SPEECH EMBEDDING WITH SEGREGATION OF PARALINGUISTIC INFORMATION FOR LOW-RESOURCE LANGUAGES

Anosha Ignatius

208054J

Thesis submitted in partial fulfillment of the requirements for the degree of Master of Science (Major Component of Research) in Computer Science and Engineering

Department of Computer Science and Engineering

University of Moratuwa

Sri Lanka

November 2021

Declaration

I declare that this is my own work and this thesis does not incorporate without acknowledgement any material previously submitted for a Degree or Diploma in any other University or institute of higher learning and to the best of my knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text.

Also, I hereby grant to University of Moratuwa the non-exclusive right to reproduce and distribute my thesis, in whole or in part in print, electronic or other medium. I retain the right to use this content in whole or part in future works (such as articles or books).

Signature: Date:

The above candidate has carried out research for the Masters thesis under my supervision.

Name of the supervisor: Dr. Uthayasanker Thayasivam

Signature of the supervisor: Date:

Abstract

Speech embeddings produced by Deep Neural Networks have yielded promising results in a variety of speech processing applications. However, the performance in speech tasks like automatic speech recognition and speech intent identification can be affected to a great extent when there is a discrepancy between training and testing conditions. This is because, in addition to linguistic information, speech signals carry para-linguistic information including speaker characteristics, emotional states, and accent. Variations in the speaker traits and states lead to compromise on performance in speech recognition applications that require only linguistic information.

Over the years, there have been various approaches that attempt to disentangle the para-linguistic information that support the linguistic information in speech. The commonly used strategy is to integrate speaker representations into speech recognition models to normalise the speaker effects. Still, it has received less attention when it comes to studies on speech-to-intent classification. Furthermore, large amounts of labeled speech data are required for these speaker normalisation techniques. Under low-resource settings, when there is only a limited number of speech samples available for training, transfer learning strategies can be used.

This study presents a speaker-invariant speech intent classification model using i-vector based feature augmentation. We investigate the use of pre-trained acoustic models for transfer-learning under low-resource settings. The proposed method is evaluated on the banking domain speech intent dataset in Sinhala and Tamil languages along with fluent speech command dataset. Experimental results show the effectiveness of the proposed method in achieving better prediction in the speech-to-intent classification model.

Keywords: speech-to-intent, speech recognition, linguistic, para-linguistic information, speaker representation

Acknowledgements

First of all, I would like to express my sincere gratitude to my supervisor, Dr. Uthayasanker Thayasivam, for all the guidance and support extended by him. He has motivated me with his immense knowledge and abundant experience throughout my research.

I am very much grateful to Prof. Gihan Dias, who was ever willing to assist me in my research with his continuous guidance. I am also thankful to Dr. Surangika Ranathunga and Prof. Sanath Jeyasena for graciously providing me with feedback on my work.

I would like to extend my sincere thanks to my progress review panel members Dr. Charith Chitraranjan and Dr. C.R.J. Amalraj for their valuable feedback which helped me strengthen my work.

This research study was supported by Accelerating Higher Education Expansion and Development (AHEAD) Operation of the Ministry of Education, Sri Lanka funded by the World Bank. I am thankful to LK Domain Registry for the financial support received to attend the SPECOM 2021 conference.

My heartfelt thanks go to my colleagues and friends who have helped me to complete my research successfully. Finally, I would like to thank my family members, who are always there to support me in all my endeavours.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

ASR	**A**utomatic **S**peech **R**ecognition
SLU	**S**poken **L**anguage **U**nderstanding
DNN	**D**eep **N**eural **N**etwork
LLD	**L**ow **L**evel **D**escriptor
MFCC	**M**el **F**requency **C**epstral **C**oefficients
LPC	**L**inear **P**redictive **C**odes
PLP	**P**erceptual **L**inear **P**rediction
GMM	**G**aussian **M**ixture **M**odel
UBM	**U**niversal **B**ackground **M**odel
RNN	**R**ecurrent **N**eural **N**etwork
APC	**A**utoregressive **P**redictive **C**oding
NPC	**N**on-autoregressive **P**redictive **C**oding

Chapter 1

Introduction

Speech-to-intent systems aim at extracting the meaning contained in spoken utterances. It has attracted very much interest from researchers in the field of speech processing, with the growing demand for voice-controlled applications. A number of recent research studies [5–7] have presented end-to-end approaches for speech intent classification and there have been promising results with the availability of large amounts of speech data and the advancement of Deep Neural Networks (DNNs). However, the performance of speech-to-intent systems can still be compromised greatly when the testing conditions differ from training conditions. This is due to variations in the speaker characteristics, speaking manner, emotional expressions, and the acoustic environment. This kind of information found in speech signals conveys non-verbal aspects of communication and is called para-linguistic information. It is challenging to develop techniques that alleviate the mismatch between test data and training data caused by speaker variability. Therefore, conducting extensive research on normalising the speaker effects is important to build robust speech-to-intent systems.

Time and frequency domain information present in speech signals are typically represented using various audio low-level descriptors (LLDs) such as Mel Frequency Cepstral Coefficients (MFCCs) [8], Perceptual Linear Prediction Cepstral Coefficients (PLP-CCs) [9], and Linear Predictive Codes (LPCs) [10]. In addition, intermediate representations extracted from DNN based speech models are widely used in recent studies. Approaches developed for generating speaker-invariant speech representations focus on speaker-aware training where the speech features are augmented with speaker embeddings. Speaker embeddings map speech utterances to fixed dimensional vectors. Several types of speaker embeddings have been investigated in research studies on speaker recognition and speaker verification. They include i-vectors [11] and DNN based embeddings [12, 13]. When the acoustic model is provided with speaker-level information as additional input features, it learns how to remove speaker variations present in speech and generate disentangled representations.

Speaker-aware training has been proven to be efficacious in enhancing the performance of DNN based Automatic Speech Recognition (ASR) models [14–18], since it helps in learning representations that separate speaker characteristics and achieving better generalisation of the model to unseen data. However, speaker normalization has been less studied in spoken intent detection systems. Speaker adaptation using i-vectors in end-to-end Spoken Language Understanding (SLU) models is investigated in the paper by Natalia Tomashenko et al. [19]. It has been shown that speaker adaptation can play a significant role in enhancing the performance of SLU models. In low resource speech-to-intent classification, when there is only a limited amount of training data, transfer learning strategy can be used. In such cases, self-supervised speech features learnt using large amounts of unlabeled training data can be used to learn acoustic models with limited amounts of labeled training data. Such representations attempts to retain as much linguistic information as possible, which is desired several downstream tasks such as speech intent classification. Many studies have investigated unsupervised learning methods for extracting meaningful speech representations [3, 20, 21] that can be used under low resource settings.

