>
<th>Dev set</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Telugu</td>
</tr>
<tr>
<td>Joint-CTC</td>
<td>21.28</td>
</tr>
<tr>
<td>Joint-CTC-Gaussian noise</td>
<td>21.25</td>
</tr>
<tr>
<td>Joint-CTC-Sub-samp+Gaussian noise</td>
<td>21.26</td>
</tr>
<tr>
<td>Joint-CTC-Language ID</td>
<td>21.41</td>
</tr>
<tr>
<td>Joint-CTC-Sub-samp+Gaussian noise mono-LM</td>
<td><strong>20.61</strong></td>
</tr>
</tbody>
</table>

3.4.4 Exploring Various Basic units for Joint Acoustic Modeling

Typically RNN-CTC models use context independent phones as the basic unit. A joint acoustic model is trained to model context-dependent phones, i.e., the phones which occur at starting middle and end of the words and singletons are considered as independent tokens. RNN-CTC is trained to minimize the token error rate where the tokens used are context-dependent phones and the results are presented in row 2, 3 of Table 3.4. Though the token accuracy of these systems is 7% lesser but the WERs are comparable to the joint acoustic model trained with context independent phones. Use of context-dependent phones has helped in pruning out competing decoding paths. Using the context-dependent phones has increased the number of tokens by a factor of 4 and this has lead to an increase in training time, i.e., time for computing CTC loss.

We have used syllable as a basic unit for training multilingual joint acoustic model. The multilingual text in IT3 format is converted to the syllables. The structure of a basic syllable is $C^nV^nC^n$, where $C$ is the consonant and $V$ is the vowel. The pronunciation dictionary is made such that it contains the mapping of the word with its corresponding syllable (CV-clusters) sequence. The RNN-CTC model is optimized to produce a minimum token error rate where the tokens are the syllables and the results obtained are presented in row 3 (Joint-CTC-syl) of Table 3.4. In a syllable-based acoustic model, we have observed that there are many syllables with very less number of repetitions in the dataset. To avoid the data sparsity, the syllables with less number of examples in the dataset are further split into context independent phones. A mixed system comprising of syllables and context independent phones is trained and the results are reported in row 4, 5 of Table 3.4. The syllables with less than 500, 100 examples are
split to context independent phones and the mixed systems trained respectively are reported as row 4, 5 (Joint-CTC-syl-100-examp, Joint-CTC-syl-500-examp) of Table 3.4.

Table 3.4: Performances of Multilingual ASR using a context independent-phone, context dependent-phone and syllable as basic units.

<table>
<thead>
<tr>
<th>Acoustic model</th>
<th>Telugu</th>
<th>Tamil</th>
<th>Gujarati</th>
</tr>
</thead>
<tbody>
<tr>
<td>Context-dependent phones +Sub-samp+ Gaussian noise+mono LM</td>
<td>20.65</td>
<td>19.82</td>
<td>14.61</td>
</tr>
<tr>
<td>Context-independent phones-Sub-samp+Gaussian noise mono-LM</td>
<td>20.61</td>
<td>20.16</td>
<td>14.19</td>
</tr>
<tr>
<td>Joint-CTC-syl</td>
<td>25.57</td>
<td>26.30</td>
<td>17.07</td>
</tr>
<tr>
<td>Joint-CTC-syl-100-examp</td>
<td>23.54</td>
<td>22.38</td>
<td>15.50</td>
</tr>
<tr>
<td>Joint-CTC-syl-500-examp</td>
<td>23.32</td>
<td>22.14</td>
<td>15.46</td>
</tr>
</tbody>
</table>

3.4.5 Data Sharing in a Multilingual Joint Acoustic Model

A joint acoustic model is expected to share the data among various languages. To study the data sharing capability of an RNN-CTC based joint acoustic model, we have trained an acoustic model with data from two languages excluding the data from the third language and the performances of these ASR systems are presented in Table 3.5. Column 2 of Table 3.5 is the performance of a joint acoustic model obtained by training with all the languages. Column 3, 4 and 5 are the performances of ASR systems trained by excluding Tamil, and Gujarati and Telugu respectively. From column 3, 4 and 5 of Table 3.5, it can be observed that the joint acoustic model has exhibited data sharing across the languages. The performances of ASR systems for the languages that are used while training the joint model are almost unchanged. For the language that has been excluded while training, the model has performed relatively well, i.e., though the WER appears to over 50% but the model has not seen any utterances from that language. By just adapting the trained joint acoustic model with 10% of the training data from the excluded language (here the excluded language is Telugu) a very significant drop in the WER can be observed.
<table>
<thead>
<tr>
<th>Training data</th>
<th>Dev set</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Telugu</td>
</tr>
<tr>
<td>Telugu-Gujarati</td>
<td>20.71</td>
</tr>
<tr>
<td>Tamil-Telugu</td>
<td>21.90</td>
</tr>
<tr>
<td>Tamil-Gujarati</td>
<td>53.67</td>
</tr>
<tr>
<td>Telugu adapted(10%)</td>
<td>31.70</td>
</tr>
</tbody>
</table>

### 3.4.6 Results & Discussion

The performance of joint acoustic models trained using HMM-SGMM is poorer than the performance of HMM-SGMM monolingual models. Using HMM-SGMM for training a joint acoustic model has lead to an increase in word error rate (WER). Indian languages share the same phonetic space and differ in phonotactics. The tri-phones modeled by HMM do not share common acoustic distribution across different languages. This has lead to the poor performance when a joint acoustic model is trained using HMM-SGMM. Performance monolingual RNN-CTC based acoustic model is less than the monolingual HMM-SGMM system which is in accordance with the earlier studies. The joint acoustic model trained using RNN-CTC has performed better than the monolingual systems. RNN-CTC acoustic models are trained to model context independent phones. The variabilities due to multiple languages have been effectively handled using RNN-CTC. In the earlier studies, it has been observed that conditioning the acoustic model on language identity (language ID) has improved the performance [95, 94]. Mostly the systems have used the global phone-set which is a union of phone-sets from all the languages. When a joint acoustic model is trained to predict the labels from global phone-set the information about the language identity has improved the performance. It has been observed that the systems trained in such fashion more faithful to language ID, upon encountering a wrong language ID the system has transliterated the acoustic sequence using the phone sequence corresponding to the mismatched language ID. The joint acoustic models trained use a phone-set which is same for all the languages, unlike the union of monolingual phone-sets. Conditioning the model to language ID has helped in convergence but has not significantly improved the performance. Sub-sampling the acoustic sequence has reduced the training time. It has been observed that sub-sampling the acoustic sequence by a factor of >3 has affected the convergence of the models. Using the language model specific to that language has reduced
the absolute WER by 0.8% compared to the performance obtained using a combined language model. Rather than conditioning the joint acoustic model on the language ID and using the LID information to select the corresponding language model while decoding the test utterance has been effective. Multi-pass decoding could also be a viable solution using a common language model initially to obtain an initial hypothesis and which could give information about the language of the hypothesized test utterance. A second pass decoding with a monolingual model or re-scoring the lattices with the monolingual language model could be beneficial in building multilingual ASR systems.

3.5 Summary and Conclusions

From chapter 3 it can be observed that a joint acoustic model is an effective solution for training a multilingual ASR system. In an under-resourced scenario, the use of a common phone-set could be an efficient approach for sharing data across the languages. This chapter studies the amenability of various acoustic models, i.e., HMM-SGMM and RNN-CTC for developing a joint acoustic model using common phone-set. It has been observed that end-to-end systems which model context independent phones as a basic unit have performed better than HMM-SGMM systems which models context dependent triphone. RNN-CTC based acoustic models have been more promising while using a common phone-set. Conditioning the joint acoustic model with language ID has not improved the performance. Converting orthographies of various languages to IT3-format can be helpful in handling the words in different languages with the same pronunciation sequences. Use of a joint acoustic model and text in IT3 format could be a viable solution to operate multilingual and code-mixed speech recognition systems irrespective of input languages. Using a monolingual language model in a multi-pass decoding framework would improve the performances.
Chapter 4

Residual Neural Networks as Joint Acoustic Models for Indian Languages

4.1 Introduction

In chapter 3, we have developed multilingual ASR systems using a joint acoustic model. In chapter 3, it has been observed that RNN-CTC has performed better as a joint acoustic model. This chapter explores the effectiveness of residual networks for developing ASR systems. The effectiveness of residual connections are studied in the perspective of HMM-DNN and RNN-CTC based acoustic models. In the initial part of this chapter, HMM-Resnet architecture is explored for developing monolingual ASR system using WSJ corpus. In the latter part of this chapter, residual networks are explored for developing multilingual ASR systems using Residual-RNN-CTC acoustic models. Along with residual networks, densely connected networks which are a variant of residual networks are also explored for developing multilingual ASR systems.

4.2 Related Work

The developments of deep learning methodologies have greatly influenced the performances of speech recognition systems. Deep learning methodologies majorly aim to learn the feature hierarchies in which lower level features are composed to form a higher level representation. To learn better representations a deeper network has to be trained. The superiority of networks when the depth an increased has been studied for various tasks such as speech recognition, language processing, and AI are described in [100]. Training and optimizing the deeper neural network is more complicated than training and optimization of a shallow network, the difficulties in training deeper architectures with the sigmoid activation units and random initializations have been studied in [21]. Recent developments in deep learning methodologies can be majorly consolidated as development on learning methodologies, initializations, and development of
new activation function to train deeper architectures formed by stacking fully connected layers. Though there is a significant amount of progress achieved in reducing the effect of exploding/vanishing gradients by the use of activation functions like ReLU [22], PReLU [101] and normalized initializations [101] and normalizations like Batch-normalization [102], but optimizing a neural network with very deep architecture is an open problem and there have been many attempts to train deeper networks with plain SGD. Learning strategies like curriculum learning, continuation methods [103], mollifying networks [104] and use of noisy activation functions [105] have been studied to aid the optimization of a highly non-convex objective function.

Initial attempts to train the deep networks were studied in deep supervision [106], where auxiliary loss forked in the intermediate layers, to provide a short path for back-propagating the gradients where the forked layers have two gradients, i.e., from main loss and auxiliary loss. Despite the better performance of deep supervision, the irrelevance of auxiliary loss at test time, the mismatch between the training and testing objective functions is a major drawback. Recent architectures called Highway networks have been successful in training neural network architectures with arbitrary depth using SGD. Highway networks are characterized by pathways which allow unimpeded information flow across the layers of a network known as highways of information [107]. In a highway network, a data-driven gated mechanism is employed to control the pathways of information and in a way, they decide whether the layer should learn the mapping function or its residual counterpart. Though the highway networks have provided a capability to train an arbitrary depth architecture, but improvements in the performance were not significantly high even for at 100 layer depth. Further studies have shown that replacing the data-driven gating mechanism of highway networks with an identity mapping has given rise to a new class of networks called residual networks [108]. The residual networks are enriched with advantageous properties such as any depth networks could be easily trained with SGD, use of the identity mapping across the information highways makes the network to learn only residual mappings and a network of any depth is not prone to vanishing and exploding gradients [109]. The effectiveness of identity mapping for a residual connection and their ease of training has been studied in [110]. Unlike the Deep supervision where an auxiliary loss is forked at intermediate layer to reduce the effective depth while training the networks, in a residual network the identity mappings provide a better way of reducing the effective depth in training the network. Due to this mechanism of reducing the effective depth a residual neural network has shown better ease in convergence and better generalization. This motivated us to study the influence of residual networks and their effectiveness for the task of speech recognition.

Apart from increasing the depth of the network, widening the networks have also shown better performance on image classification tasks [111]. In this study, we explore the effectiveness of increasing width of the network along with residual connections for the task of speech recognition. Unlike the other classification tasks speech recognition task has to handle many variabilities such as speech, speaker and
emotion variabilities that are naturally expected to exist in speech data, so the classifier should be capable of exhibiting better generalization to these variabilities.

4.3 Experimental setup of HMM-Residual Neural Networks For Speech Recognition

4.3.1 Database

Speech data from Wall Street Journal corpus (WSJ) [112] has been used during the study. We have used a 14 hrs subset of WSJ corpus (si284-set) for training the speech recognition systems, and eval-92 are used as validation and test sets respectively. The text from the training transcriptions is used to build a language model. Alignments for the training data are obtained from an HMM-GMM based tri-phone speech recognition system, and these are used for training the deep neural networks.

4.3.2 Feature Extraction

Mel-frequency cepstral coefficients (MFCC) extracted from speech signal are spliced over 9 frames (±4) in time to form a 117-dimensional feature vector. A linear discriminant analysis (LDA) is used to make this 117-dimensional input vector to a 40-dimensional vector and a feature-space maximum likelihood linear regression (fMLLR) transform is used for speaker variability normalization. The speaker normalized 40-dimensional vector is spliced in time over 11 frames (±5) resulting in a 440-dimensional feature vector. This entire feature extraction is performed using kaldi-pdnn toolkit [113].

4.3.3 Experimental Details

A deep neural network with six hidden layers comprising of ReLU units i.e., (440R-1024R-1024R-1024R-1024R-1024R-1991S) is used as a baseline system and this network is termed as stacked network. The categorical entropy of the outputs is used as the loss function. ADADELTA [114] is used as an optimizer. The dropout of 0.1 is used for all the hidden layers [24]. During the work, a continuous increase in the validation loss for five successive epochs is considered as an early stopping criterion.
4.3.4 HMM-Residual Neural Network Architecture

Residual networks are employed to model the senones. The posterior probability obtained for a frame of speech using a Resnet is used as the emission probability of HMMs. If $H(x)$ is the mapping learned by feedforward deep neural network where $x$ is input, then the network can also learn $H(x) - x$ mapping, but with a different ease of learning [108]. Thus the residual function ($F(x)$) thus becomes $F(x) := H(x) - x$, the residual network is implemented just as any deep neural network with a constraint $H(x) := F(x) + x$.

![Residual block](image)

Figure 4.1: Residual block used in this study.

The residual blocks used are presented in Figure 4.1, These blocks are termed as $Res$. In the residual block ($Res$) the first weight layer contains W1-ReLU units and the second weight layer contains W2-ReLU units. During the study, the second weight layer W2 always has the fixed number of units which is equal to the dimension of input (440), so that the output of weight layer W2 is directly added to input without any zero-padding. A dropout regularization is used in-spite of Batch normalization, a dropout factor of 0.1 is used with all the residual blocks.
4.3.5 Results & Discussion

The residual network architectures are formed by stacking the residual blocks shown in Figure 4.1. During the study, 5% of the training set is held out as a validation set, and frame error rates are computed over that set. During the study, residual network architecture formed by stacking ‘n’ residual blocks followed by a softmax is termed as nRes architecture. The performance of speech recognition systems developed using residual networks of varying depth (6Res, 8Res, and 10Res) is presented in Table 4.1.

Table 4.1: Performance of HMM-DNN and HMM-Resnet(6Res, 8Res, and 10Res) speech recognition systems in terms of Word error rate (WER).

<table>
<thead>
<tr>
<th>Speech recognition system</th>
<th>stacked network</th>
<th>6Res</th>
<th>8Res</th>
<th>10Res</th>
</tr>
</thead>
<tbody>
<tr>
<td>dev-93</td>
<td>8.72</td>
<td>8.68</td>
<td>8.56</td>
<td>8.51</td>
</tr>
<tr>
<td>eval-92</td>
<td>5.19</td>
<td>5.07</td>
<td><strong>4.86</strong></td>
<td>5.14</td>
</tr>
</tbody>
</table>

Row 1 of Table 4.1 presents various network architectures and row 2 and 3 of Table 4.1 are the WERs of various speech recognition systems on eval92 and dev93 sets respectively. As we increase the depth of the network from 6Res to 8Res an improvement in the performance can be observed from columns 2, 3 of Table 4.1. Network with 10Res has exhibited more over-fitting nature compared to the 6Res, 8Res architectures can be noted from Figure 4.2(a),(b) and the similar nature is also apparent in Table 4.1. Though the stacked network, residual networks (6Res, 8Res and 10Res) has exhibited similar performance on the validation set which can be seen from Figure 4.2, but the residual networks have exhibited better generalization than the stacked network which can be noted in Table 4.1. The performance of speech recognition systems developed using residual networks of varying depth (6Res, 8Res and 10Res) is presented in terms of frame error rate in Figure 4.2.
Figure 4.2: Comparing the performance of speech recognition systems developed using stacked network and residual network in terms of frame error rate. Figures are generated from the subset of WSJ corpus mentioned in subsection 4.3.1

From Figure 4.2.(a), it can be observed that residual networks have shown good ease in training. The performance of deep neural network formed by stacking several fully connected layers (stacked network) is shown by a solid line. From Figure 4.2.(b) the performance of residual networks is slightly better than the stacked network. The performance of speech recognition systems developed using residual networks in terms of word error rate (WER) is presented in Table 4.1. The WERs are reported on dev93 and eval93 sets. As the dev93 set shares the same data environment and the vocabulary size as that of the training set, an overfit model appears to perform better. In this study we consider the performance on eval92 as a measure of models generalization.
The criticality of the width of the residual network is also studied. Wide-residual networks are designed by stacking the residual blocks presented in Figure 4.1 with weight layer W1 comprising of 1024 units and weight W2 comprises of 440 units. The performance of speech recognition developed using wide-residual networks is presented in terms of WERs in Table 4.2.

Table 4.2: Performance of wide-residual networks for speech recognition.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>eval92</td>
<td>5.19</td>
<td>5.12</td>
<td>5.07</td>
<td>4.94</td>
<td><strong>4.77</strong></td>
</tr>
<tr>
<td>dev93</td>
<td>8.72</td>
<td>8.68</td>
<td>8.43</td>
<td>8.60</td>
<td>8.62</td>
</tr>
</tbody>
</table>

Row 1 of Table 4.2 presents various speech recognition systems developed by varying the depth of the wide-residual network. Row 2, 3 of Table 4.2 are the WERs on eval92, dev93 sets respectively. As we increase the depth of the network from 4,10 layers the performance speech recognition system has increased and at 12 layer depth the networks has exhibited an over-fitting nature and the same can be observed in terms of WER. At higher depths i.e., 10 layers the width of the network has shown a significant impact. The residual networks, wide residual networks have shown better generalization properties and an absolute improvement of 0.4% improvement in the WER is obtained. The performance of wide-residual networks in terms of frame error rate is presented in Figure 4.3.
From Figure 4.3, it can be noted that the width of the networks has shown better ease in convergence. The width of the residual blocks is also a critical parameter along with the depth of the network, with the increase in the width the performance of speech recognition systems has significantly improved. The performance of wide-residual networks is significantly higher than residual networks and the stacked network. As we increase the depth of the wide-residual network from 4 to 10 layer an increase in the performance is observed, but as we further increase the depth of the network to 12-layers an overfit in the model can be observed.
4.4 Residual Neural Network as a Joint Acoustic Model for Multilingual Speech Recognition

Residual networks have been effective in training an HMM-Resnet based acoustic models. In this study, we explore the use of residual connections in training joint acoustic models. RNN-CTC architecture mentioned in Chapter 3 section 3.3 is used with a residual connection between the hidden layers.

Table 4.3: Performances of residual neural networks for multilingual ASR system.

<table>
<thead>
<tr>
<th>Acoustic model</th>
<th>Telugu</th>
<th>Tamil</th>
<th>Gujarati</th>
</tr>
</thead>
<tbody>
<tr>
<td>Joint-CTC</td>
<td>21.28</td>
<td>21.12</td>
<td>14.86</td>
</tr>
</tbody>
</table>

Residual connections in neural network architectures have lead to a better convergence [108, 7]. A joint acoustic model has been trained using B-LSTMs with skip connections between two successive hidden layers. The use of these skip connections have eased the convergence during the start of the training but the performance gains are less significant. The performance of joint acoustic model using residual connections has been presented in Table 4.3. Column 2 is the performance of RNN-CTC based joint acoustic model. Column 3 is the performance of RNN-CTC based joint acoustic model with residual connections.

The RNN-CTC networks presented in Table 4.3 use stacked bidirectional-LSTM networks. The output vector of a bidirectional LSTM is twice the hidden size, i.e., the outputs of forward and backward LSTMs are concatenated. This increases the number of parameters in the hidden layers when Bi-LSTMs are stacked. To avoid this a linear projection layer is used at the output of every Bi-LSTM to reduce the dimension of the output. This architecture can be used to increase the depth of the network without greatly increasing the number of parameters. This architecture is termed as LSTM-LP and its formulation is presented below.
4.4.1 LSTM Layers With a Linear Projection (LSTM-LP)

For the input sequence \( S = (s_1, s_2, s_3, \ldots, s_T) \) the sequence of hidden states computed by an bidirectional-LSTMs layer is given by \( H = (h_1, h_2, h_3, \ldots, h_T) \). At each time step \( t \) the forward and backward hidden outputs are concatenated and used as input to the next layer i.e. \( \hat{h}_t = [\hat{h}_t^\rightarrow, \hat{h}_t^\leftarrow] \). The hidden state sequence is computed using the following equations.

\[
\begin{align*}
i_t &= \sigma(W_{is} s_t + W_{ih} h_{t-1} + b_i) \quad (4.1) \\
f_t &= \sigma(W_{fs} s_t + W_{fh} h_{t-1} + b_f) \quad (4.2) \\
o_t &= \sigma(W_{os} s_t + W_{oh} h_{t-1} + b_o) \quad (4.3) \\
c_t &= f_t * c_{t-1} + i_t * c_t \quad (4.4) \\
h_t &= o_t * \tanh(c_t) \quad (4.5)
\end{align*}
\]

The activations of input gate, forget gate, output gate and memory cells are given by \( i_t, f_t, o_t, c_t \) respectively. The weight matrices \( W_{is} \) connect inputs with the units, whereas \( W_{ih} \) connects the previous hidden states with the units.

\[
\begin{align*}
\hat{h}_t &= [\hat{h}_t^\rightarrow, \hat{h}_t^\leftarrow] \quad (4.7) \\
\hat{h}_t^\leftarrow &= W_{hp} \hat{h}_t \quad (4.8)
\end{align*}
\]

Here \( W_{hp} \) is the linear projection weight matrix connecting the output of Bi-LSTM and the input of the next layer. Using this projection layer depth of the networks can be further increased without increasing the number of parameters largely. Further in this chapter, the LSTMs described in this section are termed as LSTM-LP layers. Multilingual speech recognition systems mentioned in Chapter 3 section 3.3 are trained using stacked LSTMs and LSTM-LP the performance is tabulated in table 4.4.
4.4.2 Densely Connected LSTM-LP Networks

Residual networks have a skip connection between two successive layers providing an extra path of gradients [110, 109, 111, 107]. Residual connections have improved the stability of the network. Dense connections are a variant of residual networks where every layer is connected to all the layers above it through skip connections [115, 116]. The blockdiagrams describing Residual and dense connections in while stacking LSTM-LP are presented in Figure 4.4.

Figure 4.4: Blockdiagram showing LSTM-LP with residual and dense connections. In the above Figure (a) is the blockdiagram of LSTM-LP with residual connections and (b) is the blockdiagram of LSTM-LP with Dense connections.

RNN-CTC based multilingual speech recognition systems described in chapter 3 have been developed using stacked LSTM-LP, residual LSTM-LP and Dense LSTM-LP networks. The performances of these systems are tabulated in Table 4.4. Column 1 of Table 4.4 is the multilingual speech recognition systems developed and column 2 - 4 are the WERs obtained for Telugu, Tamil and Gujarati test-sets respectively. Row 3 is the joint acoustic model trained by using stacked LSTMs and the architecture of this systems is described in Chapter 3. Row 4 is the performance of ASR trained using Joint-LSTM-LP acoustic model.
with a monolingual language model. Row 5 is the performance ASR using residual LSTM-LP acoustic model and monolingual language model. Row 6 is the performance of densely connected LSTM-LP acoustic model and monolingual language model.

Table 4.4: Performances of Densely connected LSTMs networks for multilingual ASR system.

<table>
<thead>
<tr>
<th>Acoustic model</th>
<th>Telugu</th>
<th>Tamil</th>
<th>Gujarati</th>
</tr>
</thead>
<tbody>
<tr>
<td>Joint-CTC-mono-LM</td>
<td>20.61</td>
<td>20.16</td>
<td>14.19</td>
</tr>
<tr>
<td>Joint-Res-LSTM-LP-CTC-mono-LM</td>
<td>19.95</td>
<td>18.85</td>
<td>13.73</td>
</tr>
<tr>
<td>Joint-Dense-LSTM-LP-CTC-mono-LM</td>
<td>19.57</td>
<td>18.57</td>
<td>13.45</td>
</tr>
</tbody>
</table>

From Table 4.4, it can be observed that LSTM-LP has performed better compared to stacked LSTM layers. To maintain the same number of parameters LSTM-LP network has 6 Bi-LSTM layers where stacked LSTM network has 4 layers. The residual networks have not shown significant improvement in the case of stacked LSTM layers. But in the case of LSTM-LP with increased depth the Residual networks have performed better compared to LSTM-LP. Densely connected LSTM-LPs have further improved the performance of multilingual ASR system. Using the densely connected LSTM-LPs have reduced the absolute WER of Telugu, Tamil and Gujarati by 1, 1.6 and 1.4 (i.e., a relative improvement of 5%, 8.5% and 10%)
4.5 Summary and Conclusions

This chapter explores the effectiveness of residual networks for developing HMM-DNN and RNN-CTC based ASR systems. With the increased depth, the residual networks have exhibited better generalization and convergence properties. HMM-Resnets have shown superior performance compared to HMM-DNN based speech recognition systems and an absolute improvement of 0.4 in WER is observed of 14-hrs WSJ corpus. Along with the depth of the network, we have also explored the criticality of the width of the residual layers. Increase in width of the residual layers along with depth has aided the convergence. At higher depths increase in the width of the network has attained significant improvement in the performance of speech recognition systems.

The efficiency of residual connections is explored in the context of RNN-CTC acoustic models for developing multilingual ASR systems. RNN-CTC based joint acoustic models are trained with residual connections in hidden layers. Stacking Bi-LSTMs directly increases the number of parameters largely. In this work, a linear projection layer is used after each Bi-LSTM to reduce the dimension of the output vector and the networks trained by stacking these layers (LSTM-LP) have performed better. By using LSTM-LP layer the depth of the network can be increased without largely increasing the number of parameters. From the experimental results, it can be observed that, the use of residual connections in stacked LSTM-LP networks has improved the performance of joint acoustic models. The use of dense connections in LSTM-LP networks has further improved the performance. Densely connected LSTM-LPs has reduced the absolute WER of Telugu, Tamil and Gujarati by 1, 1.6 and 1.4 (i.e., a relative improvement of 5%, 8.5%, and 10%). The use of residual and dense connections have eased the convergence while training the networks.
Chapter 5

Using Fricative Landmarks in Speech Recognition Systems

In chapter 5, we explore the use of fricative landmarks for improving the performance of a speech recognition system. Fricatives are characterized by two prime acoustic properties, i.e., having high-frequency spectral concentration and possessing noisy nature. Spectral domain approaches for detecting fricatives employ a time-frequency representation to compute acoustic cues such as band energy ratio, spectral centroid, and dominant resonant frequency, etc. In Part I of chapter 5, We have explored S-Transform based time-frequency representation for detecting fricative regions. S-Transform is a time-frequency representation with a progressive resolution which is tailored for localizing the high-frequency events (i.e., onset and offset of fricative regions). Spectral evidence computed from S-Transform based time-frequency representation is observed to perform better compared to the spectral evidence computed from short time Fourier transform (STFT).

Part I of chapter 5, uses a time-frequency representation for detecting the fricative regions of speech. The detection accuracy of these approaches depends on the efficiency of the employed time-frequency representation. An approach that would not require any time-frequency representation for detecting fricatives from speech has been explored in Part II of chapter 5. In this study, a time-domain operation is proposed which emphasizes the high-frequency spectral characteristics of fricatives implicitly. The proposed approach aims to scale the spectrum of the speech signal using a scaling function $k^2$, where $k$ is the discrete frequency. The spectral weighting function used in the proposed approach can be approximated as a cascaded temporal difference operation over speech signal. The emphasized regions in spectrally weighted speech signal are quantified to detect fricative regions.

From Part I, II of chapter 5 it can be observed that spectral weighting based approach (part II) has performed better in detecting fricative regions from speech. The detected fricative evidence is combined with the conventional features through early fusion and the combined features are explored for developing ASR systems. The combined features are explored for developing both monolingual and multilingual ASR systems. These studies are reported in part III of chapter 5.1.
Part I

Detection of Fricatives using S-Transform
5.1 Introduction

In part I of chapter 5.1, we explore Stockwell-Transform (S-Transform) based time-frequency representation for detecting fricatives from continuous speech. S-Transform based time-frequency representation exhibits a progressive resolution which is tailored for localizing the high-frequency events (i.e., onset and offset of fricative regions) with time. Apart from detecting the presence of a fricative, the proposed S-Transform based approach and combined approach exhibit better accuracy in detecting the boundaries of fricatives i.e., extracting the durational information of fricatives. Spectral evidence computed from S-Transform based time-frequency representation is observed to perform better compared to the spectral evidence computed from short time Fourier transform (STFT).

5.2 Related Work

During the production of a fricative, vocal tract is constricted enough along its length to produce a noisy sound when air is forced through this constriction [117]. The phenomenon of frication introduces a noisy aperiodic energy concentrated above 3 kHz [118]. The class of fricatives considered during the study comprises of sibilant fricatives ( [s], [sh], [z], [zh] ) and nonsibilant fricatives ( [f], [v], [th], [dh] ). Affricates are produced with a supra-glottal closure followed by a burst release and a frication is caused by the turbulence of air associated with the burst release, by their acoustic manifestation affricates are more closer to fricatives than bursts [119]. During the study, affricates [ch] and [jh] are also considered in the class of fricatives. The intensity of sibilants is relatively high compared to that of nonsibilants [39]. During the production of affricate ‘ch’ the frication produced is comparable to a sibilant fricative [119].

On a broad scale, present days automatic speech recognition (ASR) systems are two types [120] i.e., statistical model based ASR and phonetic feature based ASR. In a statistical model based ASR, speech signal is parametrized to one of the features like mel-frequency cepstral coefficients (MFCC), perceptive linear predictive coefficients (PLP) or Gaussian posteriorgrams. Hidden Markov models (HMMs), artificial neural networks (ANNs) and recently deep neural networks (DNNs) [18, 121, 122] are employed to perform regression task between labeled phonetic transcription and acoustic parametrization. In phonetic feature based ASR, phonetic feature specific information is initially extracted from speech and this extracted phonetic features are used for detecting the phone identity. A landmark-based speech recognition system is an example of phonetic feature based ASR. Landmarks are the regions in an utterance, where the acoustic correlates of distinctive features are more prominent [5].

In a statistical model based ASR, a uniform parametrization scheme has to be adopted for recognizing various phones. Contrasting to a statistical model based ASR, in a landmark-based ASR speech
signal can be analyzed at various levels (segmental, sub-segmental and suprasegmental levels) and various parameterizations, which gives a better platform for engaging the information from various theories of phonetics and neuro-science etc. [120, 4]. The landmark-based ASR comprises of a broad manner classifier in its initial stage to reduce the search space [123]. Band energies, spectral peaks, and valleys in various sub-bands are used as acoustic parameters (APs) to compute landmarks [4]. Support vector machines (SVMs) trained with 39-dimensional MFCCs with appropriate framing and frequency shift parameters are employed for detecting fricative broad class [123]. Sub-band energy difference between two frames which are temporally spaced 50 ms, 26 ms apart is studied for obtaining the landmarks of fricatives [5]. A methodology for combining the acoustic-phonetic knowledge with statistical learning is explored for segmenting the speech into five broad manner classes in [124]. The developed combined system has exhibited better performance compared to HMM-based segmentation using 39-dimensional cepstral features. Recurrent neural networks (RNNs) trained with 39-dimensional cepstral features are explored for detecting various phonological features and these features are later used for developing a fricative broad class identification system [36]. Spectral and temporal evidences like dominant resonant frequency, epoch strength and the numerator of group delay at zero-frequency have been used as cues to detect the unvoiced fricatives from continuous speech [119]. The gain of inverse of the all-pole filter at zero-frequency (i.e., predictability measure) is used as an acoustic cue to perform sonorant-fricative classification [43].

Most of the above methods, rely on the information from the short time spectral envelope computed using STFT. For example, in case of a voiced fricative, both frication event and glottal closure event exists mutually exclusively in a single glottal cycle, short time spectral envelope reflects the average spectral characteristics over the entire time frame, the characteristics of a voiced fricative are not represented effectively due to averaging across the time. Usefulness of the proposed approach to represent the events like the voiced fricatives and nonsibilant fricatives of short duration compared to conventional short time Fourier transform (STFT) based approach is shown in the Section 5.4.3, 5.6.

The spectral energy of fricative has a unique frequency distribution i.e., most of the spectral energy is above 3 kHz. This property is explored in detecting fricatives through various parameters like dominant resonant frequency [119], spectral centroid [125], numerator of the group delay spectrum at zero-frequency [119] and band energy ratio [119, 126, 127]. Apart from spectral energy distribution, fricative has a noisy nature. During the production of fricative, the influence of the vocal tract is less compared to a sonorant. The relations imparted by the vocal tract in successive samples of a sonorant are more compared to a fricative i.e., fricative is less correlated signal compared to a sonorant. The correlations among the successive samples of a signal makes the signal more predictable from the past samples. A predictability measure computed as a sum of all the linear prediction coefficients is used
as an acoustic cue for sonorant fricative classification in [43]. Both the acoustic cues (i.e., distribution of spectral energy and predictability measure) reflect two complementary evidences for detecting the fricatives. This work presents a combined approach using the spectro-temporal evidence computed from the S-Transform based spectral representation and the predictability based evidence proposed in [43] for detecting fricative broad class in speech. The method is unsupervised such that the complex process of training and the tedious process of collecting the transcription of the data is not required. This evidence can be used as a complementary evidence to fricative broad class classifier in landmark-based ASR. Apart from ASR, unsupervised fricative broad class classifier is widely appreciated in audio search.

Phonological information of speech can be obtained by observing the acoustic correlates and the observed acoustic correlates are used as acoustic-phonetic descriptors of that particular acoustic event [4]. Duration of an acoustic event is one of the acoustic phonetic descriptor that can be relied on for obtaining information about the finer class (sibilant fricatives, nonsibilant fricatives, affricates and stops) from the broad class of fricatives. One major drawback of frame based analysis is that, the durational cues which can play a key role in discriminating various acoustic events cannot be stringently imposed due to averaged spectral representation over a time frame. For example the pre-frication region in stop sounds show acoustic characteristics similar to the fricatives, the release region of an affricate has an acoustic manifestation similar to that of a fricative. But the duration of frication event pertaining to a sibilant fricatives is much higher compared to pre-frication region in bursts and frication region in nonsibilant fricatives [39]. Duration of frication along with the strength of frication and the existence of associated plosion event helps to discriminate between sibilant fricatives, nonsibilant fricatives, affricates and the stops. But for extracting the durational cues of various acoustic events like frication, a signal processing tool with good temporal resolution to localize that particular acoustic event (i.e., frication) is a pre-requisite.

In this context, S-Transform based time-frequency representation is explored for detecting the acoustic cues pertaining to fricative landmarks. S-Transform was initially developed for geophysics applications [128] to detect the onset and offset of high-frequency signals. The hypothesis is that the capability of S-Transform to time localize the high-frequency signal gets reflected in detecting the boundaries of fricative regions in speech. Due to better time resolution for high-frequency events it can better represent voiced fricatives (as shown in Section 5.4.3) and durational cues from the detected frication events can also be accurately obtained (as shown in Section 5.6).
5.3 Database

Speech signals collected from a phonetician are used during the course of study. Data consists alveolar, post-alveolar and retroflex voiced fricatives \([z], [3], [\tilde{z}]\) produced in isolation. Apart from that each of the voiced fricative is produced with three vowel contexts ([a], [i], [u]). Each of the voiced fricative is succeed by one among the three vowels [a], [i], [u] to form consonant-vowel (CV) units. The junction between voiced-unvoiced fricatives is studied using three voiced fricatives succeeded by unvoiced fricatives i.e., \([z\rightarrow s], [3\rightarrow f], [\tilde{z}\rightarrow \tilde{s}]\). All the speech signals are repeated three times to form 3 tokens per sound. Speech signals are sampled at 48 kHz with 16 bit quantization are used for the study. During the study, signal is resampled to 16 kHz for analysis.

5.3.1 TIMIT-Database

Performance of the proposed method for detecting the fricative broad class from continuous speech is evaluated using TIMIT database [129]. The labeled boundaries in the TIMIT database are used during the evaluation. Fricatives considered for the present study are \([s], [z], [s\text{h}], [zh], [f], [v], [dh], [th], [ch]\) and \([jh]\). A subset of TIMIT database with 40 speakers with 20 male & 20 female, each speaker containing 10 sentences each of 2 to 3.5 sec long are considered for study. Speech signals sampled at 16 kHz and 16 bits/sample encoding is used during the course of present study. The glottal fricatives [hh], [hv] are not considered during the analysis.

5.4 S-Transform, Implementation and Implications on Speech

S-Transform and the terminology used in this article is adopted from [128]. Recently a lot of scientific interest has been shown in analyzing S-Transform and many other variants such as generalized S-Transform [130], modified S-Transform [131] and inverse of S-Transform [132]. The notations of basic S-Transform are adhered in this article.

Short time Fourier transform (STFT) is one of the most widely used time-frequency representation used to study nonstationary signals.

\[
S(\tau, f) = \int_{-\infty}^{\infty} s(t)g(t-\tau)e^{-j2\pi ft}dt,
\]

(5.1)

here \(s(t)\) is the time domain signal, \(g(t)\) is the window signal. Conversely, to study the time properties of a particular frequency, spectrum of the entire time domain signal \(S(f)\) is windowed by a frequency
window $G(f)$ and the inverse Fourier transform of windowed spectrum is a time-frequency representation termed as short frequency time transform (SFTT) [133].

$$S(f, \tau) = \int_{-\infty}^{\infty} S(f)G(f - \hat{f})e^{-j2\pi\hat{f}\tau}d\hat{f}, \quad (5.2)$$

since windowing for STFT is in time domain, the spectrum averaged over the entire time frame is used as time-frequency representation. Conversely, for SFTT the frequency representation is an average representation over the entire band. SFTT based analysis is more preferable where time boundary is crucial. Use of a window with progressive frequency resolution gives a better time-frequency representation [134].

S-Transform can be viewed as an SFTT with a progressive resolution obtained by a frequency dependent window. S-Transform uniquely combines a frequency dependent resolution of time-frequency space [128]. Continuous S-Transform of a function $s(t)$ is given by

$$S_T(\tau, f) = \int_{-\infty}^{\infty} s(t)\sqrt{\frac{f}{2\pi}}e^{-\frac{(\tau - t)^2}{2\sigma^2}}e^{-j2\pi ft}dt, \quad (5.3)$$

$S_T(\tau, f_k)$ is a one dimensional function of time for a constant frequency $f_k$ showing how the amplitude and phase for this frequency $f_k$ changes with time. A choice of Gaussian window is made owing to its better localization, minimal side-lobes, and symmetry in both time and frequency domains [135]. The prime focus of the study is to explore the temporal resolution of S-Transform for representing frication events.

### 5.4.1 Relation between S-Transform and STFT

If the signal $s(t)$ is windowed using $g(t)$ then STFT is given by

$$S(\tau, f) = \int_{-\infty}^{\infty} s(t)g(t - \tau)e^{-j2\pi ft}dt \quad (5.4)$$

For S-Transform a Gaussian window with zero mean and variance “$\sigma$” is used.

$$g(t) = \frac{1}{\sqrt{2\pi}\sigma}e^{-\frac{t^2}{2\sigma^2}} \quad (5.5)$$

Progressive frequency resolution is achieved by equating variance of the window at a frequency ($f$) to the inverse of that frequency i.e.,

$$\sigma(f) = \frac{1}{|f|} \quad (5.6)$$

As $g(t)$ is a Gaussian function (i.e., even function) $g(t - \tau) = g(\tau - t) \quad (5.7)$
From relations 5.4, 5.5 and 5.7, S-Transform ($S_T$) can be written as convolution of two functions over the variable $t$.

$$S_T(\tau, f) = \int_{-\infty}^{\infty} p(t, f) g(\tau - t, f) dt$$ (5.8)

$$S_T(\tau, f) = p(t, f) * g(t, f)$$ (5.9)

where \( p(t, f) = s(t)e^{-j2\pi ft} \),

$$g(t, f) = \frac{|f|}{\sqrt{2\pi}} e^{-\frac{j2f^2}{t^2}}$$ (5.11)

Computing the Fourier transform of $S_T(\tau, f)$ from equation 5.9

$$B(\beta, f) = \mathcal{F} \{ S_T(\tau, f) \} = P(\beta, f) G(\beta, f)$$ (5.12)

Let $P(\beta, f)$ and $G(\beta, f)$ be the Fourier transforms of the $p(t, f)$ and $g(t, f)$.

$$P(\beta, f) = \int_{-\infty}^{\infty} s(t)e^{-j2\pi f \beta} dt$$ (5.13)

$$P(\beta, f) = \int_{-\infty}^{\infty} s(t)e^{-j2\pi (f + \beta)t} dt$$ (5.14)

$$P(\beta, f) = S(\beta + f)$$ (5.15)

$$G(\beta, f) = \int_{-\infty}^{\infty} \frac{|f|}{\sqrt{2\pi}} e^{-\frac{j2f^2}{t^2}} e^{-j2\pi \beta t} dt$$ (5.16)

since \( \mathcal{F} \{ e^{-ax^2} \} = \sqrt{\frac{\pi}{a}} e^{-\frac{x^2}{4a}} \) (5.17)

$$G(\beta, f) = e^{-\frac{2\pi^2 \beta^2}{f^2}}$$ (5.18)

From 5.12 \( B(\beta, f) = P(\beta, f) \times G(\beta, f) \)

$$B(\beta, f) = S(\beta + f)e^{-\frac{2\pi^2 \beta^2}{f^2}}$$ (5.19)

S-Transform is the inverse Fourier transform of the equation 5.20 (for $f \neq 0$).

$$S_T(\tau, f) = \mathcal{F}^{-1} \{ B(\beta, f) \} = \int_{-\infty}^{\infty} S(\beta + f)e^{-\frac{2\pi^2 \beta^2}{f^2}} e^{2\pi \beta \tau} d\beta$$ (5.21)
Implementing S-Transform

- Compute the Discrete Fourier Transform (DFT) of the N-point time series.
- Compute the DFT of N-point Gaussian function to select the frequency range.
- Shift the spectrum of time series such that frequency of the spectrum to be selected matches with the zero-frequency of the frequency selecting Gaussian function and multiply both the signals.
- Compute Inverse Discrete Fourier Transform (IDFT) of the product obtained in the above step to obtain S-Transform representation of the time series.