In this study, we focus on speaker normalisation for speech intent classification under low-resource settings. We investigate the efficacy of transfer learning with pre-trained speech models and speaker normalisation using i-vectors in speech intent identification for Sinhala and Tamil languages. For experiments, we use Sinhala and Tamil speech datasets consisting of speech recordings in the banking domain. We use Fluent speech command dataset for further comparison. Experimental results indicate that the incorporation of i-vectors is effective in normalising the speaker effects and improving the recognition performance.

1.1 Speech-to-intent Classification

Spoken Language Understanding (SLU) systems which play a significant role in voice assistant applications, aim to extract the semantic information contained in a spoken utterance. Typically it is performed through transcriptions obtained from ASR systems. Such model consists of two components ASR module and Natural Language Understanding (NLU) module. The NLU module performs tasks such as domain classification, intent detection, and slot filling. In this work, we focus on the intent detection task, which maps speech to the speaker's intent. Despite the remarkable performance achieved by DNN based speech intent classification systems suffer from ASR errors. Therefore, an end-to-end speech intent classification system which classifies the intent directly from speech signal is considered as an alternative approach.

1.2 Research Problem

The presence of paralinguistic information in speech signal causes performance degradation in speech tasks. It is challenging to build a spoken intent detection system that is robust to variations in the speaker characteristics, since it requires large amounts of labeled speech data for training. In addition, the training data should consist of speech segments from a number of speakers with different traits covering a broad range of

accents and age groups. However, the available speech datasets have failed to do so. Speaker normalization techniques attempt to reduce the acoustic mismatch between testing and training conditions by generating speaker-invariant features. In case of a low resource scenario, when there is only a limited amount of labeled speech data available for training, segregating the linguistically insignificant speaker-specific content contained in the acoustic feature representations through supervised learning is challenging. This could result in poor generalization of the acoustic model.

1.3 Contributions

Speech-to-intent model for low resource languages: In this thesis, we propose an i-vector based speaker-invariant speech-to-intent model that outperforms the baseline model without i-vectors. We investigate the use of transfer learning approach in low resource settings by carrying out experiments on the banking domain speech intent dataset in Sinhala and Tamil languages. We evaluate the proposed model on English speech command dataset as well. Eventhough many researchers have explored speaker normalization techniques for speech recognition systems, speaker normalization has not been previously applied in low-resource speech-to-intent systems. Our approach proved to be effective in segregating the underlying speaker information and improving the recognition performance. Furthermore, it works well with a low amount of training data.

Tamil speech intent dataset (Healthcare domain): We built a Tamil speech intent dataset in healthcare domain via crowd-sourcing for further experimental evaluation[1]. We decided to create a new dataset as the availability of speech datasets in different domains could help in developing a well generalised model for speech intent prediction. The dataset consists of speech recordings describing a set of symptoms. The details of the dataset are presented in Chapter 5.

[1] `https://github.com/anoshai/speech-intent-classification`

Publications: During the study period, we have published two survey papers and one paper with the results obtained from the experiments conducted on Sinhala and Tamil banking domain datasets. The published papers are listed below.

- A. Ignatius and U. Thayasivam. Speech embedding with segregation of paralinguistic information for tamil language: a survey. In *Tamil Internet Conference*, 2020.
- A. Ignatius and U. Thayasivam. A survey on paralinguistics in tamil speech process- ing. In *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*, pages 94–99. Association for Computational Linguistics, Apr. 2021.
- A. Ignatius and U. Thayasivam. Speaker-invariant speech-to-intent classification for low-resource languages. In *Speech and Computer*, pages 279–290. Springer Interna- tional Publishing, 2021.

1.4 Thesis Outline

In Chapter 2, we discuss paralinguistics and acoustic features. In Chapter 3, we review the work related to speaker normalisation techniques and self-supervised speech representations. We describe the methodology of the proposed approach in Chapter 4. We discuss the experimental setup and the details of the datasets that were used for the evaluation of the model in Chapter 5. In Chapter 6, we present a detailed analysis of the experimental results obtained. Finally, we conclude the thesis with ideas for future work in Chapter 7.

Chapter 2

Background

Speech signal carries multiple layers of information: linguistic information and paralinguistic information. Linguistic information conveys the content of the spoken utterance. Paralinguistic information laying beyond the verbal aspects of communication expresses speaker traits and states. Paralinguistic information is most commonly divided into three groups from long-term traits to short-term states as depicted in Figure 2.1. In the sections that follow, we describe paralinguistics, relevant applications, and acoustic features that represent the information present in speech signals.

2.1 Paralinguistics in Speech

Paralinguistics mainly deals with speaker traits and states including gender, age, cultural background, personality, emotional states, discrepant communication, and deviant speech. Biological trait primitives including sex/gender, height, weight, and age are long-lasting characteristics that are not intentionally changeable most of the time. Each individual has a unique voice because of the physiological variations such as differences in vocal tract sizes. Cultural factors which determine the native language and regional dialect also greatly influence speech.

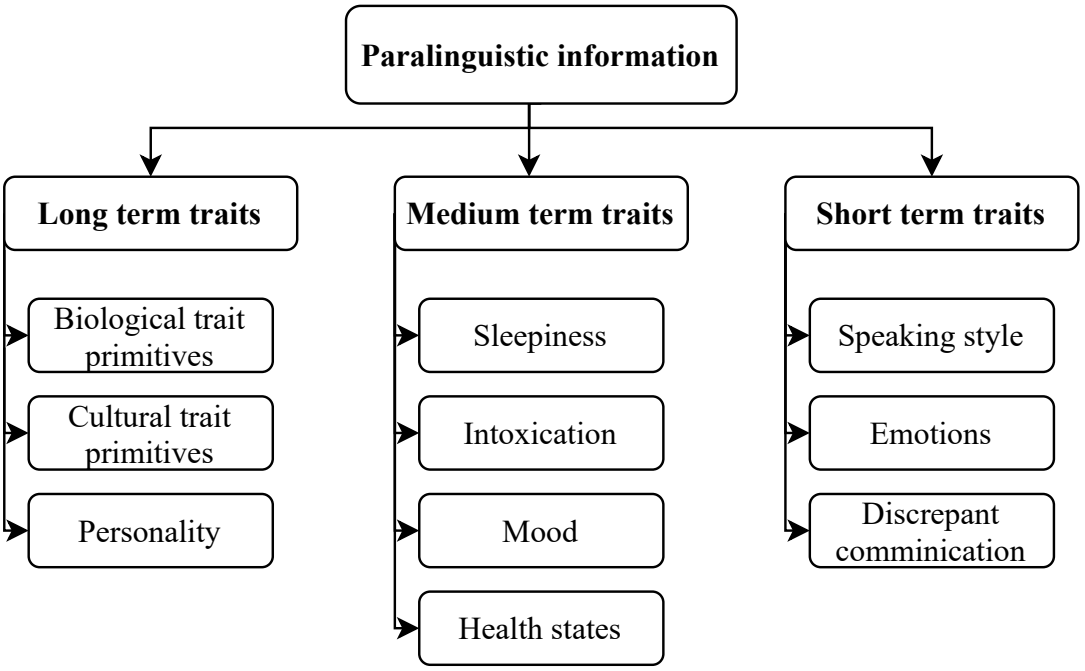

Figure 2.1: Types of paralinguistic information: Long-term traits, Medium-term between traits and states, and Short-term states