Owing to the use of frequency dependent progressive window with its variance inversely related to frequency, S-Transform gives better time resolution for high-frequency signals and frequency resolution for low frequency signals.

5.4.2 Implications of S-Transform on the Speech Signal

- Consider a 2 sec utterance with 16 kHz sampling frequency, from equation 5.21 it is evident that a 32000 point DFT is to be computed i.e., the frequency scale is divided in to 16000 points. A Gaussian of length 32000 for selecting one frequency point is needed. A 32000 point IDFT for every frequency point is to be computed and there are 16000 such points. It can be observed that, the length of the input signal dictates the frequency resolution of the S-Transform.

- It is evident that the above mentioned approach for computing S-Transform turns out to be computationally very expensive even for a 2 sec utterance. A block processing approach is used to attain a balance between the computational load and resolution of S-Transform. Suppose for obtaining S-Transform with a frequency resolution of 10 Hz i.e., one frequency point for every 10 Hz, to cover a frequency range of 0 to 8 kHz 800 points are to be used. For obtaining this frequency resolution, a block of 1600 samples are needed to compute S-Transform i.e., 100 ms block.

- S-Transform for every 100 ms block is computed and the computed S-Transform is appended with the S-Transform of remaining blocks in an utterance to get the S-Transform of an utterance.

A restricted study is carried out to study the effectiveness of S-Transform for representing speech signals. During the study, the typical example of a voiced fricative is considered and this study is presented in the following subsection 5.4.3.
5.4.3 Applying S-Transform for Speech Signals

The effectiveness of S-Transform for detecting the boundary of fricative regions is studied in this section and the database mentioned in Section 5.3 is used for this study. A voiced fricative has a unique production mechanism in which voicing and frication are present in a single glottal cycle. Though both the events voicing and frication are present in one glottal cycle both are mutually exclusive i.e., voicing is maintained only in glottal closing phase and frication is present only in the glottal opening phase. Frication energy (high-frequency energy) will be periodic with glottal period, peak in the frication energy occur at glottal opening phase and valley in frication energy are at the glottal closure instants. A periodic frication energy in correspondence with the glottal period indicates the presence of a voiced fricative. Speech signal of a voiced fricative [z] is presented along with conventional wideband spectrogram and S-Transform in Figure 5.1.

![Figure 5.1: Time frequency representation of alveolar voiced fricative [z] using conventional wideband spectrogram and S-Transform. (a) Speech signal (b) S-Transform based spectrogram (c) Conventional spectrogram with 2 ms frame size and 1 sample frame shift.](image-url)
From Figure 5.1 the nature of the voiced fricative can be clearly observed in S-Transform compared to STFT wide-band band spectrogram. As shown in Figure 5.1(b) sharp onset and offset of frication is visible in S-Transform based spectrogram compared to the conventional wide-band spectrogram. As shown in Figure 5.1(c) wide-band spectrogram may show similar results when viewed as plots, but due to the lack of sharp onset and offset points for high-frequency energy the frication energy gets spread to glottal closure cycle. The sharp boundaries of the frication are the major cues for indicating periodic presence of frication in a voiced fricative. Signal representing the periodic presence of frication energy is computed from conventional wide-band, due to the lack of sharp boundaries and the presence of wide-band artifacts computed evidence will eventually lose its periodic nature.

Figure 5.2: Characteristics of alveolar voiced fricative [z]. (a) Speech signal, (b) S-Transform based spectrogram, (c) 1-D temporal curve obtained by computing the spectral energy above 5 kHz, and (d) Short time energy of contour shown in (c).
Figure 5.2 illustrates the production mechanism and its correlation with the acoustic cue developed to detect voiced fricative. Figure 5.2(a) shows the existence of impulsing due to glottal closure instants and frication mutually exclusively in a glottal cycle. From Figure 5.2(b) it is evident that frication is confined to only glottal opening cycle and there is no frication in glottal closure cycle. Figure 5.2(c) is the 1-D temporal curve obtained by computing the spectral energy above 5 kHz and the Figure 5.2(d) shows the short time energy contour obtained by computing energy of Figure 5.2(c) with a window size of 5 ms and a window shift of 1 sample. Peak in frication index pointing the glottal opening cycle indicating high frication energy in glottal opening phase, valley pointing to glottal closure instant indicating the lack of frication. Presence of the voiced fricative in the vicinity of a vowel is presented in Figure 5.3, though vowel also exhibits periodicity similar to a voiced fricative but the strength of the frication energy for a vowel is very less.
The boundary of voiced-unvoiced fricative is presented in Figure 5.4. The efficiency of S-Transform to localize the changes in signal with time can be observed. For an unvoiced fricative, frication energy is aperiodic and is almost a noisy surface with high average energy compared to the voiced fricative. The origin of the aperiodic nature in the frication energy gives the boundary of voiced fricative when succeeded by an unvoiced fricative. Figure 5.4 illustrates characteristics of voiced-unvoiced junction. From Figure 5.4(b) it can be observed that frequency distribution of both voiced and unvoiced frication is almost similar, frication is continuous in an unvoiced fricative. From Figure 5.4(d) i.e., frication energy is high in unvoiced fricative compared to voiced fricative. The effectiveness of S-Transform for detecting the frication regions can be clearly visualized from the Figure 5.4, 5.3.
5.5 Proposed S-Transform based Approach for Detecting Fricatives

The phenomena of frication introduces a noisy aperiodic energy concentrated around above 3 kHz. Though some fricatives have some lower frequency energy but spectral energy of fricatives is mostly concentrated in higher frequency regions i.e., above 3 kHz. Spectral energy above 1.5 kHz is considered as energy due to the frication in a fricative. The ratio of sum of energies of all the frequency bands above 1.5 kHz to sum of energies at all frequency bands below 1.5 kHz at every sample gives a one dimensional time domain signal called frication ratio ($F_r$). The frication ratio is computed as

$$F_r = \frac{\sum_{k=N_f}^{N_f} |S_{T}(k,f)|}{\sum_{k=1}^{N_f} |S_{T}(k,f)|}$$

where $|S_{T}(k,f)|$ is the magnitude response of the S-Transform time-frequency representation. $N_f$ denotes the index of S-Transform coefficient corresponding to 1.5 kHz. $N$ is the S-Transform coefficient corresponding to $\frac{f_s}{2}$ kHz, where $f_s$ is the sampling rate of the speech signal. The obtained frication ratio ($F_r$) computed by the ratio is mean-smoothed with a window of 20 ms to make the temporal evolutions of frication ratio smooth and the evidence is peak normalized and this is termed as ‘frication index’ in this work. A threshold based decision is enforced on the frication index contour to detect the fricative and nonfricative regions. A duration based decision is used to discriminate true evidences (fricatives) and spurious evidences (nonfricative regions). Initially a threshold based decision is used on the frication index and then the hypothesized fricative regions greater than the 10 ms are considered as the true fricative regions. The criteria for choosing the values of threshold is presented in following subsections.

5.5.1 Arriving at the Threshold

A threshold based decision is adapted to discriminate fricative and nonfricative regions. The region of speech, where the frication index is greater than the threshold ‘$\theta$’, is the hypothesized fricative region. The region of speech where the frication index is less than the threshold ‘$\theta$’, is the detected nonfricative region. A duration based decision is used to discriminate true evidences (fricatives) and spurious evidences (nonfricative regions). Initially a threshold based decision is used on the frication index and then the hypothesized fricative regions greater than the 10 ms are considered as the true fricative regions.

The threshold value in this method is empirically obtained using dataset 5.3.1 comprising of data from 40 speakers (20 male and 20 female). Thresholds for the proposed method are obtained using the number of phones correctly classified. The fricative region computed from the algorithm is termed as detected fricative region or predicted fricative region. For a fricative, if the percentage overlap between the
predicted fricative region and the ground truth is greater than the overlap criterion, then the particular
fricative phone is termed as correctly classified fricative phone. The overlapping criterion is employed to
study the effectiveness of the algorithm in detecting the boundary of fricative regions. The nonfricative
region which is falsely detected as fricative by the algorithm is termed as falsely detected fricative re-
gion. A nonfricative phone with falsely detected fricative region greater than 10 ms duration is termed
as falsely detected fricative. In the literature, the percentage of correctly detected fricative phones and
that of falsely detected fricative phones have been reported as true-detection rate and false-alarm rate
respectively.
Table 5.1: Performance of the proposed algorithm at various threshold values.

| Fricative class | Total# Phones | 0.001 | 0.002 | 0.003 | 0.004 | 0.005 | 0.006 | 0.007 | 0.008 | 0.009 | 0.01 | 0.02 | 0.03 | 0.04 | 0.05 | 0.06 | 0.07 | 0.08 | 0.09 | 0.1 |
|-----------------|---------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------|------|-----|
| Sibilant        | 1110          | 1109  | 1108  | 1106  | 1102  | 1101  | 1100  | 1099  | 1098  | 1096  | 1075 | 1054 | 1031 | 1004 | 980  | 952  | 927  | 899  | 604  |     |
| Nonsibilant     | 651           | 492   | 441   | 395   | 367   | 337   | 307   | 290   | 277   | 259   | 239  | 145  | 104  | 77   | 58   | 45   | 38   | 32   | 28   | 18   |     |
| Affricates      | 175           | 175   | 175   | 174   | 174   | 173   | 172   | 169   | 169   | 168   | 162  | 157  | 155  | 152  | 148  | 144  | 140  | 85   |     |     |
| Correctly       |               |       |       |       |       |       |       |       |       |       |      |      |      |      |      |      |      |      |      |     |     |
| detected        |               |       |       |       |       |       |       |       |       |       |      |      |      |      |      |      |      |      |      |     |     |
| fricatives      |               |       |       |       |       |       |       |       |       |       |      |      |      |      |      |      |      |      |      |     | 1067 |
| Falsely         |               |       |       |       |       |       |       |       |       |       |      |      |      |      |      |      |      |      |      |     | 707  |
| detected        |               |       |       |       |       |       |       |       |       |       |      |      |      |      |      |      |      |      |      |     |     |
| fricatives      |               |       |       |       |       |       |       |       |       |       |      |      |      |      |      |      |      |      |      |     |     |

<table>
<thead>
<tr>
<th>Fricative class</th>
<th>Threshold value (“θ”)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sibilant</td>
<td>1110</td>
</tr>
<tr>
<td>Nonsibilant</td>
<td>651</td>
</tr>
<tr>
<td>Affricates</td>
<td>175</td>
</tr>
<tr>
<td>Correctly</td>
<td>1936</td>
</tr>
<tr>
<td>detected</td>
<td>1776</td>
</tr>
<tr>
<td>fricatives</td>
<td>1678</td>
</tr>
<tr>
<td>Falsely</td>
<td>12603</td>
</tr>
<tr>
<td>detected</td>
<td>2279</td>
</tr>
<tr>
<td>fricatives</td>
<td>1658</td>
</tr>
</tbody>
</table>
In arriving at a threshold decision (\( \theta \)), an overlapping criterion of 50\% is considered. Performance of the proposed algorithm at various thresholds is presented in the following Table 5.10. Column 1 gives the class of fricatives i.e., sibilant, nonsibilant, affricates and falsely detected fricatives. Column 2 is the total number of phones in the database pertaining to each class. Columns 3-12 are the number of correctly detected fricatives at various thresholds. From the above table the point to be speculated is that the thresholds are very low of the order 0.001 to 0.1. This is majorly because the work is aimed at detecting the boundary of a fricative region which is a region between two valleys of frication index. Although the threshold appears low from the number and class of falsely detected fricatives, it can be observed that this threshold is good enough to reject most of the nonfricative phones. A threshold of 0.01 is chosen for detecting the fricative regions. From the Table 5.10, although the threshold values less than 0.01 (present threshold) might look good in the scenario of number of correctly detected fricatives but for such thresholds a part of the vowel region get falsely detected as fricatives, to avoid such circumstances a threshold to 0.01 is used in this work.

Figure 5 shows the evidences used in this algorithm. Phonetically labeled TIMIT utterance “She had your dark suit in greasy wash water” is shown in Figure 5(a) and Figure 5(b) is its S-Transform. Figure 5(c) is a 1-D temporal curve obtained by computing the spectral energy below 1.5 kHz. Figure 5(d) is a 1-D temporal curve obtained by computing the spectral energy above 1.5 kHz and frication ratio (\( F_r \)) computed by the ratio of Figure 5(d) and Figure 5(c) is shown in Figure 5(e). Fricative index obtained is shown in Figure 5(f). In Figure 5(f), the fricative regions predicted by the algorithm is shown in dotted line and the ground truths obtained from the transcriptions are indicated using the solid line. The ground truth and predicted fricative regions are plotted with different amplitudes to indicate the accuracy of the approach in detecting the boundary.

5.6 Evaluation of Proposed S-Transform based Approach for Detecting Fricatives

Performance of the proposed method is evaluated using TIMIT database [129]. The information about the boundary and identity (fricative and nonfricative) of phones is obtained from the transcriptions provided in the database. During the present study [s], [z], [sh], [zh], [f], [v], [dh], [th], [ch] and [jh] are considered as the sounds under fricative broad-class and they are detected from continuous speech. The effectiveness of the proposed S-Transform based approach to detect the boundary of fricative and nonfricative regions is studied by computing the percentage correctly detected fricatives (TAR—true-acceptance rate) at various overlapping criteria in Table 5.2. Column 1 of Table 5.2 specifies the class of fricatives i.e., sibilant, nonsibilant and affricates. Columns 2-6 indicate the percentage of fricatives
Figure 5.5: (a) Speech signal of phonetically labeled TIMIT utterance “She had your dark suit in greasy wash water”, (b) S-Transform, (c) 1-D temporal curve obtained by computing the spectral energy below 1.5 kHz, (d) 1-D temporal curve obtained by computing the spectral energy above 1.5 kHz, (e) Frication ratio ($F_r$), (f) Friction index, fricative regions predicted by the algorithm is shown in dotted line and the ground truths are indicated using the solid line.
Table 5.2: Performance of the proposed S-Transform based approach at various overlapping criteria. (TAR- true-acceptance rate).

<table>
<thead>
<tr>
<th>Class of sounds</th>
<th>TAR at various overlapping criteria</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>50%</td>
</tr>
<tr>
<td>Sibilant fricatives</td>
<td>98.73</td>
</tr>
<tr>
<td>Nonsibilant fricatives</td>
<td>36.71</td>
</tr>
<tr>
<td>Affricates</td>
<td>96.57</td>
</tr>
</tbody>
</table>

One interesting observation from the above results is that the degradation in the performance of sibilant fricatives with increase in overlapping criteria is very less from the range of 50%-80%. This better efficiency of detecting the fricative boundary can be attributed to the efficiency in S-Transform in localizing high-frequency events in speech like frication onset and offset points. But a slight decrease in the performance can be observed if the overlapping criteria is further increased and this decrease can be viewed as the cumulative effect of two factors. One of them is due to averaging effects induced in fricative index due to the use of a time window in computing the energy of the frication ratio. The other is due to marking errors in phonetic labeling.
Table 5.3: Comparing the performance of S-Transform, STFT and Predictability based approaches at various overlapping criteria.

<table>
<thead>
<tr>
<th>Phone</th>
<th>Total</th>
<th>S-Transform based approach</th>
<th>STFT based approach</th>
<th>Predictability based approach</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Overlapping criteria</td>
<td>Overlapping criteria</td>
<td>Overlapping criteria</td>
<td>Overlapping criteria</td>
</tr>
<tr>
<td>s</td>
<td>606</td>
<td>606</td>
<td>605</td>
<td>601</td>
</tr>
<tr>
<td>z</td>
<td>301</td>
<td>291</td>
<td>287</td>
<td>284</td>
</tr>
<tr>
<td>zh</td>
<td>191</td>
<td>189</td>
<td>185</td>
<td>178</td>
</tr>
<tr>
<td>f</td>
<td>189</td>
<td>177</td>
<td>101</td>
<td>87</td>
</tr>
<tr>
<td>v</td>
<td>175</td>
<td>30</td>
<td>23</td>
<td>15</td>
</tr>
<tr>
<td>th</td>
<td>54</td>
<td>33</td>
<td>26</td>
<td>24</td>
</tr>
<tr>
<td>dh</td>
<td>233</td>
<td>59</td>
<td>53</td>
<td>39</td>
</tr>
<tr>
<td>ch</td>
<td>71</td>
<td>69</td>
<td>68</td>
<td>67</td>
</tr>
<tr>
<td>jh</td>
<td>104</td>
<td>100</td>
<td>99</td>
<td>98</td>
</tr>
<tr>
<td>Correctly detected fricatives</td>
<td>1936</td>
<td>1504</td>
<td>1456</td>
<td>1401</td>
</tr>
<tr>
<td>Falsely detected fricatives</td>
<td>12438</td>
<td>675</td>
<td></td>
<td>526</td>
</tr>
</tbody>
</table>

Correctly detected fricatives:
- [1936, 1504, 1456, 1401, 1318, 1162]
- [1434, 1382, 1298, 1161, 878]
- [1578, 1535, 1463, 1340, 884]

Falsely detected fricatives:
- [12438, 675, 526, 911]
The relevance of the S-Transformed space in representing the frication events at a phone level analysis is presented in Table 5.12. In Table 5.12, proposed S-Transform based approach for detecting fricative regions is compared with STFT based approach and predictability based approach. Columns 1 and 2 of Table 5.12 are the identity and total number of the fricative phones in the database. Columns 3-7, 8-12 and 13-17 are number of correctly detected fricatives in S-Transform, STFT and Predictability based approaches at various overlapping criteria. Row 13 of Table 5.12 is the total number of correctly detected fricative phones. Row 14 of Table 5.12 is the total number of falsely detected fricatives.

5.6.1 STFT based Approach for Fricative Detection

Short time spectral envelope computed using conventional short time Fourier transform with a frame size of 20 ms and frame shift of 5 ms is used for detecting the fricative regions. Fricative index described in the Section 5.5 computed from the short time spectral envelope is used for detecting the fricative regions. A threshold of 0.01 is chosen empirically, the regions of speech whose fricative index is greater than 0.01 is considered as a detected fricative region. A comparison between performance of proposed S-Transform based approach and the STFT based approach for fricative detection is presented in Table 5.12. The performance of STFT based approach using the wideband spectral envelope (i.e., with a frame size of 5 ms and a frame shift of one sample) is computed and the performance is poorer than the conventional STFT based approaches.

5.6.2 Comparing the Performances of STFT and S-Transform based Approaches

From column 3 and column 8 of Table 5.12, it can be observed that number of correctly detected fricative phones in S-Transform based approach is slightly higher than the STFT based approaches for a 50% overlapping criteria. From columns 3-7, columns 8-12 of Table 5.12, it can be observed that with an increase in the overlapping criteria the number of correctly detected fricatives drastically decreases in STFT based approach, which is not the case in S-Transform based approach. The better performance of S-Transform based evidence for detecting fricative regions can be attributed to the capability of S-Transform to localize the high-frequency events.

Although the number of falsely detected fricatives in STFT based approach appears to be less, but this less number in misclassified phones can attributed to the inability of the STFT based spectral envelope to represent weak frication in the burst regions due to its averaged spectral representation over
the time frame.

The interesting observation that can be noted in case of voiced fricatives ([z], [zh], [v] and [dlh]) i.e., from the rows 4, 6, 8, 10 of Table 5.12 is that the performance of S-Transform based approach is consistently better than STFT based approach. The better performance of S-Transform can be attributed to its capability of localizing the high-frequency events. In S-Transform representation, a spectral envelope is obtained at every sample but in STFT based representation spectral envelope is obtained for every time frame. In an S-Transform representation the spectral envelope is an averaged spectral representation over all the frequency components while STFT based spectral envelope is an averaged spectral representation over a time frame. The representation in S-Transform is chosen with progressive resolution such that high-frequency events like frication onset and offset are better time localized. In case of voiced fricative both voicing and frication present in a glottal cycle mutually exclusively in time which demands a representation that can maintain the mutually exclusive nature of voiced fricative. S-Transform can better represent the characteristics of a voiced fricative maintaining both frication and voicing in a single glottal cycle but conventional STFT gives an averaged spectral envelope for a time frame of 20 ms (more than one glottal cycle). Though STFT has inability to reflect the characteristics of a voiced fricative in spectral domain but number of voiced fricatives getting correctly detected in STFT based approach is not as low as expected at 50% overlapping criterion i.e., voicing in a voiced fricative is sustained only to a part of the voiced fricative and later part of the voiced fricative behaves similar to unvoiced fricative [117, 119]. But for computing the boundary of a voiced fricative S-Transform exhibits better capability which can seen in rows 4, 6, 8, 10 in Table 5.12 at high overlapping criteria i.e., (60%-90%). Performance of both the approaches is poor in detecting the fricative [f] which can be observed from row 7 of Table 5.12. The fricative [f] has a weak high-frequency energy the approaches that rely on capturing the frequency distribution information (S-Transform and STFT based approaches) for detecting fricatives perform poorly.

5.6.3 Comparing the Performances of Predictability and S-Transform based Approaches

The approach presented in [43] is termed as predictability based approach or LPC based approach for fricative broad class detection. Performance of S-Transform and predictability based approaches is compared in Table 5.12. Fricatives are noisy in nature i.e., Fricative sounds are less correlated signals compared to sonorant sounds. Less predictable nature of a fricative is captured in predictability based approach where S-Transform and STFT rely on capturing the frequency distribution properties of fricatives for detecting the fricatives. Though the performance of S-Transform and predictability based
Table 5.4: A phone level analysis of falsely detected fricatives in S-Transform based approach. (FAR-False-alarm rate).

<table>
<thead>
<tr>
<th>Class</th>
<th>Phones</th>
<th>Total # phones</th>
<th># Falsely detected fricatives</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bursts</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>k</td>
<td>390</td>
<td>160</td>
<td></td>
</tr>
<tr>
<td>t</td>
<td>319</td>
<td>224</td>
<td></td>
</tr>
<tr>
<td>d</td>
<td>300</td>
<td>113</td>
<td></td>
</tr>
<tr>
<td>g</td>
<td>167</td>
<td>30</td>
<td></td>
</tr>
<tr>
<td>b</td>
<td>183</td>
<td>11</td>
<td></td>
</tr>
<tr>
<td>p</td>
<td>213</td>
<td>25</td>
<td></td>
</tr>
<tr>
<td>Others</td>
<td>9568</td>
<td>112</td>
<td></td>
</tr>
<tr>
<td>Nonfricative phones</td>
<td>12438</td>
<td>675 (5.4% FAR)</td>
<td></td>
</tr>
</tbody>
</table>

approaches for detecting fricative broad class are comparable, both the approaches exploit two different properties of a fricative sound (i.e., spectral distribution and noisy nature). The complementary nature of both the approaches can be clearly observed in case of fricatives [z] and [f] i.e., row 7 and row 4 of Table 5.12. In case of the fricative phone [z] both voicing and frication are present in a single glottal cycle as shown in Figure 5.4. Voiced fricative can also be viewed as a periodic frication modulated by a glottal source. The predictability measure computed as the sum all linear predictive coefficients is used in predictability based approach [43] for detecting fricatives. Linear predictive coefficients are obtained by minimizing the mean squared error between the signal predicted by the linear combinations of past samples and the original signal. Though frication is less predictable the voicing envelope of a voiced fricative is predictable so the predictability in voiced fricative makes the predictability based approach perform poorly in detecting voiced fricatives. Similar observations can be made in case of other voiced fricatives ([zh] and [v]). In case of fricative [f] high-frequency energy is very weak so the S-Transform based approach which relies on capturing the spectral energy distribution performs poorly but predictability based approach which relies on the predictability of the signal performs better than the S-Transform based approach. From Table 5.12, it can be observed that the performance of sibilant fricatives is higher than nonsibilant fricatives which is in accordance with the earlier studies [119, 119]. In the nonsibilant class the performance of voiced sounds is even more poor and the similar trend is observed in case of sibilant fricatives.

A phone level analysis is carried out to study the phone distribution of falsely detected fricatives and the observations are tabulated in Table 5.15. Column 1 & 2 of Table 5.15 are the class and identity of falsely detected fricatives. Column 3 of Table 5.15 are the total number of phones in the database. Column 4 are the number of phones falsely detected as fricatives and the percentage(%) of falsely de-
detected fricatives. From Table 5.15, among the total number of nonfricative phones (12438) only (5.4%) of phones got detected as fricatives. Majority of falsely detected fricatives belong to the class of stop sounds and sounds associated with aspiration. Around 83 % of the falsely detected fricatives are stops, 10 % are aspirated sounds.

5.7 Combining S-Transform and Predictability based Approach for Detecting

As described in Section 5.6, both the approaches i.e., S-Transform based approach and Predictability based approach exploit two complementary properties of fricatives. Although the over all accuracy of both the approaches is comparable, the phone distribution of correctly detected fricatives in these approaches is quite complementary, which can be observed from Table 5.12. Complementary nature of both the approaches is exploited for fricative detection by combining both the approaches.

The evidences from both the approaches (S-Transform and Predictability based approaches) is combined in the manner shown below:

$$\gamma[n] = \alpha_{ST}[n] + \alpha_{LPC}[n]$$ (5.23)

Here, $\alpha_{ST}[n]$ and $\alpha_{LPC}[n]$ are the predicted fricative regions from S-Transform and Predictability based approaches. $\alpha_{ST}[n]$, $\alpha_{LPC}[n]$ are the binary contours with the value ‘1’ in the predicted fricative regions and ‘0’ in the nonfricative regions. The regions of speech where the combined evidence greater than or equal to ‘1’ is considered as the predicted fricative regions in combined approach. A fricative region detected in any one of the approaches is considered as a predicted fricative region in combined approach. Predicted fricative regions in combined approach ($\alpha_{comb}[n]$) is given by equation 5.36.

$$\alpha_{comb}[n] = \begin{cases} 1, & \text{if } \gamma[n] \geq 1 \\ 0, & \text{elsewhere} \end{cases}$$ (5.24)

Column 1 of Table 5.14 is the identity of the fricative phone. Column 2 is the total number of a particular fricative phones in the dataset. Columns 3-7 is the number of fricatives detected using combined approach at different overlapping criteria. Row 12-14 is the number of sibilant, nonsibilant and affricate phones detected using combined approach. Row 16-18 are the percentage of sibilant, nonsibilant and affricate phones detected using combined approach.
Performance of combined approach is expected to have the cumulative effect of performances in both the approaches. From Table 5.14 an overall improvement in detecting all the fricatives can be observed. In case of fricative \([z]\) i.e., Column 3 of Table 5.14 performance of combined approach is higher than Predictability based approach and comparable to S-Transform based approach. In case of fricative \([f]\) i.e., Column 6 of Table 5.14 performance of combined approach is higher than the S-Transform based approach and comparable to Predictability based approach. As the predicted fricative regions from both the approaches are combined at the decision level the best performing scenarios from both the approaches get reflected in the combined approach.

A comparative analysis of the percentage of correctly detected fricatives in S-Transform based approach, predictability and combined approaches is presented in Table 5.16. Column 1 of Table 5.16 is the class of fricatives i.e., sibilants, nonsibilants and affricates. Column 2 is the total number of fricatives in the dataset. Columns 3-7, 8-12 and 13-17 are the percentages of correctly detected fricatives in S-Transform based approach, predictability based approach and combined approach at different overlapping criteria varying from 50%-90%. From row 1 of Table 5.16, it can be observed that, in case of sibilant fricatives performance of S-Transform based approach is consistently more than the predictability based approach, the inefficiency of detecting the voiced fricatives \([z]\) in predictability based approach majorly gets reflected in the better performance of S-Transform based approach. It can be observed from row 2 of Table 5.16, that in case of nonsibilant fricatives due to their weak frequency distribution the performance of S-Transform based approach is poorer compared to the predictability measure based approach. Performance of combined approach is consistently higher than both S-Transform and predictability based approaches. The cumulative effect of higher detection accuracies for sibilant fricatives in S-Transform based approach and higher detection accuracies of nonsibilant fricatives in predictability based approach can be observed in the combined approach. A significant improvement in percentage of correctly detected fricatives at higher overlapping criteria can be observed in combined approach.

The performance of the combined approach for detecting fricative manner class is compared with the state-of-the-art manner class detector frameworks presented in [136, 37]. A GMM/HMM based manner class detector and a DNN/HMM based manner class detectors are implemented using the KALDI recipe and the results are tabulated in Table 5.17. From the results of Table 5.17, it can be noted that though the performance of both the detectors is comparable at 50% overlapping criteria but at higher overlapping criteria a drastic decrease in the performance can be observed. The percentage of falsely detected fricatives is 6% and 3.6% for GMM/HMM and DNN/HMM based detectors respectively which is lesser than that of proposed combined approach (12%). The less number of falsely detected fricatives in both the detector frameworks can be attributed to the following reason, in the proposed combined approach
Table 5.5: Performance of combined approach at various overlapping criteria.

<table>
<thead>
<tr>
<th>Phone</th>
<th>Total</th>
<th>Overlapping criteria</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>50</td>
</tr>
<tr>
<td>s</td>
<td>606</td>
<td>606</td>
</tr>
<tr>
<td>z</td>
<td>301</td>
<td>293</td>
</tr>
<tr>
<td>sh</td>
<td>191</td>
<td>191</td>
</tr>
<tr>
<td>zh</td>
<td>12</td>
<td>10</td>
</tr>
<tr>
<td>f</td>
<td>189</td>
<td>186</td>
</tr>
<tr>
<td>v</td>
<td>175</td>
<td>38</td>
</tr>
<tr>
<td>th</td>
<td>54</td>
<td>51</td>
</tr>
<tr>
<td>dh</td>
<td>233</td>
<td>101</td>
</tr>
<tr>
<td>ch</td>
<td>71</td>
<td>71</td>
</tr>
<tr>
<td>jh</td>
<td>104</td>
<td>101</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Sibilant fricatives</th>
<th>1110</th>
<th>1100</th>
<th>1096</th>
<th>1091</th>
<th>1088</th>
<th>1053</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nonsibilant fricatives</td>
<td>651</td>
<td>376</td>
<td>350</td>
<td>318</td>
<td>280</td>
<td>185</td>
</tr>
<tr>
<td>Affricates</td>
<td>175</td>
<td>172</td>
<td>172</td>
<td>171</td>
<td>169</td>
<td>160</td>
</tr>
</tbody>
</table>

majority of stop sounds got detected as fricatives, but in a detector framework stop are modeled as a separate class so the percentage of stops getting detected as fricatives reduced greatly. An alternate acoustic cue to detect the bursts onset are to be explored to further reduce the percentage of falsely detected fricatives.
Table 5.6: Comparing the performance of proposed S-Transform based approach, Predictability based approach and the combined approach. (TAR- true-acceptance rate).

<table>
<thead>
<tr>
<th>Class of Phone</th>
<th>Total</th>
<th>S-Transform based approach (TAR)</th>
<th>Predictability based approach (TAR)</th>
<th>Combined approach (TAR)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Overlapping criteria</td>
<td>Overlapping criteria</td>
<td>Overlapping criteria</td>
</tr>
<tr>
<td></td>
<td></td>
<td>50</td>
<td>60</td>
<td>70</td>
</tr>
<tr>
<td>Phone</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sibilant fricatives(%)</td>
<td>100</td>
<td>98.73</td>
<td>97.83</td>
<td>96.48</td>
</tr>
<tr>
<td>Nonsibilant fricatives (%)</td>
<td>100</td>
<td>36.71</td>
<td>31.18</td>
<td>25.34</td>
</tr>
<tr>
<td>Affricates(%)</td>
<td>100</td>
<td>96.57</td>
<td>95.42</td>
<td>94.28</td>
</tr>
</tbody>
</table>

Table 5.7: Comparing the performance of proposed combined approach with the state of the art manner class detectors. (TAR- true-acceptance rate).

<table>
<thead>
<tr>
<th>Class of Phone</th>
<th>Total</th>
<th>GMM/HMM detector (TAR)</th>
<th>DNN/HMM detector (TAR)</th>
<th>Combined approach (TAR)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Overlapping criteria</td>
<td>Overlapping criteria</td>
<td>Overlapping criteria</td>
</tr>
<tr>
<td></td>
<td></td>
<td>50</td>
<td>60</td>
<td>70</td>
</tr>
<tr>
<td>Phone</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sibilant fricatives(%)</td>
<td>100</td>
<td>97.92</td>
<td>97.65</td>
<td>96.47</td>
</tr>
<tr>
<td>Nonsibilant fricatives (%)</td>
<td>100</td>
<td>52.69</td>
<td>51.61</td>
<td>47.31</td>
</tr>
<tr>
<td>Affricates(%)</td>
<td>100</td>
<td>68.39</td>
<td>66.67</td>
<td>59.2</td>
</tr>
</tbody>
</table>
Table 5.8: A phone level analysis of falsely detected fricatives in combined approach. (FAR- false-alarm rate).

<table>
<thead>
<tr>
<th>Class</th>
<th>Phones</th>
<th>Total # phones</th>
<th># Falsely detected fricatives</th>
</tr>
</thead>
<tbody>
<tr>
<td>Burst</td>
<td>k</td>
<td>390</td>
<td>275</td>
</tr>
<tr>
<td></td>
<td>t</td>
<td>319</td>
<td>309</td>
</tr>
<tr>
<td></td>
<td>d</td>
<td>300</td>
<td>185</td>
</tr>
<tr>
<td></td>
<td>g</td>
<td>167</td>
<td>71</td>
</tr>
<tr>
<td></td>
<td>b</td>
<td>183</td>
<td>44</td>
</tr>
<tr>
<td></td>
<td>p</td>
<td>213</td>
<td>119</td>
</tr>
<tr>
<td>Others</td>
<td>9568</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nonfricative phones</td>
<td>12438</td>
<td>1234 (9.9%- FAR)</td>
<td></td>
</tr>
</tbody>
</table>

A phone level analysis is carried out to study the phone distribution of falsely detected fricatives in combined approach and the observations are tabulated in Table 5.8. Columns 1 & 2 of Table 5.8 are the class and identity of falsely detected fricatives. Column 3 of Table 5.8 are the total number of phones in the database. Column 4 are the number of phones falsely detected as fricatives. In combined approach, the fricative region associated with bursts gets detected more accurately leading an increase in the number of falsely detected fricative phones. From Table 5.8, among the total number of nonfricative phones (12438) 9.9 % of phones got falsely detected as fricatives. Majority of falsely detected fricatives belong to the class of stop sounds and sounds associated with aspiration. Around 81 % of falsely detected fricatives are stops, 11 % are aspirated sounds. The turbulence generated during burst release in stop consonants get detected as fricatives, abruptness at the boundary, presence of an associated plosion event can be used as a evidence to discriminate fricatives and stops.

5.8 Summary and Conclusions

In this study, S-Transform based time-frequency representation is explored for fricative landmark detection. The implications of S-Transform on speech signal is studied and S-Transform is customized for speech signal analysis. It is observed from the study that spectral domain cues are more prominent when they are used in S-Transformed space. The S-Transform based approach which relies on capturing the spectral energy distribution is further combined with an existing predictability based approach which relies on in the predictability in the signal. The performance of the combined approach is superior to the performances of both the approaches implemented individually. S-Transform is more efficient when the exact time location of the particular landmark is crucial. The S-Transform based landmark approach can be further extended to detection of landmarks such as bursts, voicing onset locations. The S-Transform basis is to be further analyzed and modified to generate new spaces with interesting
properties. Abruptness at the boundary of the fricatives and the presence of an associated plosion event can be used to discriminate between the fricatives and bursts.
Part II

Detection of Fricative Landmarks using Spectral Weighting: A Temporal Approach
5.9 Introduction

Part I of chapter 5 has explored S-Transform based time-frequency representation for detecting fricative regions from speech. Fricatives are characterized by two prime acoustic characteristics i.e., presence of majority of spectral concentration above 3 kHz and noisy nature. Majority of spectral domain approaches which rely on capturing the high frequency concentration of fricatives use acoustic cues such as band energy ratio, spectral centroid and dominant resonant frequency. These approaches require a time-frequency representation for detecting fricatives and their performances are influenced by the characteristics of time-frequency representation used. In part II of chapter 5, we propose a temporal cue for detecting fricative regions of speech, which would not require any time-frequency representation. The temporal cue depends on the unique spectral distribution of fricatives i.e., the majority of spectral concentration lies above 3 kHz.

5.10 Related Work

Spectro-temporal evidences computed from zero frequency filtering (ZFF) method and the time-frequency representation obtained from zero time-lifting (ZTL) method are used for detecting unvoiced fricatives from speech [119]. The strength of excitation computed from ZFF method is used as a cue for detecting unvoiced regions of speech, dominant resonant frequency (DRF) and the numerator of group delay function at zero frequency (NGD at 0 Hz) are used to discriminate fricatives from silence and unvoiced bursts. But the approach in [119] uses cues like numerator of the group delay and dominant resonant frequency which are computationally very intense. Recently, band energy ratio computed from S-Transform based time-frequency representation is explored for detecting fricatives from speech [137]. S-transform is a progressive time-frequency representation tailored to time localize the high-frequency events (onset, offset of frication events). S-transform can give a better time localization for high-frequency events. The S-transform based approach in [137] can accurately detect the duration of the fricative events. Along with the above approaches, most of the fricative detecting algorithms employ acoustic cues such as band energy ratios, spectral centroid, and dominant resonant frequency aiming to model the spectral concentration of fricatives. In doing so, all the approaches rely on time-frequency representations like STFT (Short time Fourier transform), S-Transform and ZTL-Spectrogram etc., where the detection accuracy of these approaches is influenced by the efficiency of time-frequency representation to represent the fricative events. In this work, a temporal cue is put-forth for detecting fricatives, though the proposed cue would not require any time-frequency representation, the method indirectly exploits the frequency distribution of fricatives i.e., the presence of the majority of spectral distribution in high-frequency regions.
In the proposed approach, the speech signal is scaled using a scaling function along the frequency such that \( f_s/2 \) has maximum weight. Due to the presence of the majority of spectral distribution in high-frequency regions, the fricative regions in the obtained spectrally weighted signal are more emphasized. The relative changes in the energy of speech signal and its spectrally weighted signal are quantified to detect fricative regions. In addition to the presence of high-frequency distribution, fricatives are noisy in nature. Due to the turbulence generated during the production of a fricative, the relations among the successive samples in a fricative are less compared to a sonorant. The correlations among the successive samples make the signal predictable from the past samples. A predictability measure obtained by computing the gain of inverse all-pole filter at zero frequency is used as a cue to discriminate sonorant and fricative regions of speech [43]. The proposed approach and the predictability measure based approach exploit two complementary properties of fricatives, a combination of both the approaches is put-forth for detecting fricatives from speech. The proposed approach being a temporal measure, along with the presence of frication, the duration of the frication can also be obtained accurately. Duration of the frication is used as a cue for discriminating the finer class of fricatives such as sibilant fricatives, non-sibilant fricatives, and affricates. The duration of frication in a sibilant fricative is longer than the non-sibilant fricatives and the pre-frication region in a stop [39].

### 5.11 Database

Proposed approach for detecting the fricative regions from continuous speech is evaluated using TIMIT database [129]. The labeled boundaries from TIMIT database are used during the evaluation. Fricatives considered for the present study are \([s], [z], [sh], [zh], [f], [v], [th], [dh], [ch], [jh]\). A subset of TIMIT database with 40 speakers with 20 male & 20 female, each speaker containing 10 sentences each of 2 to 3.5 sec long are considered for the study. Speech samples sampled at 16 kHz and 16 bits/sample encoding is used during the course of present study.

The class of fricatives comprises of sibilant phones \([s], [sh], [z], [zh]\) and non-sibilant phones \([f], [v], [th], [dh]\). During the production of affricates a supra-glottal closure is followed by a burst release, so affricates are closer to fricatives than stops [119]. During the study, affricates \([ch]\) and \([jh]\) are also considered in the class of fricatives. In this study, the glottal fricatives \([hl], [hv]\) are not considered for analysis as they exhibit sonorant nature [120].

### 5.12 Spectral Weighting Approach for Detecting Fricatives

If the spectrum of a speech signal is scaled by a weighing function \( k \) to obtain a spectrally weighted signal \( kX(k) \), where \( k \) is the discrete frequency. The temporal equivalent of \( kX(k) \) can be approximated
as the first difference of the time domain signal $x[n]$.

$$kX(k) \longrightarrow x'[n];$$ (5.25)

where $kX(k)$, $k^2X(k)$, $k^3X(k)$, are computed by the following equations respectively

$$x'[n] = x[n] - x[n - 1];$$ (5.26)
$$x''[n] = x'[n] - x'[n - 1];$$ (5.27)
$$x'''[n] = x''[n] - x''[n - 1]$$ (5.28)

The nature of the obtained spectrally weighted signals ($x'[n]$, $x''[n]$ and $x'''[n]$) is different for different sound units based on their frequency distributions. In case of fricatives major part of the spectral energy distribution (spectral peak) is scaled by a larger scaling factor but for the other sounds like sonorants, major frequency distribution is scaled by a smaller factor. By this scaling function as $k$ fricative regions of speech are emphasized compared to the non-fricative regions of speech. By choosing the scaling function optimally a solution can be arrived such that, the energy of spectrally weighted signal increases compared to the original signal only in the fricative regions. In this approach, three different scaling functions such as $k$, $k^2$, $k^3$ are explored. The scaling functions are chosen as the integer powers of $k$ due to their ease in implementations as differencing the time domain signal. The rest of the algorithm is described using the scaling function $k$ and the same is applicable for the scaling functions $k^2$ and $k^3$.