Pathological speech is a type of deviant Speech, which is usually due to long-term conditions caused by congenital malformations, or by diseases such as Parkinson's disease, motor neuron disease, Alzheimer's disease, and autism spectrum disorder (ASD). Medium-term states including intoxication by alcohol consumption and drowsiness do influence normal speech. Even if it can create a pathological impression, it returns to normal state before long. Non-native speech, usually considered as deviant speech consists errors in pronunciation and prosody. They need to be identified, investigated, and corrected. As opposed to deviant speech, the speaker deliberately chooses to use deceptive speech, indirect speech, deviant speech, irony, or sarcasm in discrepant speech or discrepant communication. Detection of pathological speech could provide an effective screening method for diseases that cause speech impairment whereas deceptive speech identification could be valuable in forensic and therapeutic applications.

Tone, volume, and speech rate are influenced by speaker's emotional condition. Emotions are short-term states and typical emotions include anger, fear, joy, surprise etc.

Acoustic profiles of each emotional state differ and some emotions share common characteristics. Personality, language, and culture influence and shape the expression of different emotions. There have been several studies on emotion recognition from speech to understand the actual intentions of the speaker. Identifying the speaker's emotional state helps in improving the performance in speech recognition systems by compensating for the effects of emotional expression.

Speaker recognition considers the combined traits of a speaker to identify an individual. Aim of the speaker recognition task is to assign the input speech segment to a particular class of speaker using speech features. Applications of speaker recognition include personal identification in forensics and speaker verification, which is used in access control systems. Speaker verification systems attempt to verify that the speaker belongs a particular group of people. Research on paralinguistics could be useful in a variety of speech recognition applications. Performance of ASR and SLU systems can be compromised by variabilities in the speech that occur due to speaker traits and states. Modeling the para-linguistic information contained in the speech signal helps in developing techniques to normalise speaker effects, thereby improving the performance.

2.2 Acoustic Features

Speech processing tasks make use of a number of features that describe the speech content. Commonly used acoustic low-level descriptors (LLDs) [22] include intensity, intonation, mel frequency cepstral coefficients (MFCCs), linear prediction cepstral coefficients (LPCCs), perceptual linear prediction cepstral coefficients (PLP-CCs), gammatone frequency cepstral coefficients (GFCCs), harmonicity, formants, vocal cord perturbation etc. The low-level descriptors aim to capture the useful information present in speech signal. Table 2.1 shows the breakdown of LLDs into three groups: prosodic, spectral, and voice quality features. The acoustic features are usually augmented with delta coefficients or regression coefficients which are derived from the raw LLDs.

Feature Type	Feature
Prosodic	Pitch Duration Energy
Spectral	MFCCs GFCCs LPCCs PLP-CCs Formants
Voice quality	Jitter Shimmer Harmonics to noise ratio Normalized amplitude quotient Quasi open quotient

Table 2.1: Acoustic features categorized into three types: prosodic, spectral, and voice quality features

MFCCs are among the most commonly used speech features in a wide range of speech processing tasks. The following steps are involved in the extraction of MFCCs. Speech signal is segmented into overlapping frames and a window function is applied. The windowed speech signal is then transformed into the frequency-domain using discrete Fourier transform. The magnitude spectrum is passed through mel filter bank and discrete cosine transform is applied on the log of the magnitude to convert the mel-spectrum into the cepstral domain. Lower order coefficients consist of phonetic information whereas the higher order coefficients describe more the speaker-specific information.

Formants, the resonant frequencies of the vocal tract change depending the speech content. The lower formants correlate highly with the phonetic content whereas the higher formants represent the speaker-specific information. Formants are usually estimated using LPCs. The fundamental frequency and formant frequencies are significant components of the speech signal and applications of formant detection include emotion recognition and gender separation [23].

Voice quality features such as jitter and shimmer are also useful in the field of speech

processing. Jitter and shimmer measure variations in the fundamental period and amplitude from one cycle to the next respectively. Jitter and shimmer measurements are relevant markers in detecting speaker's age and voice pathologies since they describe the pathological characteristics of the voice [24]. Derived features, for example the combinations of the afore-mentioned features can be computed. The most common features which capture the dynamics of the signal are first and second-order derivatives. Recent research studies focused on high-level acoustic features learnt using DNNs. High-level features can be generated from LLDs as well as from raw speech signals. DNN based speech representations have yielded great performance in a wide range of speech applications including speech recognition, speaker recognition and verification, spoken intent detection, and emotion recognition.

2.3 Summary

This chapter focused on the importance of multiple layers of information conveyed through speech signals in several speech applications. Linguistic information in speech covers the verbal content while paralinguistic information describes speaker characteristics. Paralinguistic aspects of communication affect speech in various ways as discussed earlier and result in performance degradation in speech recognition and speech-to-intent systems which requires only the linguistic content. To represent the underlying information present in speech, low-level descriptors, as well as DNN based speech representations that capture high-level features are used. The next chapter will discuss the related work on speech representations and how they are used in ASR and speech-to-intent systems.

Chapter 3

Literature Review

To alleviate the discrepancy between testing and training data, speaker information can be provided as an additional input to acoustic DNN models, making them invariant to speaker effects. Speaker embeddings are used to model speaker-specific information. Speaker embeddings represent long-term speaker characteristics which cover biological trait primitives including weight, height, age, and gender, cultural trait primitives, and personality traits. It is difficult for acoustic models to learn long-term speaker characteristics from short-term speech features. Augmenting with speaker embeddings enables transforming the acoustic feature space into a speaker-normalised one. The literature review focuses on speaker embedding, feature augmentation, and unsupervised speech representations.

3.1 Speaker Embedding

Speaker embedding encodes long-term speaker characteristics into a fixed dimensional vector. i-vector is the traditional approach which maps variable-length speech utterances to low dimensional representations. i-vector extraction involves Universal Background Model (UBM) and Total variability matrix. UBM, a speaker independent Gaussian Mixture Model (GMM) identifies and clusters similar acoustic features together. The total variability matrix represents the basis of the total variability space which models the speaker variability and channel variability subspaces in a common low dimensional space. UBM and the total variability matrix are learnt in an unsupervised manner using expectation maximization algorithm. High dimensional statistics from the UBM are then mapped to a low dimensional i-vector. i-vectors have been successfully employed in speaker identification and verification systems for many years. i-vectors have also been used to compensate for speaker effects in speech recognition systems.

DNN based embedding techniques can also be used for extracting speaker representations. In DNN based speaker embedding, speaker representations are extracted from the bottleneck layer of the DNN which is trained to discriminate between speakers. Among the recent DNN speaker embeddings, x-vectors extracted from a Time Delay Neural Network (TDNN) with a pooling layer [12] and h-vectors extracted using a hierarchical attention network [13] achieved better performance in speaker recognition systems. These vectors are not commonly used in ASR systems for normalising the speaker effects since there have been no consistent improvements in performance over i-vectors in the experiments.

3.2 Speaker Normalisation

Speaker normalisation using feature augmentation techniques involves incorporating the speaker level information extracted as a fixed dimensional vector in the input. This helps the DNN in compensating for the speaker effects, thus improving the performance. Feature augmentation can be performed by directly concatenating the speaker vectors to acoustic features or activations of a hidden layer. Speaker vectors transformed using separate neural networks can also be provided to the DNN during training.

In the conventional feature augmentation approach, i-vectors are provided in addition to the input speech features to the acoustic model. For each frame of a given utterance, the same utterance level i-vector is appended to the acoustic features. By providing speaker-level information at the input of the acoustic DNN, it learns to segregate speaker characteristics and generate speaker-invariant representations [14, 15]. In the study by Sri Garimella et al., speaker i-vectors are initially passed through a non-linear hidden layer and then combined with the acoustic features [16]. In this model, connectivity from to the rest of network i-vectors is restricted to improve the robustness of the model. Transforming i-vectors using a non-linear hidden layer showed improvement over direct concatenation of the i-vectors with the acoustic features.

Miao et al. presents two i-vector transformation networks namely ivecNN and adaptNN. These networks attempt to project the input speech features to a speaker-normalized space using i-vectors as additional inputs [17]. A bias vector estimated using ivecNN is added to the input acoustic features resulting in a speaker-invariant feature space. ivecNN generates a feature shift when i-vectors are given as the input. The output of the network is added to the acoustic features and fed to the DNN speech model. Alternatively below the original DNN speech model, adaptNN employs multiple adaptation layers. With acoustic features provided as the input, all the adaptation layers until the

output layer concatenate i-vector with its output. Adaptation layers convert the original acoustic features into more speaker-invariant features using i-vector integration. Experimental outcomes indicate that these two approaches outperformed the original DNN without i-vectors.

Xiangang Li et al. investigated the idea of augmenting speech features with d-vectors, a speaker embedding learnt from long short-term memory (LSTM) networks which is a type of recurrent neural networks (RNNs) [25]. They proposed cross-LSTMP to conduct speaker recognition and speech recognition simultaneously. cross-LSTMP consists of two LSTMs, senone-LSTM for classifying senones and speaker-LSTM for classifying speakers. In each frame, previous activations of speaker-LSTM are fed into the senone-LSTM while the previous activations of senone-LSTM are into the speaker-LSTM to make the network speaker aware. Conducted experiments showed that the proposed approach can be effectively used to enhance the performance by compensating for the speaker invariability in ASR.

In the paper by Xiaodong Cui et al., a speaker adaptive training approach based on speaker embeddings [18] aimed at creating a canonical feature space is presented. This approach uses a control network to map i-vectors to layer-dependent elementwise affine transformations. The speaker-dependent affine transformations are applied to the outputs of the main model's selected hidden layers to normalise speaker effects. This approach outperformed DNNs with input acoustic features augmented using i-vectors.

Zhiyun Fan et al. proposes a speech transformer [26] with a speaker attention module called Speaker-Aware Speech Transformer [27]. The speaker attention module comprises a multi-head attention layer and a speaker knowledge block containing i-vectors belonging to a group of speakers from the training data. It generates a soft speaker embedding which is computed as a weighted combination of a set of i-vectors for a given acoustic feature input. Since there can be similarities across different speakers, a speaker vector can be expressed as a linear combination of a set of speaker vectors.

Given an acoustic feature input, the speaker attention module computes the similarity between the acoustic feature input and every i-vector from the speaker knowledge block using attention mechanism to derive the combination weight corresponding to every i-vector in the speaker knowledge block and generate the soft speaker embedding. The encoder output is concatenated with the soft speaker embedding and fed to decoder. Soft speaker embedding allows the model to compensate for the speaker variability and generalize better to unseen speakers. Similar to this approach, Jia Pan et al. proposes selecting the relevant i-vector from the memory for each frame using attention mechanism [28]. Speaker-aware speech transformer has been shown to be more effective when compared to other i-vector based feature augmentation techniques.

3.3 Disentangled Speech Representations with Unsupervised Learning

The unsupervised learning method makes use of a great amount of untranscribed speech samples to learn meaningful feature representations. These representations could be very useful in many low-resource speech tasks. Autoencoders are one of those networks involving the reconstruction of their inputs. The encoding network of the autoencoders extracts a latent representation capturing as much useful information as possible and the decoding network attempts to recover the original input from the latent representation. Autoencoders are aimed at learning a desired latent representation by imposing constraints. The latent representation attempts to eliminate irrelevant speaker-specific information while preserving as much of the information necessary for a perfect reconstruction.

Wei-Ning Hsu et al. proposes a Factorized Hierarchical Variational Autoencoder (FHVAE) [29] to learn disentangled representations from sequential data in an unsupervised

manner. They have focused on segregating the linguistic content and speaker characteristics in speech signal by manipulating a set of latent variables. Speaker characteristics affect the speech features at the sequence-level and the phonetic content impacts the speech features at the segment-level. The sequence-level attributes show smaller variations within an utterance while segment-level attributes show similar variations within an utterance and across utterances. This study demonstrates the ability of FHVAE to factorize the segment-level and sequence-level attributes of speech into different sets of latent variables [30, 31]. Hence, the two types of attributes are encoded into latent segment variables and latent sequence variables. Latent variable that encodes the linguistic content while discarding the speaker characteristics can be used for building speech recognition systems robust to speaker variations.