We hypothesize that, by scaling the spectrum optimally with a scaling function increasing along frequency, the fricative regions of speech are more emphasized. The effectiveness of the scaling function in emphasizing the fricative regions can be observed by computing a temporal measure $H$ as defined in Eq.5.29.

$$H[n] = \frac{E_x[n]}{E_{x'}[n]},$$ (5.29)

Where $E_x[n]$, $E_{x'}[n]$ are the short-term energies of the speech signal and spectrally weighted speech computed with a window size of 1 ms and one sample shift. The ratio $H[n]$ can also be computed as the ratio of $|x'[n]|$ and $|x[n]|$, but the ratio tends to be very spiky at some locations where the absolute value of $x[n]$ turns to be smaller in value. To avoid this, the energy ratio between the signals $x'[n]$ and $x[n]$ is computed using a small temporal window. For $H[n]$ to reflect the local changes in $x'[n]$ and $x[n]$, a temporal window is chosen with minimum size i.e., typically a window size of 0.25-2 ms is observed to
give similar performance.

$$E_x[n] = \sum_{k=-q}^{q} |x[k]|^2$$  
(5.30)

$$E_{x'}[n] = \sum_{k=-q}^{q} |x'[k]|^2$$  
(5.31)

where $2q + 1$ is the window size in samples corresponding to about 1 msec.

In $H[n]$, the energy of the spectrally weighted signal ($E_{x'}[n]$) is more compared to $E_x[n]$, so the ratio of these two energies is less-than “1” in fricative regions. The evidence $l[n]$ is computed as shown in equation 5.32.

$$l[n] = \begin{cases} 1, & \text{if } H[n] < 1 \\ 0, & \text{elsewhere} \end{cases}$$  
(5.32)

$$g[n] = \frac{1}{2p + 1} \sum_{k=-p}^{p} l[k]$$  
(5.33)

where $2p + 1$ is the window size in samples corresponding to about 10 msec. Due to the presence of some large high-frequency energy compared to low-frequency energy for a certain small duration in some sonorants these regions also get detected in the initial evidences ($l[n]$) but unlike fricatives, these evidences do not sustain over a certain duration. To eliminate these spurious evidences mean of the initial evidences is computed with a window of 10 msec and the regions of speech with computed mean greater than the threshold ($\theta$) is treated as a detected fricative region ($d_{frc}$). An empirical analysis is performed to calculate the value of the threshold ($\theta$).

$$d_{frc}[n] = \begin{cases} 1, & \text{if } (g[n] > \theta) \\ 0, & \text{elsewhere} \end{cases}$$  
(5.34)

In this paper $d_{frc}$ is the predicted evidence from the fricative detection algorithm. Apart from fricatives, a pre-friction region is associated with stop sounds and to eliminate most of the stop sounds, a durational constraint of 20 ms is used i.e., if the detected fricative evidence ($d_{frc}$) is less than 20 msec then it is considered as spurious evidence. In this study, an energy based criterion is employed to discriminate between speech and non-speech regions.

The proposed hypothesis can be observed in Fig.5.6. Speech signal and the corresponding spectrogram computed with a window length of 20 ms and a window shift of 10 ms. In Fig.5.6 (a, b) is the original speech signal and its spectrogram. In Fig.5.6 (c, d), (e, f) and (g, h) are the spectrally weighted speech signals and the corresponding spectrograms for the scaling functions $k, k^2, k^3$. In Fig.5.6. (a, c, e, and g), the boundaries of fricative phones are marked and the corresponding phone label is placed at the center of the fricative region. A relative increase in the amplitude of the fricative phone is noted compared to
Figure 5.6: Figure showing the effect of various scaling functions $k, k^2, k^3$ on speech signal and the corresponding spectrogram. Speech signal in the figure is taken from the TIMIT utterance with the transcription “She had your dark suit in greasy wash water”. In the figure, fricative regions are labeled to emphasize the changes due to scaling. (a,b) Speech signal and its corresponding spectrogram with a frame size of 20 ms and frame shift of 10 ms. Spectrally weighted speech signals and the corresponding spectrograms for the weighting functions $k$, $k^1$ and $k^2$ are shown in (c, d), (e, f), (g, h).
other phones for different scaling functions \( (k, k^2, k^3) \). In Fig. 5.6. (e, f, and h), the spectral energy in fricative regions getting emphasized with different scaling functions from \( k, k^2 \) and \( k^3 \) can be observed.

### 5.12.1 Analysis of Various Scaling Functions

The performance of the proposed approach is computed for various scaling functions and the results are tabulated in Table 5.9. Column 1 of Table 5.9 is the identity of the fricative phone. Column 2 of Table 5.9 is the total number of phones in the dataset. Column 3-6 of Table 5.9 indicate the number of phones predicted by the proposed approach using different scaling functions. Row 7, 8 of Table 5.9 indicate the total number of detected fricative phones and a total number of falsely detected fricative phones i.e., non-fricative phones detected as fricatives. The results in Table 5.9 are obtained using a threshold value of 0.45.

Table 5.9: Comparing the performance of various scaling functions \( (k^1, k^2 \text{ and } k^3) \) in detecting fricative regions from speech. The database comprises of 1936 fricatives and 12438 non-fricative phones.

<table>
<thead>
<tr>
<th>Phone</th>
<th>Total</th>
<th>Various scaling functions</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>( k )</td>
</tr>
<tr>
<td>s</td>
<td>606</td>
<td>606</td>
</tr>
<tr>
<td>z</td>
<td>301</td>
<td>288</td>
</tr>
<tr>
<td>sh</td>
<td>191</td>
<td>186</td>
</tr>
<tr>
<td>zh</td>
<td>12</td>
<td>10</td>
</tr>
<tr>
<td>f</td>
<td>189</td>
<td>173</td>
</tr>
<tr>
<td>v</td>
<td>175</td>
<td>45</td>
</tr>
<tr>
<td>th</td>
<td>54</td>
<td>37</td>
</tr>
<tr>
<td>dh</td>
<td>233</td>
<td>32</td>
</tr>
<tr>
<td>ch</td>
<td>71</td>
<td>69</td>
</tr>
<tr>
<td>jh</td>
<td>104</td>
<td>92</td>
</tr>
<tr>
<td>Total</td>
<td>1936</td>
<td>1538</td>
</tr>
<tr>
<td>No.of falsely detected phones</td>
<td>12438</td>
<td>430 ((3.4%))</td>
</tr>
</tbody>
</table>

The phone labels and the corresponding time stamps from the transcriptions from TIMIT are used as ground-truth to evaluate the performance of the proposed approach. From Table 5.9, it can be observed that the performance of the proposed approach increases drastically from \( k, k^2 \), but a significant improvement in the performance is not observed using the scaling function \( k^3 \). A slight increase in the number of falsely detected phones can be observed using the scaling functions \( k \) and \( k^2 \), but in case of
$k^3$, a drastic increase of falsely detected phones can be observed. When higher-order scaling functions like $k^3$ are used as scaling functions, the faint high-frequency energy is also getting emphasized due to larger scaling factor. Further, in this work the scaling function $k^2$ is used to detect the fricative regions from speech.

5.12.2 Proposed Algorithm

- Compute $x''[n]$, which is equivalent to scaling the spectrum of the signal by a scaling factor $k^2$ i.e., $k^2 X(k) \rightarrow x''[n]$.
- Compute $E_x[n], E_{x''}[n]$, which are energies of $x[n], x''[n]$ computed with a window of 1 ms.
- Compute $H[n] = \frac{E_x[n]}{E_{x''}[n]}$.
- Compute $l[n]$ as shown in Eq.5.32. $l[n]$ is ‘1’ for the points at which the $H[n]$ is less than ‘1’ and ‘0’ elsewhere.
- Compute mean of $l[n]$ with a 10 ms window as shown in Eq.5.33 to obtain $g[n]$.
- The regions where $g[n]$ greater than the threshold ($\theta$) are treated as detected evidences $d_{frc}$.
- Predicted evidences with less than 20 ms duration are treated as spurious evidences and are removed from $d_{frc}$.

Various intermediate parameters computed in proposed approach are presented in Fig.5.7. Fig.5.7.(a) is the speech signal along with the labeled phonetic transcription of TIMIT utterance with text “she had my dark suit in greasy wash water all year”. Fig.5.7.(b) is the energy of speech signal ($E_x[n]$) computed with a window of 1 ms. Fig.5.7.(c) is the energy of two time differenced speech signal ($E_{x''}[n]$), which is equivalent to the time signal whose spectrum is scaled by a scaling function $k^2$. In Fig.5.7.(c) an increase of energy in fricative regions compared to Fig.5.7.(b) can be observed. Fig.5.7.(d) is the $H[n]$ ratio computed using Eq.5.29. Fig.5.7.(e) is the proposed evidence $g[n]$ computed using Eq.5.33. Fig.5.7.(f) shows the intermediate evidences ($g[n]$), detected fricative regions ($d_{frc}$), and the groundtruth. In Fig.5.7(f) the detected fricative regions are marked with the dotted line and the groundtruth are marked with the solid line, for better visualization the groundtruth and the detected fricative regions are presented at different amplitude.

5.12.3 Arriving at a Threshold

A threshold based decision is used to detect the fricative and non-fricative regions, the region of speech where $d_{frc}$ is greater than $\theta$ is the hypothesized fricative region and the region of speech where $d_{frc}$ is
Figure 5.7: Intermediate evidences computed in the proposed approach. (a) Speech signal (Speech signal in the above figure is taken from the TIMIT utterance with the transcription “She had your dark suit in greasy wash water”), (b) Energy of speech signal ($E_x[n]$), (c) Energy of spectrally weighted speech signal using a scaling function $k^2$ ($E_{x'}[n]$), (d) $H[n]$ computed by the ratio of $E_x[n], E_{x'}[n]$, (e) Proposed fricative evidence ($g[n]$) and (f) Proposed fricative evidence ($g[n]$), detective fricative region ($d_{frc}$) shown in dotted line and the groundtruth in solid line.
less than $\theta$ is the hypothesized non-fricative region. The threshold $\theta$ is empirically using the dataset 5.11. A fricative phone with more than 50% overlap between the hypothesized fricative region and the ground truth is termed as the correctly classified fricative region. A non-fricative phone with hypothesized fricative region greater than 20 ms is considered as falsely detected fricatives. In literature, the percentage of correctly classified fricatives and falsely detected fricatives are termed as true acceptance rate and false alarm rate. Performance of the proposed approach at various thresholds is presented in Table 5.10.
Table 5.10: Performance of the proposed algorithm at various threshold values.

<table>
<thead>
<tr>
<th>Fricative class</th>
<th>Total# Phones</th>
<th>Threshold value ('θ')</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.1</td>
<td>0.15</td>
</tr>
<tr>
<td>Sibilant</td>
<td>1110</td>
<td>1109</td>
</tr>
<tr>
<td>Non-sibilants</td>
<td>651</td>
<td>466</td>
</tr>
<tr>
<td>Affricates</td>
<td>175</td>
<td>175</td>
</tr>
<tr>
<td>Correctly detected fricatives</td>
<td>1936</td>
<td>1750</td>
</tr>
<tr>
<td>Falsely detected fricatives</td>
<td>12438</td>
<td>1196</td>
</tr>
</tbody>
</table>

Column 1 of Table 5.10 gives the class of fricative phones such as sibilants, non-sibilants, and affricates. The total number of phones present in the database is presented in column 2 of Table 5.10. Columns 3-20 are the number of fricatives detected at different thresholds. Row 6, 7 of Table 5.10 is the number of correctly detected fricatives and falsely detected fricatives respectively. In this work, a threshold of 0.45 is considered for further analysis. It can be observed that for the thresholds below 0.45, though the number of correctly detected fricatives is high, the number of wrongly detected phones increases rapidly.
5.13 Evaluation of the Spectral Weighting Approach for Detecting Fricatives

TIMIT Dataset described in Section 5.11 is used for evaluating the performance of the proposed approach. The phone identity and boundary information from the manual transcriptions is used in generating ground truth for evaluating the proposed method. To study the efficiency of the proposed approach in detecting the duration of a frication region, the performance of the proposed approach at various overlapping criteria is presented in Table 5.11.

Table 5.11: Performance of the proposed approach at various overlapping criteria. (TAR- true-acceptance rate).

<table>
<thead>
<tr>
<th>Class of sounds</th>
<th>TAR at various overlapping criteria</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>50%</td>
</tr>
<tr>
<td>Sibilant fricatives (%)</td>
<td>99.09</td>
</tr>
<tr>
<td>Non-sibilant fricatives (%)</td>
<td>55.76</td>
</tr>
<tr>
<td>Affricates (%)</td>
<td>98.85</td>
</tr>
</tbody>
</table>

Different classes of fricatives such as sibilants, Non-sibilants, and affricates are presented in column 1 of Table 5.11. Performances of the proposed approach at various overlapping criteria are presented in column 2-6. From Table 5.11, it can be observed that at high overlapping criteria a degradation in the performance of non-sibilants can be observed compared to sibilants. As the approach relies on capturing the high-frequency spectral concentration, the performance is slightly poorer for the phones with very faint high-frequency spectral energy. A slight degradation in the performance of the fricatives can be observed at higher overlapping criteria at 90% and this decrease can be viewed as a combined effect of two factors i.e., the averaging effects introduced in computing the evidences and the manual marking errors in the groundtruth.

5.13.1 Comparing the Performance of the Proposed Approach with S-transform based Approach and Predictability Measure based Approach

Performance of the proposed approach is compared with the S-Transform and predictability measure based approaches and a phone level comparison is presented in Table 5.12.
Table 5.12: Comparing the performance of S-Transform, Proposed approach and Predictability based approaches at various overlapping criteria.

<table>
<thead>
<tr>
<th>Phone</th>
<th>Total</th>
<th>S-Transform based approach</th>
<th>Proposed approach</th>
<th>Predictability based approach</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Overlapping criteria</td>
<td>Overlapping criteria</td>
<td>Overlapping criteria</td>
</tr>
<tr>
<td></td>
<td></td>
<td>50  60  70  80  90</td>
<td>50  60  70  80  90</td>
<td>50  60  70  80  90</td>
</tr>
<tr>
<td>s</td>
<td>606</td>
<td>606 605 601 593 551</td>
<td>606 606 605 601 569</td>
<td>606 605 604 584 392</td>
</tr>
<tr>
<td>z</td>
<td>301</td>
<td>291 287 284 270 238</td>
<td>295 293 291 273 238</td>
<td>262 251 222 156 71</td>
</tr>
<tr>
<td>sh</td>
<td>191</td>
<td>189 185 178 166 144</td>
<td>189 189 187 176 143</td>
<td>191 191 190 182 129</td>
</tr>
<tr>
<td>zh</td>
<td>12</td>
<td>10  9  8  7  7</td>
<td>10  10  10  10  9</td>
<td>10  9  8  4  2</td>
</tr>
<tr>
<td>f</td>
<td>189</td>
<td>117 101 87 61 33</td>
<td>186 182 174 143 66</td>
<td>181 172 164 132 65</td>
</tr>
<tr>
<td>v</td>
<td>175</td>
<td>30  23  15  9  8</td>
<td>58  50  36  22  10</td>
<td>27  16  11  8  3</td>
</tr>
<tr>
<td>th</td>
<td>54</td>
<td>33  26  24  20  12</td>
<td>50  48  40  30  15</td>
<td>48  46  37  48  27</td>
</tr>
<tr>
<td>dh</td>
<td>233</td>
<td>59  39  31  21</td>
<td>69  61  41  22  10</td>
<td>86  78  65  69  63</td>
</tr>
<tr>
<td>ch</td>
<td>71</td>
<td>69  68  67  66  62</td>
<td>71  71  68  64  56</td>
<td>71  71  70  69  63</td>
</tr>
<tr>
<td>jh</td>
<td>104</td>
<td>100  99  98  95  86</td>
<td>102 101  94  83  60</td>
<td>96  96  92  88  69</td>
</tr>
</tbody>
</table>

| Correctly detected fricatives | 1936 | 1504 | 1456 | 1401 | 1318 | 1162 |
| Falsely detected fricatives   | 12438 | 675 | 680 | 911 (648) |
5.13.2 Comparing the Performance S-transform based Approach and Proposed Approach

The proposed method relies on capturing the spectral properties of fricatives i.e., the concentration of spectral energy in high-frequency regions. Performance of the proposed approach is compared with another spectral domain approach i.e., S-Transform based approach.

S-Transform based Approach: S-transform is a progressive time-frequency representation such that high-frequency events such as onsets and off-sets of frication are time-localized. Spectral envelope of S-transform is obtained at every time sample whereas for an STFT (Short time Fourier transform) representation spectral envelope is obtained at every frame. In an S-transform, spectral representation is averaged over all the frequency components while STFT based spectral representation gives a time-frequency representation averaged over time. Owing to the above properties S-transform can better represent the fricative events [137]. The method described in [137] for detecting fricatives using S-transform is termed as S-Transform based approach. S-transform based approach is implemented and the results are presented in column 3-7 of Table 5.12.

From Column 3 of Table 5.12, it can be observed that the performance of the proposed approach at 50% is better than S-transform based approach. As the approach relies on capturing the spectral energy distribution performance of the proposed approach is high for sibilants compared to non-sibilants. One interesting observation that can be made is that in case of voiced fricatives [Z], [zh], [v] and [dh] where S-transform based time-frequency representation is known to well represent the voiced fricatives, interestingly the performance of the proposed approach is better than S-transform based approach. In case of fricative [f] and other non-sibilant fricatives due to the weak high-frequency spectral energy, the non-sibilant fricatives are hard to be detected using spectral domain cues like band energy ratio, spectral centroid and dominant resonant frequency, but the proposed method relies on the presence of high-frequency spectral concentration rather than absolute value of the high-frequency spectral energy. So the proposed method can better detect the non-sibilant fricatives compared to the other spectral domain approaches. In case of non-sibilant fricatives like [dh] and [jh] only a weak frication exists for a less duration, so the performance of the proposed approach is slightly lesser at higher overlapping criteria.
5.13.3 Comparing the Performance of Proposed Approach and Predictability Measure based Methods

The proposed method relies on capturing the spectral properties of fricatives, in literature the noisy nature of fricatives is exploited for detecting fricatives using predictability measure based approach. The performance of the proposed approach is compared with predictability measure based approach.

**Predictability Measure based Approach:** The approach presented in [43] is termed as predictability measure based approach or LPC (linear prediction coefficients) based approach for detecting the fricative broadclass. The predictability measure based approach relies on capturing the noisy nature of fricatives. In the production of fricatives, the influence of vocal tract is less compared to a sonorant. The turbulence generated during the production of fricatives makes them less predictable from the past samples compared to the other sounds like sonorants. A predictability measure computed from the gain of all pole filter at ‘0’ Hz i.e., the sum of all the LPC coefficients can be used to detect fricatives. In this work, the method proposed in [43] is implemented and the results are reported in column 13-17 of Table 5.12 and the performance is compared with the proposed approach. In this work, the predictability measure based approach is implemented and to remove the spurious detections an additional durational constraint of 20 ms is employed i.e., if the detected fricative region is less than 20 ms the detected region is treated as spurious detection and considered as a non-fricative region. Owing the use of this additional durational constraint the number of falsely detected fricatives has reduced from 911 to 648.

In case of voiced fricative both voicing and frication exist mutually exclusively in a glottal cycle. It can also be considered as frication getting modulated over the glottal source. The predictability measure is obtained by computing the sum of linear prediction coefficients (LPC). LP coefficients are obtained by minimizing the mean squared error between the signal and the signal predicted by the linear combination of the past samples. In case of a voiced fricative, though the frication is less predictable the voicing envelope is predictable so the predictability measure based approach performs poorly in case of voiced fricatives such as [z], [zh], [v] and [jh]. As the predictability measure based approach does not rely on capturing the high-frequency spectral concentration, the predictability measure based approach performs better in case of non-sibilant fricatives. To study the phone distribution of falsely detected fricatives a phone level analysis is presented in Table 5.13.
Table 5.13: A phone level analysis of falsely detected fricatives using the proposed approach. (FAR-False-alarm rate).