In the paper by Jan Chorowski et al., extraction of speech representations in an unsupervised manner using a Vector Quantized Variational Autoencoder (VQ-VAE) is explored. The VQ-VAE is aimed at learning a speech representation that captures only high-level semantic content whilst removing the speaker-level information in the signal[32]. It conditions the decoder on speaker identity which enables the encoder to focus only on capturing the semantic content, thus resulting in a more speaker-invariant representation. Best representations are obtained when using MFCC features as inputs and raw waveforms as targets. Waveforms are reconstructed using a WaveNet decoder that combines auto-regressive information about past wave samples, speaker information, and latent information extracted by the encoder. VQ-VAE achieved a better separation between phonetic content and speaker information, performing greater in the phonetic unit classification task.

Yu-An Chung et al. proposed a novel unsupervised model, Auto-regressive Predictive Coding (APC) that is trained for predicting the spectrum of the future frame which is several steps ahead of the input frame [3, 20, 21]. It is focused more on inferring more global structures present in speech than making use of the local smoothness in the speech signal. APC can be utilised as a pre-training strategy to learn meaningful

and transferable speech representations. Unlike the previously discussed research studies which attempted to discard the irrelevant information, APC model aims to preserve as much information such that the extracted representations could be used for a variety of downstream tasks. Experiments performed using intermediate representations extracted at different layers of the APC model, with different speech applications such as speaker verification, phone classification, and automatic speech recognition indicate that the model captures different types of speech information at different levels. Lower layers focus on capturing more speaker-specific information whereas the upper layers try to capture richer phonetic information. Internal representations derived from different layers can be combined to learn disentangled speech representations which could be beneficial in speech recognition tasks.

APC as well as other self-supervised acoustic features including as wav2vec [33], Tera [34], audio Albert [35], Mockingjay [36], and NPC [4] have been used in various speech tasks including phoneme classification, ASR, speaker identification and verification, speech-to-text translation, and SLU. SLU experiments conducted by Edmilson et al. demonstrated that wav2vec outperforms filterbank features [37] in prediction of intent and entity labels. The outcomes of the research studies discussed above show the efficacy of applying unsupervised learning methods to a larger amount of untranscribed speech samples to generate speaker normalised speech representations which could be helpful in building robust speech recognition systems under low-resource settings.

3.4 Spoken Intent Detection

Recent studies have worked on RNN and transformer based on end-to-end models for spoken intent classification [38, 39]. Pre-training strategies have been used in different ways to improve the performance. Loren Lugosch et al. [2] proposes an end-to-end

SLU model which uses a pre-training approach. The model is initially trained for predicting words and phonemes, and the meaningful feature representations learnt from the pre-trained part of the model are used as input features for the SLU task. The use of self-supervised speech representations, pre-trained model initialisation, and multi-task learning have been explored in the paper by E. Morais et al.[37]. wav2vec [33] features outperformed filterbank features in their experiments for prediction of intent and entity labels. Combination of wav2vec features with multi-task learning involving speech-to-text and text-semantic tasks further improved the performance.

Yohan Karunanayake et al. [7, 40] make use of character probability and phoneme probability features generated from pre-trained ASR models to perform intent prediction for Sinhala and Tamil Languages under low resource settings. Natalia Tomashenko et al. [19] investigate speaker adaptive training in end-to-end SLU models which has not been addressed in other research studies. They investigate the effectiveness of speaker adaptation with the integration of i-vectors into the end-to-end model. To improve the performance, they use transfer learning approaches with ASR systems.

3.5 Summary

Disentangling the linguistically irrelevant information is important to alleviate the problem of mismatch between testing and training conditions in speech processing applications. The literature review discussed the existing solutions under two categories: speaker embedding based feature augmentation and unsupervised speech representation learning. Incorporating the speaker information into the acoustic model through speaker embedding has resulted in improved recognition but such supervised acoustic models require a great amount of labeled speech samples. In the case of low-resource settings, unsupervised learning method can be adopted where speech representations learnt from large-scale unlabeled data are used in downstream tasks with a limited amount of labeled training data. Notable results observed in previous studies on ASR

and SLU systems indicate that these techniques can be applied in low resource speech-to-intent classification systems.

Chapter 4

Methodology

As discussed in the literature review, feature augmentation using i-vectors has been shown to be effective in learning speaker-invariant speech features. Furthermore, self-supervised speech features have been successfully used in many speech processing applications. Therefore, we propose using a transfer-learning approach, in which we pre-train an acoustic model using high-resource speech data and then extract intermediate speech representations from the pre-trained model with low-resource speech data.

These intermediate representations are augmented with utterance level i-vectors. Feature augmentation using i-vectors helps in normalising the speaker effects. Before combining with the intermediate speech features, a feature mapping network is used to transform the i-vectors. The combined feature vector is then fed to the intent classifier consisting of multiple fully connected layers, which predicts the intent for a given speech utterance. The architecture of the proposed speech-to-intent model is depicted in Figure 4.1.

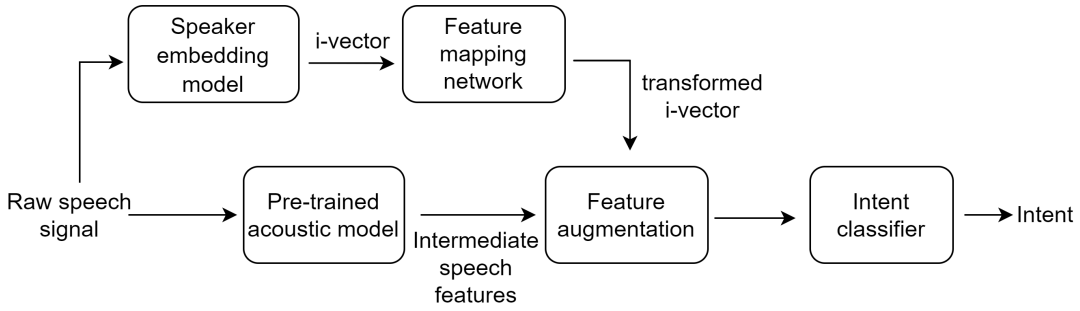

Figure 4.1: Architecture of the proposed model: Intent classifier takes i-vectors combined with intermediate speech features as the inputs

4.1 Pre-trained Model

Pre-training strategy is used since only a limited amount of labeled speech data is available for speech intent classification. Pre-trained models can be obtained from ASR systems or self-supervised learning networks. ASR models can be trained using various targets including phonemes, graphemes, wordpieces, or words. Huge amounts of available unlabeled speech data cannot be used by the speech recognition models since they require audio transcriptions. Self-supervised speech representation learning method make use of unlabeled speech data to generate features capturing high-level properties.

As discussed in Chapter 3, self-supervised features such as APC [3, 20, 21], wav2vec [33], Mockingjay [36], Tera [34], audio Albert [35], and NPC [4] have shown promising results. They have been applied in a variety of speech tasks including phoneme recognition, ASR, speaker recognition, SLU, and speech-to-text translation. Thus, we exploit pre-trained acoustic models to extract meaningful speech representations that can be used for intent classification task.

4.2 Feature Augmentation

In feature augmentation, we propose to use i-vectors as additional input features to the DNN model. i-vectors encode speaker characteristics and we want our model to be invariant to speaker variations. We selected i-vectors for feature augmentation as previous studies did not observe consistent improvements in performance of ASR systems with DNN based speaker representations when compared to i-vectors. Passing the i-vectors through multiple non-linear layers before combing with intermediate speech representations proved to be more effective than directly appending the i-vectors [16, 17]. Thus, we apply feature transformation on i-vectors and combine them with acoustic features. Appending the bias vector estimated from the i-vector network to the intermediate speech features, makes the resulting feature space speaker-invariant. Feature representations from pre-trained acoustic models are passed through multiple Gated Recurrent Unit (GRU) layers before combining them with the bias vector. We use an i-vector neural network to map the utterance-level i-vector to bias vector. The i-vector network contains multiple dense layers with sigmoid activation and an output dense layer with rectified linear activation. It generates a features shift when i-vectors are given as the input. i-vectors together with intermediate speech representations form the augmented feature vector.

4.3 Summary

The proposed model utilises the knowledge learnt from pre-trained models and i-vector based feature augmentation to generate speaker-invariant speech representations under low-resource settings. The previous studies have shown the effectiveness of combining i-vectors with speech features in ASR and SLU systems. By making the model invariant to speaker effects, we aim to build a robust speech-to-intent system. The next chapter will present the experimental setup and the datasets used for the evaluation of the proposed approach.

Chapter 5

Experiments

We designed the experimental set up so as to understand the role of pre-trained speech models and i-vector based feature augmentation in low-resource speech-to-intent classification models. In our experiments, we investigated four different feature representations listed below. We experimented with two ASR models as used by Yohan et al. in [7, 40] and two self-supervised speech models. Self-supervised features have been proven to be successful in SLU tasks. These models were initially pre-trained on English speech corpus and then used to derive intermediate speech representations from speech intent data.

1. ASR Intermediate Representations
    - Character probability representation
    - Phoneme probability representation
2. Self-supervised Features
    - Autoregressive Predictive Coding (APC)
    - Non-autoregressive predictive Coding (NPC)

Intent classifier models were built using the above intermediate features and i-vectors as inputs. The performance of the models using intermediate representations combined with i-vectors was compared to that of the models using only the intermediate speech features. The SIDEKIT toolkit [41] was used to extract i-vectors at the utterance level. i-vector extractor from the toolkit takes MFCCs as input features and generates a 100-dimensional i-vector.

5.1 ASR Intermediate Representations

As presented in [7, 40], we used the DeepSpeech model [1, 42] pre-trained on the Common Voice English corpus [43] for extracting the character probability representation. The Deepspeech model consists of multiple convolutional layers, recurrent layers, and fully connected layers. Given a sequence of acoustic features, Deepspeech tries to predict the text transcription. The input sequence of MFCCs is initially converted into a sequence of character probabilities. From the character probabilities, the output transcript is generated using beam search algorithm. Figure 5.1 shows the architecture of the Deepspeech model used for the pre-training approach.

We employed the end-to-end SLU model proposed by Loren Lugosch et al. [2] to obtain the phoneme probability representation. As shown in Figure 5.2, the model is first pre-trained on the LibriSpeech ASR corpus [44] for predicting words and phonemes. Then the feature representations extracted from the word and phoneme classifiers are used to train the end-to-end SLU model. The phoneme module consists of a SincNet layer, multiple convolutional layers, and recurrent layers with pooling and dropout. It is trained to predict phonemes from the raw speech signal. The word module takes the output from phoneme module and uses recurrent layers with dropout and pooling, to predict word targets with a linear classifier. We selected the intermediate representations from the phoneme module, the phoneme probabilities to extract the input feature representations for the intent classification task.

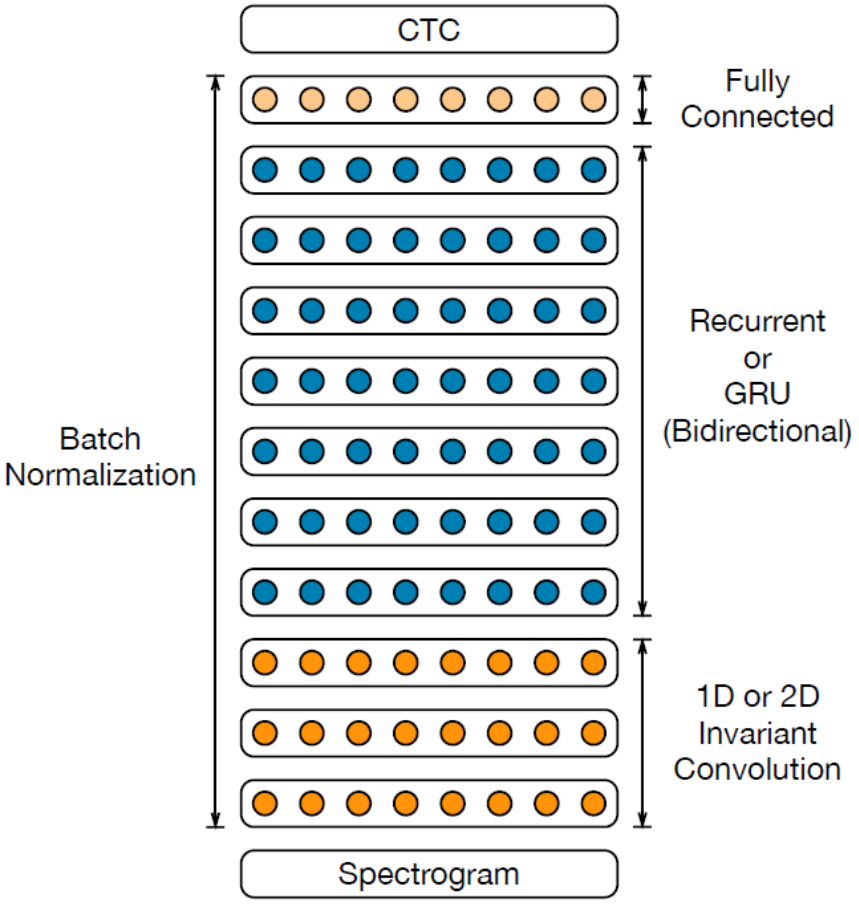

Figure 5.1: Architecture of the deepspeech system[1]

5.2 Self-supervised Features

As described in Chapter 3, APC and NPC features are learnt from self-supervised speech models. APC incorporates an autoregressive model which attempts to predict the spectrum of the future frame which is a number of steps away from the current frame whereas NPC is aimed at predicting the input in a non-autoregressive manner, depending only on the neighbor frames within the receptive field. We used the APC and NPC models that had been initially trained on the LibriSpeech ASR corpus for extracting the self-supervised speech representations.