<table>
<thead>
<tr>
<th>Class</th>
<th>Phones</th>
<th>Total # phones</th>
<th># Falsely detected as fricatives</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bursts</td>
<td>k</td>
<td>390</td>
<td>214</td>
</tr>
<tr>
<td></td>
<td>t</td>
<td>319</td>
<td>256</td>
</tr>
<tr>
<td></td>
<td>d</td>
<td>300</td>
<td>97</td>
</tr>
<tr>
<td></td>
<td>g</td>
<td>167</td>
<td>39</td>
</tr>
<tr>
<td></td>
<td>b</td>
<td>183</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>p</td>
<td>213</td>
<td>42</td>
</tr>
<tr>
<td>Others</td>
<td></td>
<td>9568</td>
<td>25</td>
</tr>
<tr>
<td>Non-fricative phones</td>
<td>12438</td>
<td>680 (5.4%- FAR)</td>
<td></td>
</tr>
</tbody>
</table>

Class and identity of falsely detected phones is presented in Table 5.13. Column 3, 4 are the total number of phones in the database and number of falsely detected phones. From Table 5.13, it can be noted that of the non-fricative phones 5.4% got falsely detected as fricatives. Majority of the falsely detected fricatives belongs to the class of the stop sounds. Around 96.3% of stops and 3.6% aspirated sounds got falsely detected fricatives.

5.14 Combining the Proposed and Predictability based Approaches

Proposed approach and predictability measure based approaches capture the two different properties of fricatives. They exhibit complementary nature in their phone distributions. Due to the presence of predictability in voicing envelope, voiced fricatives ([z], [zh], [ɣ]) are less detected using predictability measure based approach compared to the proposed approach. The performance of the proposed approach in detecting non-sibilant phones with weak frication ([dh], [jh], [th]) is less compared to predictability measure based approach. To exploit the complementary nature of both the approaches i.e., S-Transform based approach and predictability measure based approach, a combination of both the approaches is studied for detecting fricative regions from speech. The predicted fricative regions from both the approaches i.e., proposed approach ($\alpha_{d_{frc}}$) and predictability based approach ($\alpha_{LPC}$) are combined at decision level.

$$\gamma[n] = \alpha_{d_{frc}}[n] + \alpha_{LPC}[n]$$ (5.35)
Table 5.14: Performance of combined approach at various overlapping criteria.

<table>
<thead>
<tr>
<th>Phone</th>
<th>Total</th>
<th>Overlapping criteria</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>50</td>
</tr>
<tr>
<td>s</td>
<td>606</td>
<td>606</td>
</tr>
<tr>
<td>z</td>
<td>301</td>
<td>295</td>
</tr>
<tr>
<td>sh</td>
<td>191</td>
<td>191</td>
</tr>
<tr>
<td>zh</td>
<td>12</td>
<td>10</td>
</tr>
<tr>
<td>f</td>
<td>189</td>
<td>188</td>
</tr>
<tr>
<td>v</td>
<td>175</td>
<td>62</td>
</tr>
<tr>
<td>th</td>
<td>54</td>
<td>51</td>
</tr>
<tr>
<td>dh</td>
<td>233</td>
<td>99</td>
</tr>
<tr>
<td>ch</td>
<td>71</td>
<td>71</td>
</tr>
<tr>
<td>jh</td>
<td>104</td>
<td>102</td>
</tr>
<tr>
<td>Sibilant fricatives</td>
<td>1110</td>
<td>1102</td>
</tr>
<tr>
<td>Non-sibilant fricatives</td>
<td>651</td>
<td>400</td>
</tr>
<tr>
<td>Affricates</td>
<td>175</td>
<td>173</td>
</tr>
</tbody>
</table>

Here $\alpha_{dfr}[n]$, $\alpha_{LPC}[n]$ binary contours with fricative region marked as ‘1’ and non-fricative region marked as ‘0’. In the combined approach, the region of speech with the combined evidence $\gamma[n]$ greater than ‘0’ is considered as the hypothesized fricative region and the region of speech with $\gamma[n]$ is zero is the hypothesized non-fricative region. The predicted fricative region in combined approach is given by

$$
\alpha_{comb}[n] = \begin{cases} 
1, & \text{if } \gamma[n] > 0 \\
0, & \text{elsewhere} 
\end{cases}
$$

(5.36)

The performance of the combined approach is presented in Table 5.14. The performance of the combined approach is superior to both the approaches. The better performances of both the approaches get reflected in the performance of the combined approach as the evidences from both the approaches are combined at decision level. In case of voiced fricative, the better performance exhibited by the proposed approach gets reflected in the combined approach and in case of weak non-sibilants, the performance of predictability measure gets reflected in the combined method.
Table 5.15: A phone level analysis of falsely detected fricatives in combined approach. (FAR- False-alarm rate).

<table>
<thead>
<tr>
<th>Class</th>
<th>Phones</th>
<th>Total # phones</th>
<th># Falsely detected as fricatives</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bursts</td>
<td>k</td>
<td>390</td>
<td>258</td>
</tr>
<tr>
<td></td>
<td>t</td>
<td>319</td>
<td>302</td>
</tr>
<tr>
<td></td>
<td>d</td>
<td>300</td>
<td>188</td>
</tr>
<tr>
<td></td>
<td>g</td>
<td>167</td>
<td>63</td>
</tr>
<tr>
<td></td>
<td>b</td>
<td>183</td>
<td>47</td>
</tr>
<tr>
<td></td>
<td>p</td>
<td>213</td>
<td>109</td>
</tr>
<tr>
<td>Others</td>
<td></td>
<td>9568</td>
<td>30</td>
</tr>
<tr>
<td>Non-fricative phones</td>
<td>12438</td>
<td>997 (8%- FAR)</td>
<td></td>
</tr>
</tbody>
</table>
Table 5.16: Comparing the performance of proposed approach, predictability based approach and the combined approach. (TAR- true-acceptance rate).

<table>
<thead>
<tr>
<th>Class of Phone</th>
<th>Total</th>
<th>Proposed approach (TAR)</th>
<th>Predictability based approach (TAR)</th>
<th>Combined approach (TAR)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Overlapping criteria</td>
<td>Overlapping criteria</td>
<td>Overlapping criteria</td>
</tr>
<tr>
<td></td>
<td></td>
<td>50</td>
<td>60</td>
<td>70</td>
</tr>
<tr>
<td>Sibilant fricatives (%)</td>
<td>100</td>
<td>99.09</td>
<td>98.91</td>
<td>98.46</td>
</tr>
<tr>
<td>Non-sibilant fricatives (%)</td>
<td>100</td>
<td>55.76</td>
<td>52.38</td>
<td>44.70</td>
</tr>
<tr>
<td>Affricates (%)</td>
<td>100</td>
<td>98.85</td>
<td>98.28</td>
<td>92.57</td>
</tr>
</tbody>
</table>

Table 5.17: Comparing the performance of proposed combined approach with the state of the art manner class detectors. (TAR- true-acceptance rate).

<table>
<thead>
<tr>
<th>Class of Phone</th>
<th>Total</th>
<th>GMM/HMM detector (TAR)</th>
<th>DNN/HMM detector (TAR)</th>
<th>Combined approach (TAR)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Overlapping criteria</td>
<td>Overlapping criteria</td>
<td>Overlapping criteria</td>
</tr>
<tr>
<td></td>
<td></td>
<td>50</td>
<td>60</td>
<td>70</td>
</tr>
<tr>
<td>Sibilant fricatives (%)</td>
<td>100</td>
<td>97.92</td>
<td>97.65</td>
<td>96.47</td>
</tr>
<tr>
<td>Non-sibilant fricatives (%)</td>
<td>100</td>
<td>52.69</td>
<td>51.61</td>
<td>47.31</td>
</tr>
<tr>
<td>Affricates (%)</td>
<td>100</td>
<td>68.39</td>
<td>66.67</td>
<td>59.2</td>
</tr>
</tbody>
</table>
The phone distribution of the falsely detected fricatives is presented in Table 5.15. From Table 5.15, due to the decision level combination of both the approaches, a slight increase in the falsely detected fricatives can be observed. As the combined approach better represents regions of frication, the pre-frication regions in stops getting detected as fricatives slightly increases. In the combined approach, of the total Non-fricative phones 8% of non-fricative phones got falsely detected as fricatives. Around 97% of the non-fricative phones detected are stops and remaining 3% are the aspirated sounds.

The performance of the combined approach is compared with the proposed approach and the predictability based approach and the results are tabulated in Table 5.16. Column 1 of Table 5.16 is the class of fricatives, column 2 of Table 5.16 is the total performance and columns 3-7, 8-12 and 13-17 are the performances of the proposed approach, predictability measure based approach, and the combined approach respectively. It can be observed that the combined approach performs superior to both the individual approaches. Due to the complementary nature of both the approaches an increase in the performance can be observed when the individual approaches are combined. In this work, the performance of the combined approach for detecting the fricative broad class is compared with the state of the art fricative broad class detectors and the comparison is presented in Table 5.17. In this work, a GMM-HMM and HMM-DNN based broad class detector frameworks employed in [37, 136] are compared with the combined fricative broadclass detection approach. From Table 5.17, it can be observed that the performance of the combined approach is comparable to the state of the art fricative broadclass detectors at 50% overlapping criteria but at higher overlapping criteria the performance of the combined approach is superior to the HMM-GMM, HMM-DNN based approaches. But the percentage of falsely detected fricatives is slightly lesser for HMM-GMM, HMM-DNN based approaches i.e., 6%, 3.6% compared to the 8% of falsely detected fricatives in the combined approach. The less number of falsely detected fricatives can be attributed to the following reason as the HMM-GMM, HMM-DNN based detectors model the stops as a separate class the number of stops getting falsely detected as fricatives is less.

5.15 Summary and Conclusions

Most of the existing spectral domain approaches rely on a time-frequency representation like S-transform, STFT for detecting fricatives and the performance depends on the characteristics of the time-frequency representation. In this chapter, a temporal measure is presented which does not require any time-frequency representation for detecting fricative landmarks. The proposed approach relies on exploiting the unique frequency distribution of fricatives. A predictability measure based approach relies on exploiting the noisy nature of fricatives. As both the approaches rely on two characteristics of fricatives, a combination of both the approaches is put-forth for detecting the fricatives. Apart from
detecting the presence of fricatives, the proposed approach can accurately detect the duration of fricative regions.
Part III

Using Fricative Landmarks for Improving the Performance of Speech Recognition Systems
5.16 Introduction

Part I and II of this chapters have described S-Transform and spectral weighting based approaches for detecting fricative regions from speech. In this part, we use the detected fricative regions for improving the performance of speech recognition systems. The detected fricative evidences are combined with the conventional features, and the combined features are used for developing speech recognition systems. In this chapter monolingual and multilingual ASR systems have been developed. TIMIT and WSJ corpus are used for developing mono-lingual ASR systems and the multilingual ASR systems are developed using multilingual corpus described in chapter 3.

5.17 Exploring the Weighted Spectral Evidence for Improving the Performance of Speech Recognition System

Significance of the proposed approach is studied with relevance to speech recognition. Speech recognition systems are developed using feature-space maximum log-likelihood transform features (fMLLR) [138]. The $g[n]$ from equation 5.33 is averaged over the frame duration and appended with fMLLR features. During the study, $g[n]$ is computed for three different scaling functions $k,k^2$ and $k^3$ and the corresponding evidences obtained deve$_1$, deve$_2$ and deve$_3$ are appended with fMLLR features.

5.17.0.0.1 fMLLR Features

The 13-dimensional Mel-frequency coefficients (MFCC) are spliced over time with a context window of 9 frames ($\pm 4$) giving a vector of 117 dimensions. The resulting 117-dimensional vector is then de-correlated and its dimensionality is reduced to a 40-dimensional vector using linear discriminate analysis (LDA). The resulting 40-dimensional data is de-correlated using maximum log-likelihood transform and the speaker normalization is done using feature-space maximum log-likelihood transform(fMLLR), this is also known as CMLLR (constrained MLLR). The resulting fMLLR feature is of 40-dimensional while training an end-to-end speech recognition system the fMLLR is used with 40-dimensions and while training an HMM/DNN based speech recognition system the fMLLR feature is spliced in time using a context window of 11 ($\pm 11$) making a feature dimension of 440. Two types of speech recognition systems i.e., HMM/DNN based speech recognition system and end-to-end speech recognition with RNN-CTC based training are used. The details of both the systems are described briefly.
5.17.0.0.2 HMM-DNN based Speech Recognition System  An HMM/DNN based speech recognition system with a deep network of 4 hidden layers depth and each hidden layer comprising of 1024 units with \texttt{tanh} activation functions is used. The network is initially pre-trained with a greedy layer-by-layer approach and the pre-trained model is then discriminatively finetuned. A learning rate of 0.008 is used and the learning rate is reduced by a factor of 0.5 if there is a decrease in validation accuracy at certain epoch and the fine-tuning is stopped when the increment in the validation accuracy is less than 0.05 between two successive epochs. KALDI-PDNN toolkit [139] is used to implement the HMM-DNN based speech recognition systems.

5.17.0.0.3 RNN-CTC based Speech Recognition System  An RNN-CTC based end-to-end speech recognition system with a deep bidirectional long short-term memory networks (Bi-LSTMs) architecture is trained with connectionist temporal classification objective function. A learning rate of 0.0001 is used with a batch size of 10 sequences. The learning rate is reduced by a factor of 0.5 when there is a decrease in the validation accuracy. Training is progressed as long as a minimum increase of 0.05 is observed in the validation accuracy. EESEN toolkit [30] is used to implement the RNN-CTC based speech recognition systems.

Table 5.18: Performance of speech recognition systems developed by appending the proposed features \(\text{deve}_1, \text{deve}_2, \text{deve}_3\) with \textit{fMLLR} features. In the Table \(l_2, l_3, l_4\) corresponds to depth of hidden layers in RNN.

\[
\begin{array}{|c|c|c|c|}
\hline
& \text{PER\%} & \text{fMLLR features} & \text{HMM-DNN} & \text{RNN-CTC} \\
\hline
& & & l_2 & l_3 & l_4 \\
\hline
\text{baseline} & 20.2 & 20.96 & 19.89 & 19.53 \\
\text{ideal detector} & 18.90 & 19.09 & 17.92 & 17.33 \\
\hline
\text{deve}_1 & 20.1 & 21.11 & 19.86 & 19.36 \\
\text{deve}_2 & 20.1 & 20.29 & 19.99 & 19.17 \\
\text{deve}_3 & 20.1 & 20.14 & 19.68 & 18.83 \\
\text{deve}_1+\text{deve}_2 & 20.0 & 20.26 & 19.49 & 18.82 \\
\text{deve}_1+\text{deve}_2+\text{deve}_3 & 19.6 & 19.93 & 18.92 & 18.72 \\
\hline
\end{array}
\]

Performance of various speech recognition systems developed by appending the proposed evidences \textit{deve}_1, \textit{deve}_2 and \textit{deve}_3 is presented in Table 5.18. The fricative evidences obtained from the groundtruth labels of TIMIT dataset are treated as the evidences from an ideal fricative detector, the performance of ASR system developed by appending these features with fMLLR features is presented in row 2 of Table 5.18. Performances of the speech recognition system developed by fMLLR features is considered as a baseline system. The proposed evidence appended with the fMLLR features and the performances
are presented in column 1. Column 2 is the performances of HMM-DNN based speech recognition system. Column 3-5 are the performances of RNN-CTC based speech recognition systems developed with fMLLR features at various depths. The architecture of RNN-CTC based speech recognition system is reported in Table 5.18, has Bi-LSTMs of different depths i.e., 2, 3, 4 termed as \( l_2 \), \( l_3 \) and \( l_4 \) and each layer comprising of 320 units.

Table 5.19: Performance of speech recognition systems developed by appending the proposed features \((deve_1, deve_2, deve_3)\) with fMLLR using WSJ corpus.

<table>
<thead>
<tr>
<th></th>
<th>Baseline ( (fMLLR) )</th>
<th>( fMLLR + deve_1 + deve_2 + deve_3 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>WSJ corpus (14 hrs)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RNN-CTC+lexicon+trigram</td>
<td>9.40</td>
<td>9.13</td>
</tr>
<tr>
<td>WSJ corpus (80 hrs)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RNN-CTC+lexicon+trigram</td>
<td>7.60</td>
<td>6.98</td>
</tr>
<tr>
<td>HMM-DNN++lexicon+trigram</td>
<td>4.48</td>
<td>4.43</td>
</tr>
</tbody>
</table>

From Table 5.18, it can be observed that by appending the proposed evidences with the baseline feature representation an improvement in the performance of ASR system can be observed. Due to the additional information from the proposed evidence an improvement in the performance of speech recognition system can be observed. It can also be noted that the improvement is uniform across the speech recognition systems, viz., HMM-DNN based systems and RNN-CTC based end-to-end recognition systems. As the study majorly aims at detecting the fricative landmarks, in this work we have used only \( deve_2 \) evidence for detecting the fricative landmarks. But for an ASR system, as the model has access to other information from the feature representation along with \( deve_2 \), the evidences \( deve_1 \) and \( deve_3 \) are also appended. From the Table 5.18 as the depth of the Bi-LSTM network is increased, an improvement in the performance can be observed. In an HMM-DNN based speech recognition system, using the proposed feature appending i.e., \((deve_1 + deve_2 + deve_3 + fMLLR)\) an absolute improvement of 0.8% in PER (phone error rate) can be observed. In an end-to-end network, the proposed feature appending i.e., \((deve_1 + deve_2 + deve_3 + fMLLR)\) has shown an absolute improvement of 0.6%. Performance of the proposed feature appending has been studied in case of large vocabulary continuous speech recognition (LVCSR). The best performing architecture from Table 5.18, i.e., 4 hidden layer depth and each layer comprising of 320 units are used for further experiments. RNN-CTC based speech recognition systems using \( fMLLR \) features is considered as a baseline system. In addition to RNN-CTC based speech recognition system with lexicon and trigram language model, we have also developed speech recognition systems in multiple scenarios i.e., using a dataset with 14, 80 hours, and without a language model. The performance of these systems are presented in 5.19. From Table 5.19 it can be observed that the proposed
feature appending has improved the performance of ASR systems when the datasize is small. Significant improvement in the performances has not been observed using the proposed appended features in the context of larger sized datasets (here WSJ dataset comprising 80 hours of data).

5.18 Exploring the Weighted Spectral Evidence for Improving the Performance of Multilingual ASR

The fricative evidences described in the above section are combined with spectral features and the combined features are used for developing multilingual ASR. The multilingual joint acoustic model described in Chapter 3 and the performances of these ASR systems are tabulated in Table 5.20. Column 2 of Table 5.20 is the performance of multilingual ASR with a joint acoustic model trained using 39-dimensional spectral features. Column 3, 4, 5 are the performances of ASR systems developed by combining the spectral features with frictive evidences $deve_1$, $deve_2$ and $deve_3$ as described in the above section. Column 6 is the performance obtained by combining all the three evidences($deve_1$, $deve_2$ and $deve_3$) along with spectral features. From Table 5.20, it can be noted that a significant improvement in WER is not observed. This can be attributed to the reason that the database is sufficiently large enough and the additional feature has not improved the performance as in the case of TIMIT.

Table 5.20: Improving the performance of multilingual ASR using the detected fricative evidences.

<table>
<thead>
<tr>
<th>Test set</th>
<th>Telugu</th>
<th>Tamil</th>
<th>Gujarati</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>20.61</td>
<td>20.16</td>
<td>14.19</td>
</tr>
<tr>
<td>$deve_1$</td>
<td>21.12</td>
<td>20.16</td>
<td>14.33</td>
</tr>
<tr>
<td>$deve_2$</td>
<td>20.88</td>
<td>20.31</td>
<td>14.44</td>
</tr>
<tr>
<td>$deve_3$</td>
<td>20.85</td>
<td>20.36</td>
<td>14.17</td>
</tr>
<tr>
<td>$deve_123$</td>
<td>20.52</td>
<td>20.18</td>
<td>14.17</td>
</tr>
</tbody>
</table>
5.19 Summary and Conclusions

The work is aimed at detecting fricative landmarks in speech. In part 1 of this chapter, we have used S-Transform based time-frequency representation for detecting fricatives. In part 2 of this chapter, a temporal measure which does not require any time-frequency representation for detecting fricative landmarks has been explored. Both approaches have worked as a good fricative detector and have performed better than the state-of-the-art fricative detectors. Apart from detecting the presence of fricatives, both the approaches can accurately detect the duration of fricative regions. The information in the proposed evidence has to be studied with relevance to a speech recognition system. Due to the proposed feature appending an absolute improvement of 0.8, 0.6 in PER is obtained on the TIMIT dataset using HMM-DNN, RNN-CTC based speech recognition systems respectively. The proposed feature appending has consistently performed better than the baseline across the datasets i.e., TIMIT, WSJ. In the case of multilingual ASR, the detected fricative evidence used in combination with spectral features has not shown significant reductions in WER.
Chapter 6

Using Distinctive Features for Improving Multilingual ASR

6.1 Introduction

In the previous chapter, fricative regions from speech have been detected using S-transform and spectral weighting based approach. The detected fricative landmarks are used as features for improving the performance of a speech recognition system. In this chapter, we explore the use of various distinctive features for improving the performance of ASR systems. Distinctive features are detected by a neural network based approach. We formulate the detection of distinctive features as a multi-label classification task. The distinctive features from the neural network are fused with spectral features and the combined features are used for developing ASR system.

6.2 Related work

Incorporating cues from the speech production mechanism into a speech recognition system is one of the long-standing questions. There have been attempts to use the cues from the speech production mechanism in a statistical speech recognition system. Studies intended in this direction can be majorly categorized to two different approaches:

- Characterizing speech signal in terms of various production events such as distinctive features or landmarks and using them for discriminating words in a lexical access based speech recognition system.

- Using the cues from the production mechanism in tandem with the conventional features to train an ASR system.

Lexical access system in [3], describes the speech signal as a sequence of segments, where each segment is described by the set of binary distinctive features. Features that can specify phonemic contrast between words of a language are termed as distinctive features. In these systems, each word is represented by
a sequence of feature bundles, where a change in a distinctive feature generates a new word in the lexicon. In [5], distinctive features from speech signal are computed using landmarks as anchor points. Landmarks are the instants of time at which the distinctive features are salient. Abrupt landmarks are computed by detecting energy abruptness in five frequency bands with two temporal resolutions [5]. Acoustic correlates of phonetic features are extracted reliably in [140, 141, 142, 143] etc. Though the landmark systems performed well in detecting and characterizing the speech signal in terms of phonetic features, as these systems are primarily rule-based systems where the performance of ASR suffered due to a large number of insertions, deletion and substitutions [144]. A hierarchically structured point process based representation has been explored to probabilistically integrate the distinctive features to detect the phonological sequence [123] and it has performed equally well on a digit classification task.

Alternatively, use of phonetic features has improved the performance of ASR system. The burst onset landmarks detected from speech signal are used along with conventional spectral features through an early fusion of features and the appended features have shown an improvement in speech recognition systems [45]. Multiple MLPs (multi-layer perceptrons) have been trained to detect different articulatory features and the detected features are used for developing robust speech recognition systems [145, 146]. The trajectories of vocal tract constriction variables along used along with spectral features have lead to a better performing ASR [147, 148, 44].

Most of the above approaches use rule-based systems with a unique algorithm for different articulatory features, gaining expertise on all the algorithms and fine-tuning the algorithms for the task of LVCSR is a tedious task. Most of the approaches have studied the use of distinctive features on typical tasks using smaller sized datasets. The motive of this work is to study the relevance of using the distinctive features in an end-to-end LVCSR system. Though the studies in [146, 147] have used MLPs they have trained separate networks for different features such as place of articulation, manner of articulation and vowel height etc. But while training different networks only a subset of data will be used to train any of the networks. We explore a single framework for detecting various phonetic features. This requires a system or a network that can transduce the acoustic sequence to the sequence of phonetic features. Acoustic vectors are mapped to a distinctive feature vector to indicate the presence or absence of distinctive features. The task of mapping the sequence of acoustic vectors to a distinctive vector sequence is treated as a multi-label classification task. The detected distinctive feature vector is used in tandem with the spectral features and tandem features are explored for speech recognition.

6.3 Proposed Distinctive Feature Network

During the study, the distinctive network is trained to transduce the sequence of acoustic vectors to a sequence of distinctive features. The distinctive features considered in the study are presented
in Table 6.1. The distinctive feature vector used in this study comprises of a 28-dimensional binary vector 6.1. Neural networks are explored to learn the mapping between the acoustic vector to distinctive feature vector. Mapping the sequence of acoustic vectors to distinctive feature vectors is considered as multi-label classification. The logistic units are used as the output layer of a distinctive feature network, and the network is trained using binary cross entropy objective function.

In this study, HMM-GMM based ASR is trained using WSJ corpus, and the phone level alignment between the acoustic sequence and the corresponding phone label is obtained. The obtained phone labels are mapped to the corresponding distinctive feature vectors as mentioned in the table 6.1 and this mapping is taken from festival toolkit [149]. During the study, deep neural network and RNNs architectures have been employed as distinctive feature networks. A DNN with four hidden layers, each of 1024 Relu units, and a Bi-directional LSTM with 2 hidden layers each layer comprising of 250 units have been used. Logistic units are used as the output layer in both the networks. The network is optimized using ADAM optimizer [98] with a learning rate of 0.0001. The learning rate is halved whenever a decrease in the validation accuracy is observed. Training is stopped upon observing an increase in the validation accuracy over three successive epochs. The distinctive feature vectors predicted by the distinctive network are appended with the spectral features and the tandem features are explored for developing speech recognition.

The performance of the distinctive feature network in detecting the corresponding distinctive feature is measured in terms of frame error rate and the results are reported in Table.6.2. Column 1 is the distinctive feature network used and column 2 is the mean frame error rate. During the study, training data of two different sizes is used for training the distinctive feature network i.e., using 30 utterances per speaker and using the full dataset. The performance of both the distinctive feature networks networks is presented in Table.6.2.
Table 6.1: Distinctive feature groups used in the study.

<table>
<thead>
<tr>
<th>Feature group</th>
<th>Classes</th>
<th># bits</th>
</tr>
</thead>
<tbody>
<tr>
<td>vowel or consonant</td>
<td>± vowel</td>
<td>1</td>
</tr>
<tr>
<td>vowel length</td>
<td>short, long, diplhong, schwa</td>
<td>4</td>
</tr>
<tr>
<td>vowel height</td>
<td>high, mid, low</td>
<td>3</td>
</tr>
<tr>
<td>vowel frontedness</td>
<td>front, mid, back</td>
<td>3</td>
</tr>
<tr>
<td>vowel lip rounding</td>
<td>+,-</td>
<td>2</td>
</tr>
<tr>
<td>Manner of articulation</td>
<td>stop, fricative, affricate, nasal, lateral, approximant</td>
<td>6</td>
</tr>
<tr>
<td>Place of articulation</td>
<td>labial, alveolar, palatal, labio-dental, dental, velar, glottal</td>
<td>7</td>
</tr>
<tr>
<td>consonant voicing</td>
<td>+,-</td>
<td>2</td>
</tr>
</tbody>
</table>

Table 6.2: Performance of distinctive feature network reported in terms of mean frame error rate.

<table>
<thead>
<tr>
<th>Distinctive network</th>
<th>Frame error rate(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dist-DNN-30utt</td>
<td>81.75</td>
</tr>
<tr>
<td>Dist-DNN-fullutt</td>
<td>82.5</td>
</tr>
<tr>
<td>Dist-RNN-30utt</td>
<td>84.37</td>
</tr>
<tr>
<td>Dist-RNN-fullutt</td>
<td>84.46</td>
</tr>
</tbody>
</table>

6.4 Using Distinctive Features in Developing Monolingual ASR

RNN-CTC based acoustic model has been used for developing speech recognition systems. A 4 layer deep bi-directional LSTM network with each layer comprising of 320 units is trained using CTC objective function. The network is trained to predict the character sequences from the acoustic sequences. A character based lexicon and NIST language model are used in the study. Spectral features used in the study are mean and variance normalized on per utterance basis and no speaker normalization has been done. During the study, the distinctive feature network is trained to predict the distinctive features from the spectral features. The predicted distinctive feature is appended with the corresponding spectral features and the tandem features are used for training ASR. The blockdiagram of the proposed distinctive feature based approach is presented in Figure 6.1. While training the ASR the weights of the distinctive
feature network are not updated (i.e., weights are frozen) so that no additional parameters are added to the network. The performance of ASR trained in the proposed distinctive feature based approach is presented in Table 6.3.

![Block diagram](image)

**Figure 6.1:** Block diagram describing the proposed distinctive feature based approach.

**Table 6.3:** Performance of ASR trained using the proposed distinctive features.

<table>
<thead>
<tr>
<th>Speech recognition system</th>
<th>WER</th>
</tr>
</thead>
<tbody>
<tr>
<td>baseline</td>
<td>10.17</td>
</tr>
<tr>
<td>Dist-DNN-30utt</td>
<td>9.87</td>
</tr>
<tr>
<td>Dist-DNN-fullutt</td>
<td>9.67</td>
</tr>
<tr>
<td>Dist-RNN-30utt</td>
<td>9.62</td>
</tr>
<tr>
<td>Dist-RNN-fullutt</td>
<td>9.50</td>
</tr>
</tbody>
</table>

Column 1 of Table 6.3 gives the features used for training the ASR system and column 2 of Table 6.3 is the corresponding word error rate (WER) obtained. Row 2 of Table 6.3 is the performance of baseline ASR system trained using spectral features with mean and variance normalized based on per utterance basis. The distinctive feature network is trained using DNN and LSTM networks. To study the influence of these networks on the size of data a subset of 30 utterances from the training data is used to train the distinctive feature network. The Distinctive feature networks trained using the subset of training data is termed as Dist-DNN-30utt, Dist-RNN-30utt and the ones trained with full data are termed as
Dist-DNN-fullutt, Dist-RNN-fullutt. Row 3, 4 of Table 6.3 are the performances of ASR trained using the distinctive features from DNN trained with 30 utterances per speaker and full training dataset. Row 5, 6 of Table 6.3 are the performances of ASR trained using the distinctive features from LSTM network trained with 30 utterances per speaker and full training dataset.

Apart from distinctive features, the use of information about the phone identity is used along with the spectral features and the performance is presented in Table 6.3. In this study, DNN with an input and four hidden layers and each layer comprising of 512 Relu units and an output softmax layer, and a 3 layer bidirectional-LSTM network with 100 nodes per layer and a softmax output layer are trained for phone classification.

### 6.5 Using Distinctive Features in Multilingual ASR

We have explored the use of distinctive features for developing multilingual ASR. The multilingual ASR systems described in chapter 3 has been used. Column 2 of Table 6.4 is the performance of multilingual ASR system using joint acoustic model and spectral features. Column 3, 4 and 5, 6 of Table 6.4 are the performances of multilingual ASR systems trained by fusing the spectral features from the distinctive feature networks. From the Table 6.4, it can be observed that the use of distinctive features along with the conventional features has not improved the performances significantly.

<table>
<thead>
<tr>
<th>Test set</th>
<th>Telugu</th>
<th>Tamil</th>
<th>Gujarati</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>20.61</td>
<td>20.16</td>
<td>14.19</td>
</tr>
<tr>
<td>Dist-DNN-30utt</td>
<td>20.89</td>
<td>20.51</td>
<td>14.11</td>
</tr>
<tr>
<td>Dist-DNN-fullutt</td>
<td>20.87</td>
<td>20.36</td>
<td>13.91</td>
</tr>
<tr>
<td>Dist-RNN-30utt</td>
<td>20.89</td>
<td>20.51</td>
<td>14.24</td>
</tr>
<tr>
<td>Dist-RNN-fullutt</td>
<td>20.54</td>
<td>20.22</td>
<td>13.98</td>
</tr>
</tbody>
</table>

### 6.6 Summary and Conclusions

In this chapter, we have explored the use of distinctive features for improving the performance of ASR systems. The task of detecting distinctive features from speech signal is considered as a multi-label classification. Deep neural networks have been employed to learn the task of converting the acoustic features to the corresponding distinctive features. Use of the distinctive features shown a
slight improvement on WSJ corpus. When the distinctive features have been employed as features to multilingual joint acoustic model a significant reduction in WER is not observed.
Chapter 7

Speaker Normalization for Low Resourced Scenarios

7.1 Introduction

As mentioned in Chapter 2, resources for developing ASR systems can be categorized into three types, i.e., transcribed data, lexical resources and the meta-information about the data. Meta-level information about the data refers to the information about the data environments, speaker ID, type of microphone type, distance of the microphone etc. For developing ASR systems in low resourced scenarios, data scarcity can be reduced by pooling data from different sources where the speaker labels might not be present. In this chapter, we explore a speaker normalization approach which can handle the low resourced scenarios. We have initially explored the proposed speaker normalization for a monolingual system, and later we have extended this work for multilingual ASR systems. The Speaker-normalization approach proposed in this work can be operated efficiently even when a single utterance from the unseen speaker is available. Speaker normalization is employed to reduce the performance drop in ASR due to speaker variabilities. Traditionally speaker normalization is performed by estimating per-speaker linear transform over the input data, and such transforms would be efficient with sufficient data. We hypothesize that by suitably providing information about the speaker’s identity while training an end-to-end neural network, the capability to normalize the speaker variability could be incorporated into an ASR system. The efficiency of these normalization methods depends on the representation used for unseen speakers.

7.1.1 Related Work

Speaker normalization is a crucial module in operating an ASR with high performance. Though there have been several approaches for speaker normalization in hybrid ASR, an optimal mechanism of speaker normalization in an end-to-end speech recognition systems is largely unexplored. In this study, we explore approaches for speaker normalization in the perspective of end-to-end speech recognition
system. The key role of speaker normalization is to reduce the performance drop when speech recognition system encounters an unseen speaker [150]. The degradation in the performance of speech recognition systems due to speaker variabilities is addressed by two major approaches, the former approach is to reduce the speaker variabilities by speaker normalization [151, 152, 153] and later is by speaker adaptation [154, 155]. The variabilities that are specific to a speaker are normalized to develop a speaker independent recognition system and the developed system can be adapted to operate for a specific speaker.

Some of the widely used speaker normalization approaches are Vocal tract length normalization scheme (VTLN) [152, 153] and Feature space maximum likelihood regression (fMLLR) [151]. VTLN approach aims to warp the frequency axis to account for the varying vocal tract lengths across the speakers and the warp factor is computed from the speaker’s data. VTLN warp factor for a speaker can also be computed through a grid search to choose the optimal warping factor to maximize the likelihood of speakers data for the corresponding label sequence [153]. Speaker normalization using fMLLR is carried out by estimating a linear transform over the input features that would maximize the likelihood of speakers data for the given label sequence, during the test time an initial label sequence is obtained for the existing model and the features from the speaker are transformed to maximize the likelihood [151]. Cepstral mean and variance of the data normalized on per speaker basis is used as a kind of speaker normalization procedure.

Majority of the approaches model the speaker characteristics as a linear transform over input features or certain model parameters and use them to normalize the speaker-specific characteristics. The performance of such speaker normalization hugely relies on the amount of data available to compute the transform, less data per speaker results in a poor estimate of transform and poor normalization [151]. In a practical scenario, only a single utterance of an unknown speaker can be accessed, and the speaker normalization approach should be able to normalize the speaker variabilities and yield a better performance using the single available utterance. In VTLN and fMLLR normalization methods, every speaker is modeled independent of the remaining speakers, and such approaches can not model the similarity and variability across the speakers [156]. In the proposed approach, we hypothesize that by suitably providing information about the identity of the speaker along with features, the capability of speaker normalization can be embedded into a neural network that is trained for speech recognition. Where the performance of this normalization approach relies on efficiently representing the characteristics of unseen speakers from the characteristics of trained speakers.

Some of the related studies for speaker normalization in a hybrid ASR are briefly described, spectral features are appended with i-vectors and the appended features are explored for training hybrid ASR [157]. In [158], the bottleneck features accumulated over the entire speaker is appended with spectral features and the appended features have been used as speaker normalized features. Some additional
weights are learned at every layer using i-vectors, and the network is used to develop a speaker adapted hybrid ASR. In [159], two bottleneck networks were trained to classify speaker and phone identities, these bottleneck features are appended and the appended features are used to train a deep neural network for speech recognition. In an end-to-end ASR, the acoustic model used is sequential as opposed to a deep neural network (DNN) in hybrid ASR. The end-to-end networks are expected to benefit more from the speaker information locally, and the accumulation of local information can happen in the sequential acoustic model, where utterance level speaker representations have proven to perform better in the case of a hybrid ASR.

This work proposes a new speaker normalization method, which aims at representing the characteristics of the test speaker as a weighted combination of the set of known speakers on which ASR has been trained. Such a system would demand a single unifying framework to normalize over multiple speakers. In this work, the representational power of DNN is used for modeling the speaker characteristics of unseen speakers.

7.2 Database & Experimental setup for end-to-end Speech Recognition System

7.2.1 Database

Wall street journal corpus (WSJ) comprising 80 hrs of speech data has been used during the study [112]. The utterances from si-284 are used for training the speech recognition systems and the performance is evaluated on eval-92. The training corpus comprises of data from 282 speakers, while testing corpus comprises of 8 speakers that are independent of the train set. The standard evaluation data (eval 92) comprises of 333 utterances i.e., approximately 40 utterances per speaker.

7.2.2 Architecture

Recurrent neural network (RNN)-CTC based speech recognition systems have been explored. The network has 4 layers of bidirectional long short-term memory networks (BLSTMs) and each layer is comprising of 360 units. A batch size of 10 is used. During the training, forward propagation is done with a batch of 10 utterances and backpropagation is done for every utterance. Learning rate is reduced by a factor of 0.5 when a decrease in validation accuracy is encountered. The training is progressed till a minimum increase of 0.05 is maintained in validation accuracy. RNN-CTC is optimized to generate character sequences from the input acoustic sequences. The character-based lexicon generated from CMU dictionary and NIST trigram language model has been used in this work. During the study, spectral features with deltas and delta-deltas are used as a feature representation. The gradient at each
layer is clipped at a maximum value of 50, all the parameters of the model are initialized from a uniform distribution between [-0.1, 0.1].

7.3 Proposed Approach for Speaker Normalization

7.3.1 Approach 1

Widely used speaker normalization methods can be generalized as the linear transform over the input data computed per speaker. The hypothesis for the proposed approach is that, by suitably giving information about the speaker’s identity while training the model, the model would be optimized such that the speaker normalization occurs implicitly. The efficiency of the approach relies on the form in which information about training and the unseen speaker’s identity is presented to the network. The identity of the speaker in the train set is represented using a one-hot vector. The one-hot vectors of the speaker are appended with spectral features and the appended features are used for training the RNN-CTC model. In this approach, the unseen speaker characteristics are to be represented by a weighted combination of known (trained) speakers. A speaker recognition system trained to classify the training speakers (SPKID-DNN) is employed for this task. For the test utterance, the softmax vectors obtained from the SPKID-DNN are appended with the spectral features and the appended features are used to test the speech recognition system. The softmax vectors from SPKID-DNN is expected to represent the unseen test speaker as a weighted combination of known (training) speakers. The performance of the proposed approach is presented in Table 7.1.

SPKID-DNN is a fully connected feed forward deep neural network comprising of 4-hidden layers with rectified linear units (ReLU) and a softmax output layer. The architecture of SPKID-DNN is 39R-1024R-1024R-1024R-1024R-282S. During the study, SPKID-DNN needs to be generalizable over the unseen speakers, to obtain such a network various SPKID-DNNs have been trained with different sized datasets i.e., full, 10, 20, 30, and 60 utterances per speaker from the training set and 10 utterances from each speaker is used as validation set for training SPKID-DNN. Spectral features are used to train SPKID-DNN. The network is optimized using Adadelta [114] optimizer to minimize the cross-entropy loss, an initial learning rate of 0.001 is used. Learning rate is reduced by a factor of 0.5 upon

<table>
<thead>
<tr>
<th>Speaker normalization method</th>
<th>fMLLR per utt</th>
<th>fMLLR per-spk</th>
<th>MVN per-utt</th>
<th>MVN per-spk</th>
<th>SPKID-DNN-10utt</th>
<th>SPKID-DNN-20utt</th>
<th>SPKID-DNN-30utt</th>
<th>SPKID-DNN-60utt</th>
<th>SPKID-DNN-full</th>
<th>SPKID-DNN-30utt-sent label</th>
</tr>
</thead>
</table>
encountering a decrease in validation accuracy. The network is trained until a decrease in validation accuracy is observed over three successive epochs.

Row 1 of Table 7.1 are various speaker normalization approaches and row 2 are the word error rates (WERs) of ASR obtained using the corresponding normalization. Column 2 of Table 7.1 is WER of ASR with fMLLR speaker normalization. The reported WER is on the eval-92 set comprising of 8 speakers and each speaker has around 40 utterances. WER in Column 2 is obtained by using data from all 40 utterances in estimating the fMLLR transform. Column 3 is the WER obtained when data from a single utterance is used to estimate fMLLR. Form Column 2 and 3 it can be observed that the performance of the systems degrades with insufficiency in the data for estimating fMLLR transform. Column 4 is the system trained using the features with mean and variance normalization (MVN) performed on per speaker basis. Column 5 is the ASR system trained with mean and variance normalization performed using the data from single utterance (MVN-per utt). In the proposed approach, the identity of the training speaker is presented as on-hot vectors. Spectral features are appended with one-hot speaker vectors of 282 dimensions to form 321-dimensional input features and these features are used to train an RNN-CTC network. The performance of the proposed approach evaluated using various SPKID-DNNs is presented in column 6-9 in Table 7.1. From columns 6-9, it can be noted that the SPKID-DNN-30utt have performed better compared to other systems. From Table 7.1, the ASR system with the proposed normalization (i.e., trained with one-hot speaker vector during the training and representing unseen speaker as a weighted combination of trained speakers) have reduced the absolute word error rate by 0.6. The approach is more practically feasible as it used only single utterance to normalize the speaker, unlike fMLLR which uses around 40 utterances for normalizing speaker variabilities. For the given test utterance, the closest speaker from the train speaker is found using SPKID-DNN and the corresponding one-hot vector is used for testing and the obtained WER is given by column 10 of Table 7.1. Though speaker is an attribute of the entire utterance, a DNN is preferred as it would account for the local variabilities and accumulation of these representations over the utterance is left to the sequential acoustic model (RNN-CTC).