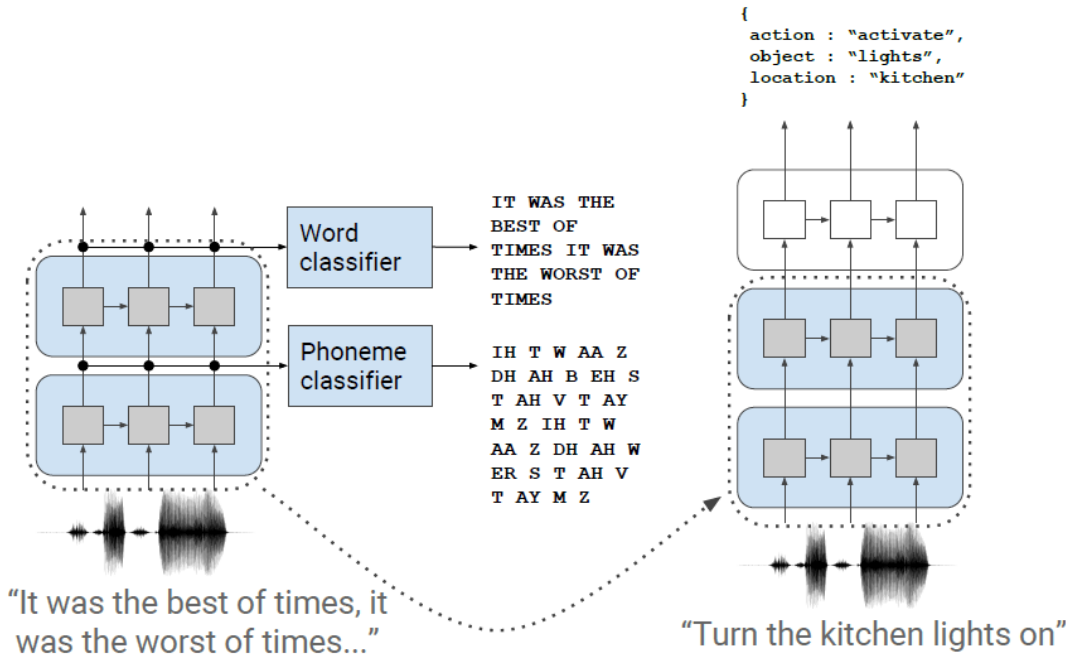

Figure 5.2: Architecture of the end-to-end SLU model; The lower layers of the model are pre-trained using ASR targets. The features from the pre-trained part are used as the input to the next module.[2]

For a given input sequence of speech features (x_1, x_2, ... , x_t), APC uses the autoregressive property to predict the spectrum of the future frame x_{t+n} which is n steps away from the current frame x_t. For APCs to focus on inferring more global structures than the local information in the speech signal, the model is trained for predicting a future frame that is n steps away from the current frame. Number of steps to the target frame controls what is learned in the representation. Figure 5.3 illustrates the architecture of the Vector Quantized Autoregressive Predictive Coding (VQAPC) model, which includes additional quantization layers to force the model to retain information needed for attaining maximal prediction. The amount of information flowing from one layer to the next is controlled by the codebook size and the code dimension of a VQ layer. The capacity of the models can be explicitly controlled by varying these two factors. We took the output from the last layer to use as the input speech representations in the target model.

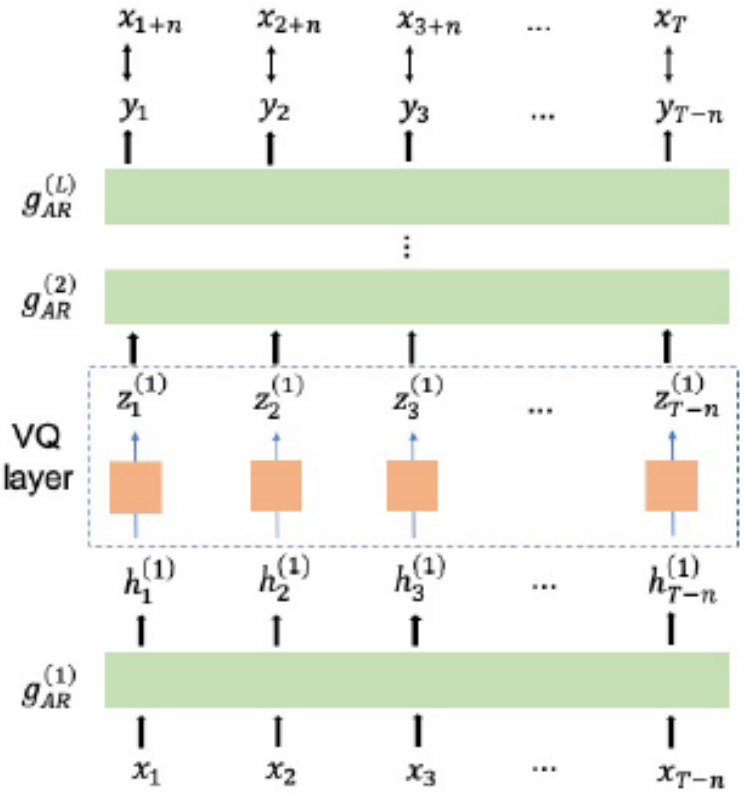

Figure 5.3: Architecture of the VQ-APC model; $g_{AR}^{(l)}$ denotes l_{th} unidirectional GRU layer with 512 hidden units[3]

NPC model with stacked convolutional blocks (ConvBlocks) learns representations in a non-autoregressive manner, observing only the local dependency of speech. Figure 5.4 shows an illustration of NPC at time t with desired input mask size $M_{in} = 3$ and receptive field size R = 13. Input frames are processed by ConvBlock layers and the features from Masked ConvBlock at each layer are summed up to the context representation h_t. It is then passed into Vector Quantization layer followed by a linear projection predicting y_t to match the target frame x_t.