7.3.2 Approach 2

Approach-1 implicitly assumes that every speaker is independent of the other speakers in the training set. Such an assumption can not model inter-speaker similarity and variability, which might be helpful in modeling the unseen speaker. Moreover, approach-1 cannot model the phonetic context along with the speaker’s identity. To embed this additional information, the softmax vectors of SPKID-DNN is used as a speaker vector. The speaker vector is appended with spectral features, and the appended feature representation is used for training RNN-CTC model. Blockdiagram of speaker normalization using approach 2 is presented in Fig.7.1. The WERs obtained from the proposed approach are reported
Table 7.2: Performance of the proposed speaker normalization approach using softmax vectors from
SPKID-DNN as representations for the speakers identity.

<table>
<thead>
<tr>
<th>Speaker normalization method</th>
<th>SPKID-DNN-10utt</th>
<th>SPKID-DNN-20utt</th>
<th>SPKID-DNN-30utt</th>
<th>SPKID-DNN-60utt</th>
<th>SPKID-DNN-full</th>
<th>i-vector</th>
<th>SPKID-DNN-Bottleneck</th>
</tr>
</thead>
<tbody>
<tr>
<td>WER</td>
<td>9.84</td>
<td>9.21</td>
<td>8.88</td>
<td>9.14</td>
<td>8.93</td>
<td>10.97</td>
<td>10.10</td>
</tr>
</tbody>
</table>

in Table 7.2. Multiple ASR systems have been trained using different SPKID-DNNs to generate speaker vectors.

From Table 7.1 and 7.2 it can be observed that, though the proposed speaker normalization method uses only a single utterance it can yield a performance that is equivalent to using 40 utterances in computing a fMLLR transform. The proposed method is more advantageous and employing the proposed speaker normalization approach an absolute reduction of 1.3 in WER can be observed.

Row 1 of Table 7.2, is the speaker normalization approach and row 2 is the WER of ASR trained with the corresponding speaker normalization. Column 2-6 are the WERs of ASR systems trained using speaker normalization method proposed in approach-2. Column 8 is the ASR system trained by appending 100-dimensional i-vector along with spectral features and the i-vectors used in this work are computed using a Universal background model (UBM) of 512 components, all the training data is used to compute i-vectors. Pre-softmax activations of SPKID-DNNs is used as speaker vector and appended
with the spectral features and the performance of the ASR systems is presented in column 8. The performance of ASR is superior when softmax vectors from SPKID-DNN-30utt are used training the network. Having some utterances for training ASR systems which are unseen by SPKID-DNN have helped the model to generalize over the unseen test set.

During the study, recurrent neural networks (RNNs) have been explored to classify the training speakers (SPKID-RNN). Deep bi-directional LSTMs comprising of three layers and each layer comprising of 250 units have been used to train the SPKID-RNN. The network is optimized as mentioned in section 7.3.1. The speaker vectors obtained from SPKID-RNN are used in the proposed approach and the performances of ASR are tabulated in column 2,3 of Table 7.3. The speaker vectors obtained from the SPKID-DNN are appended with speaker vectors SPKID-RNN are combined speaker vectors are used in the proposed approach the performance of the ASR using the combined speaker vector is presented column 4,5 of Table 7.4. From Table 7.3, it can be observed that the speaker vectors from SPKID-DNN have performed better that SPKID-RNN. It can also be noted that the performance of proposed approach using speaker vectors from both SPKID-DNN and SPKID-RNN have performed similar to using only SPKID-DNN.

Table 7.3: Performance of the proposed speaker normalization approach using softmax vectors from SPKID-DNN and SPKID-RNN as representations for the speakers identity.

<table>
<thead>
<tr>
<th>Speaker normalization method</th>
<th>SPKID-LSTM-30utt</th>
<th>SPKID-LSTM-fullutt</th>
<th>SPKID-LSTM-DNN-fullutt</th>
<th>SPKID-LSTM-DNN-30utt</th>
</tr>
</thead>
<tbody>
<tr>
<td>WER</td>
<td>11.48</td>
<td>9.82</td>
<td>11.02</td>
<td>8.90</td>
</tr>
</tbody>
</table>

From Table 7.1 and 7.2 it can be observed that, though the proposed speaker normalization method uses only a single utterance it can yield a performance that is equivalent to using 40 utterances in computing a fMLLR transform. The proposed method is more advantageous and employing the proposed speaker normalization approach an absolute reduction of 1.3 in WER can be observed.

7.4 Complementarity of the Proposed Approaches with fMLLR based Speaker Normalization

As fMLLR is a global transform to normalize the speaker variabilities and the proposed approach is expected to account for local variabilities. To study the complementary nature of the proposed approach along with fMMLR based normalization. Both the proposed normalization methods are employed while
training an ASR with fMLLR features. The performances obtained by combining both the normalizations are reported in Table 7.4. SPKID-DNN is also trained with fMLLR features.

Table 7.4: Performance of the speech recognition systems developed using fMLLR features and the proposed normalization approaches.

<table>
<thead>
<tr>
<th>Speaker normalization method</th>
<th>one-hot speaker codes</th>
<th>SPKID-DNN softmax vectors</th>
</tr>
</thead>
<tbody>
<tr>
<td>fMLLR-SPKID-DNN-30utt</td>
<td>8.75</td>
<td>8.93</td>
</tr>
<tr>
<td>fMLLR-SPKID-DNN-full</td>
<td>8.77</td>
<td>8.59</td>
</tr>
</tbody>
</table>

Form Table 7.4, the complementary nature of the proposed approach can be observed, a slight decrease in WERs is observed. SPKID-DNNs trained with spectral features can also be used and they have also yielded similar performances.

7.5 Using the Proposed Speaker Normalization for Multilingual ASR System

The speaker normalization approach described in the above section is extended for developing a multilingual ASR system. The speaker ID networks trained on WSJ corpus have been used for this study. AS mentioned in the previous section the speaker vectors for the input utterance is combined with the spectral features and combined features are used for developing multilingual ASR system with a joint acoustic model. Column 2 is the performance of joint acoustic model trained as described in chapter 3. Column 3,4 are the WERs obtained using DNN based speaker ID networks and Column 5,6 are the DNN-LSTM based speaker ID networks. From the Table 7.5 it can be observed that used of the proposed speaker normalization approach has improved the performance of multilingual ASR system.

Table 7.5: Improving the Performance of Multilingual-ASR using proposed speaker normalization approach.

<table>
<thead>
<tr>
<th>Test set</th>
<th>Telugu</th>
<th>Tamil</th>
<th>Gujarati</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>20.61</td>
<td>20.16</td>
<td>14.19</td>
</tr>
<tr>
<td>SPKID-DNN-30utt</td>
<td>19.79</td>
<td>20.75</td>
<td>14.11</td>
</tr>
<tr>
<td>SPKID-DNN-fullutt</td>
<td>19.55</td>
<td>19.61</td>
<td>13.79</td>
</tr>
<tr>
<td>SPKID-RNN-30utt</td>
<td>19.76</td>
<td>19.69</td>
<td>13.98</td>
</tr>
<tr>
<td>SPKID-RNN-fullutt</td>
<td>19.42</td>
<td>19.33</td>
<td>13.03</td>
</tr>
</tbody>
</table>
7.6 Summary and Conclusions

Information about the speaker ID of an utterance can be considered as the meta-information about the utterance. In this work, it is hypothesized that by suitably describing the identity of the speaker while training an end-to-end neural network, the model is optimized such that it is robust to speaker variabilities. During the study, the identity of the known speaker is represented by using two approaches viz., one-hot speaker codes and a weighted combination of training speakers, while the un-seen speaker’s identity is represented as a weighted combination of the seen speakers representations. For an utterance, the weighted combination of training speakers is obtained by a DNN trained to classify the training speakers. This normalization method would be effective even for single test utterance and the proposed approaches reduce the absolute WER by 0.6 and 1.3 respectively. Later this works has been extended to multilingual ASR systems using a joint acoustic model. The performance of the proposed speaker normalization is consistent in developing multilingual ASR systems. The proposed speaker normalization methods would efficiently normalize the speaker variabilities even when only a single utterance from the test data.
Chapter 8

Summary and Conclusions

In this thesis, we have addressed various related to the development of a multilingual ASR system for Indian languages. The thesis proposes an integrated multilingual ASR for Indian languages. The proposed ASR can efficiently be operated in multilingual code-mixed data environments. The thesis addresses various issues related to suitable and efficient acoustic models for developing an integrated ASR, use of use of articulatory features along with conventional features and a speaker normalization method for these acoustic models. The major contributions of this thesis are as follows:

8.1 Conclusions

• An integrated multilingual ASR:
  We have proposed a framework for developing an integrated multilingual ASR. The acoustic modeling approaches which are suitable for training joint acoustic models for different languages have been explored. The acoustic models which can model mono-phones are found to be efficient for developing joint acoustic models. This work has show that RNN-CTC based acoustic model which models context independent phones is efficient in training a joint acoustic model compared to HMM-SGMM which models context dependent tri-phones. This works has shown that the model exhibits a better data sharing capabilities across languages.

• Use of Residual networks:
  When the network is sufficiently deep having residual connections between successive training have stabilized the training and improved the performances of the acoustic model. Dense connections which are a recent variant of residual networks have improved the performance of RNN-CTC and sequence-to-sequence joint acoustic models.

• Use of articulatory features for improving the performance of ASR:
  This work proposes S-Transform and spectral weighting based approaches for detecting fricative
regions from continuous speech. The detected fricative evidences are used along conventional features by an early fusion. The combined features have shown improvements when the size of the data-set is small. When the size of the dataset is large the improvements are not significantly better. A deep-neural network is trained to predict articulatory features from the acoustic sequence and these predicted articulatory feature vector is used along with the conventional feature for developing multilingual ASR. They have slightly improved the performance of the system.

- **Speaker normalization using speaker embeddings:**
  The information about the speaker ID is considered as the meta-level information about the data. This work proposes a speaker normalization when this information is not available. This work has shown that, by suitably describing the identity of the speaker while training an end-to-end neural network, the model can be optimized such that it is robust to speaker variabilities. The un-seen speaker’s identity is represented as a weighted combination of the seen speakers representations. For an utterance, the weighted combination of training speakers is obtained by a DNN trained to classify the training speakers.

- **Sequence-to-sequence based multilingual integrated ASR**
  Sequence-to-sequence with attention models have been explored for developing multilingual ASR systems. Similar to RNN-CTC sequence to sequence with attention can also be used as joint acoustic models.

### 8.2 Future Work

Following are some directions for future work:

- **Adapting to new Indian languages:**
  Extending the integrated ASR framework to new Indian languages can be very beneficial. The systems share phones across different languages. Most of the phones that are going to be present in the new language are already present in the systems. The partial decoding or the phone sequence obtained form the trained model can be used to reduce the transcription cost.

- **Data-Driven approach for obtaining common phone-set:**
  The phones that are similar in different languages are merged in designing a common phone-set. For Indian languages these are done by rule based approaches. Deriving this rules from data driven approach can help to extend this systems for various languages.
• **Integrating English to multilingual ASR**
  Most Indian languages code-switch with English. An efficient way of integrating Indian English to multilingual ASR needs to be explored.

• **Better RNNs:**
  Improving the RNNs can improve the performance of the acoustic models.

• **Sequence-to-sequence learning:**
  Sequence-to-sequence with attention models for developing multilingual ASR systems is gaining a significant interest. These models could be adopted for developing integrated ASR systems for Indian scenarios. Better RNNs and efficient attention mechanism can always improve the performance of these systems.

• **Sequence-to-sequence learning with less data:**
  Sequence-to-Sequence with attention is efficient when the data size is sufficiently large. Approaches to train these models with smaller size data needs to be explored. Training these models with unsupervised and partially supervised examples can be explored.
Appendix A

Multilingual Speech Recognition using Attention based Sequence-to-Sequence Learning

A.1 Introduction

In Chapter 3, RNN-CTC models have been used for developing an integrated multilingual ASR system. Similar to RNN-CTC models sequence-to-sequence with attention models are being widely explored for developing end-to-end ASR systems. RNN-CTC models predicts the targets for every frame and assumes that the targets are conditionally independent of each other. This assumption limits the model form leaning sequential relations among the labels. Sequence-to-sequence with attention models does not have this assumption and this model can learn sequential relations among the data. Attention models can efficiently learn to produce the label sequence (phonemes or characters) without any additional knowledge sources such as lexicon and language models combined as WFST graph. A character RNNLM integrated with attention models has improved the performances.

Recently attentional models have been widely studied for developing end-to-end ASR systems [32, 2, 33]. This models can be seamlessly extended to train multilingual ASR systems learning the character or orthographic sequences from acoustic sequences [95, 160, 161, 162, 73, 74, 75]. This chapter investigates the performance attention models for developing an integrated speech recognition systems described in Chapter 3.

In the most familiar form, these models consists of an encoder RNN to convert the acoustic sequence to sequence of hidden states and the decoder is an autoregressive RNN to generate the output sequence. At each decoding time step decoder is conditioned by the attention module, which allows the decoder to refer back to certain parts of encoded hidden states. The blockdiagram describing various blocks of sequence-to-sequence with attention model is presented in Fig. A.1. The attention mechanisms studied in literature can be majorly grouped to two types i.e., the attention mechanism in [33] is termed as
LA-ATT and attention mechanism in [163] is termed as MO-ATT in this chapter. The block describing various blocks of sequence to sequence with attention model is presented in Figure. A.1.

Let the input sequence be $x = (x_1, x_2, ..., x_L)$ and $h = (h_1, h_2, ..., h_L)$ is the output of the encoder RNN and output $y$ is the sequence of phonemes. The output units comprises of phonemes or characters and some additional symbols to represent space(<SPACE>), start of sequence (SOS), end of sequence (EOS). Label output $y_i$ is generated by the following process:

Figure A.1: Blockdiagram showing different blocks in sequence-to-sequence with attention.
This chapter studies the use of both the attention mechanisms for developing multilingual ASR systems.

A.2 Experiments and Results

The database described in Chapter 3 is used for developing multilingual ASR using attention models. MFCC features are used to represent the input acoustic sequence. The phoneme sequences are used as the output sequences, along with the phoneme sequence the output sequence also comprises of extra symbols such as space (<SPACE>), start of sequence (SOS) and End of Sequence (EOS). The encoder comprises of LSTM-LP layers described in Chapter 4. The encoder is a stacked Bi-LSTM-LP with 320 nodes per each layer and decoder has a unidirectional LSTM with 320 nodes. This study has used additive attention with and without locate based mechanism. The encoder sequence is sub-sampled by a factor of 2 after first and second layers. The network is optimized using ADAM optimizer. The network is trained with an initial learning rate of 0.0001 and the learning rate is reduced by a factor of upon observing an increase in the validation accuracy. An increase in the accuracy for three successive epochs is considered as an early stopping criterion. Networks are trained with a batch size of 10. A teacher forcing factor of 0.6 is used while training and no teacher forcing is used while validating the model. The results are reported in terms of phone error rate (PER). No language model is used in this study. The PER obtained for different attention models is presented in Table A.1. The attention plots describing the process of alignment between the acoustic and label sequence is presented in Figure A.2, A.3, A.4, A.5.
Figure A.2: Plots showing the attention pattern of an utterance across different epochs. The figure is obtained from LA-ATT with location based attention trained using LSTM-LP-7H. The figure is generated during the training phase with a teacher forcing of 0.6.
Figure A.3: Plots showing the attention pattern of an utterance across different epochs. The figure is obtained from MO-ATT with location based attention trained using LSTM-LP-7H. The figure is generated during the training phase with a teacher forcing of 0.6.
Figure A.4: Plots showing the attention pattern of an utterance across different epochs. The figure is obtained from LA-ATT with location-based attention trained using LSTM-LP-7H. The figure is generated during the validation phase with a teacher forcing of 0.
Figure A.5: Plots showing the attention pattern of an utterance across different epochs. The figure is obtained from MO-ATT with location based attention trained using LSTM-LP-7H. The figure is generated during the validation phase with a teacher forcing of 0.
Table A.1: Performances of multilingual ASR system developed using sequence-to-sequence with attention model.

<table>
<thead>
<tr>
<th>Attention mechanism</th>
<th>Encoder model</th>
<th>Telugu</th>
<th>Tamil</th>
<th>Gujarati</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dev set</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>LA-ATT:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Additive attention</td>
<td>LSTM-LP-3H</td>
<td>15.23</td>
<td>12.51</td>
<td>17.15</td>
</tr>
<tr>
<td></td>
<td>LSTM-LP-5H</td>
<td>15.36</td>
<td>12.43</td>
<td>16.89</td>
</tr>
<tr>
<td><strong>LA-ATT:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Location based</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Additive attention</td>
<td>LSTM-LP-3H</td>
<td>14.02</td>
<td>11.22</td>
<td>15.13</td>
</tr>
<tr>
<td><strong>MO-ATT:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Additive attention</td>
<td>LSTM-LP-3H</td>
<td>15.05</td>
<td>12.08</td>
<td>16.21</td>
</tr>
<tr>
<td></td>
<td>LSTM-LP-5H</td>
<td>12.86</td>
<td>10.36</td>
<td>14.14</td>
</tr>
<tr>
<td><strong>MO-ATT:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Location based</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Additive attention</td>
<td>LSTM-LP-3H</td>
<td>14.21</td>
<td>11.55</td>
<td>16.11</td>
</tr>
<tr>
<td></td>
<td>LSTM-LP-5H</td>
<td>12.42</td>
<td>10.19</td>
<td>13.86</td>
</tr>
<tr>
<td></td>
<td>LSTM-LP-7H</td>
<td>11.70</td>
<td>9.61</td>
<td>12.72</td>
</tr>
<tr>
<td></td>
<td>RES-LSTM-LP-7H</td>
<td>12.15</td>
<td>10.22</td>
<td>13.21</td>
</tr>
<tr>
<td></td>
<td>Dense-LSTM-LP-7H</td>
<td>11.37</td>
<td>9.51</td>
<td>12.57</td>
</tr>
</tbody>
</table>

Column 1 of Table A.1 is the attention model used for developing the multilingual speech recognition system. Column 2 is the encoder networks used. Column 3, 4 and 5 are the performances of multilingual ASR systems on Telugu, Tamil and Gujarati respectively. LSTM-LP encoders described in Chapter 4 are used in this study. From Table A.1, it can be noted that the PER of the system has increased as we increase the depth of the network. From Table A.1, increasing the depth of the network from 5 to 7-hidden layers an improvement in PER can be noted. Using the location based attention along content based attention the PER has improved using LA-ATT and MO-ATT. From the Table A.1, ASR
trained using MO-ATT has performed better than the model trained with LA-ATT. Dense LSTM-LP has further improved the performance of multilingual ASR.

A.3 Summary and Conclusions

Sequence to sequence with attention models have been explored for developing an integrated multilingual ASR systems. This study has explored different attention mechanism for developing multilingual ASR. The performance of the systems is reported in terms of phone error rate (PER). These systems have obtained an average PER of 11.3%. Along with phones, special symbols such as SPACE (<SPACE>), start-of-sequence (SOS) and end-of-sequence (EOS) as output symbols. The model could be used to generate the text sequence for the acoustic sequence. The model can be efficiently operated without a lexicon and a language model. Using RNNLM based language model can improve the performance of this system.
Bibliography


[76] T. Hussain and K. Samudravijaya, “Comparison and usefulness of asr11 scheme over previous schemes for transliteration and label set purposes for indian languages,”


Publications

Related publications

Journals


• Hari Krishna Vydana, Anil Kumar Vuppala “Detecting Fricatives by Spectral Weighting : A Temporal Approach” (under review Speech Communication.)

Conferences


• Hari Krishna Vydana, Anil Kumar Vuppala “Residual Neural Networks for Speech Recognition” EUSIPCO 2017 Kos Island,GREECE


Other publications

Journals


• Ravi Kumar V, Hari Krishna Vydana and Anil Kumar Vuppala “Curriculum Learning Based Approach for Noise Robust Language Identification using DNN With Attention”. Accepted in Elsevier Expert Systems with Applications.

Conferences


Academic record

Name: Hari Krishna Vydana
Date of Birth: 11-01-1990
Program Name: PhD (in ECE)
Area of Research: Speech Processing
Date of Joining for Ph.D: Dec-2014

Courses attended:

<table>
<thead>
<tr>
<th>Courses attended</th>
<th>Course Code</th>
<th>Grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>Artificial Neural Networks</td>
<td>Spring 2015-ECE449</td>
<td>A</td>
</tr>
<tr>
<td>ECE-Independent Study</td>
<td>Spring 2015-ECE411</td>
<td>A</td>
</tr>
<tr>
<td>Speech Systems</td>
<td>Spring 2015-ECE446</td>
<td>A</td>
</tr>
<tr>
<td>Speech Signal Processing</td>
<td>Monsoon 2015-ECE448</td>
<td>A</td>
</tr>
<tr>
<td>Time Frequency Analysis</td>
<td>Monsoon 2015-ECE442</td>
<td>B-</td>
</tr>
<tr>
<td>Topics in Speech Systems: Text to Speech Conversion</td>
<td>Monsoon 2015-CSE973</td>
<td>A</td>
</tr>
</tbody>
</table>

CGPA 9.5