5.3 Extraction of i-vectors

An i-vector is a fixed-dimensional representation of a variable-length speech segment aimed at capturing the speaker-level information present in the speech signal. Initially

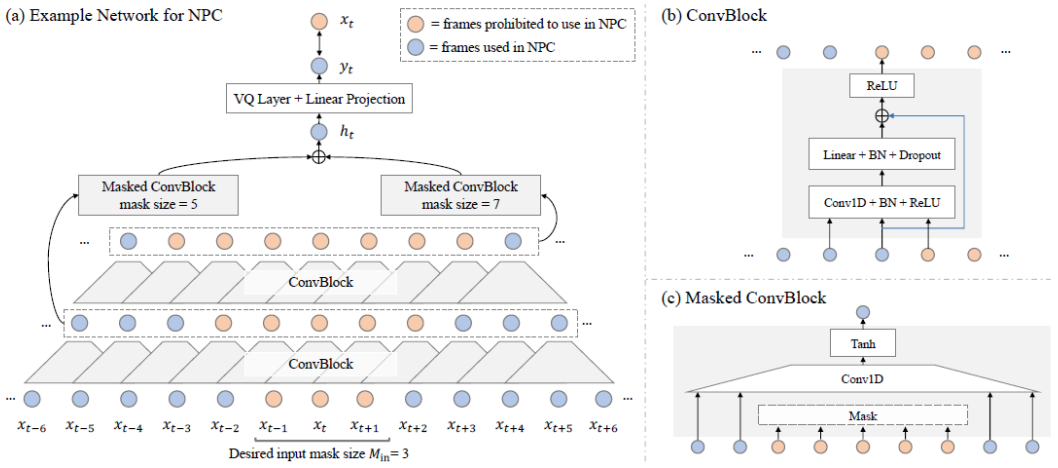

Figure 5.4: Architecture of the NPC network; orange nodes represent the frames that contain information of the target frame x_t, not used for prediction.[4]

developed for speaker identification and verification systems, i-vectors have now been used in different speech processing applications. By using i-vectors as additional inputs along with the acoustic features in our model, we intend to make the DNN learn to compensate for the speaker variations.

We used SIDEKIT toolkit [41] to extract 100-dimensional i-vectors with MFCCs as input features. An illustration of i-vector extraction system is shown in Figure 5.5.

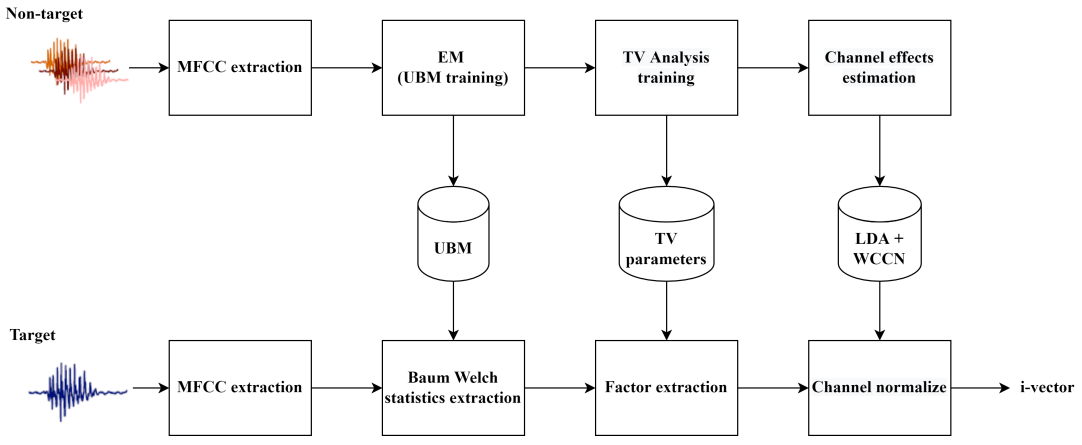

Figure 5.5: i-vector extraction system: for channel effects compensation, Linear Discriminant Analysis (LDA) and Within Class Covariance Normalization (WCCN) are used

MFCC feature extraction involves the steps shown in Figure 5.6. The speech signal is segmented into overlapping frames and a window function (typically Hamming window) is applied to each frame. Discrete Fourier Transform (DFT) is then used to transform the windowed speech signal to frequency-domain. The magnitude transform is passed through mel filters and p-dimensional filter bank coefficients are obtained. Logarithmic operation is performed to compress the dynamic range of the signal and Discrete Cosine Transform (DCT) is applied to obtain the Mel frequency cepstral coefficients.

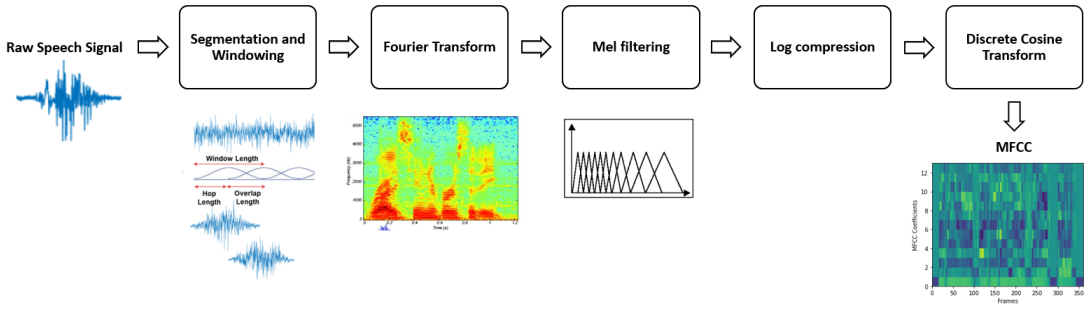

Figure 5.6: Steps involved in the MFCC feature extraction: windowing the signal, applying the DFT, passing through mel-filter bank, applying the DCT to the mel spectrum in log scale

i-vector extraction consists of two stages: Universal Background Model (UBM) state alignment and i-vector computation. The Universal Background Model (UBM) is a speaker-independent GMM model, which is trained on the speech segments from many speakers using expectation maximization (EM) algorithm.

A given speaker and channel-dependent Gaussian Mixture Model (GMM) supervector M can be modeled as shown below:

$$M = m + Tw$$

where

M - speaker and channel-dependent supervector

m - speaker and channel-independent supervector

T - total variability matrix

w - i-vector

The UBM state alignment involves identifying and clustering the similar acoustic content which allows the i-vector estimation to be less impacted by the phonetic variations within the features. The total variability matrix represents the basis of the total variability space which models speaker variability and channel variability subspaces in a common low dimensional space. UBM and the total variability matrix are learnt in an unsupervised manner using EM algorithm. High-dimensional statistics from the UBM are then mapped to a low-dimensional i-vector.

5.4 Datasets

We used a speech intent dataset in Sinhala and Tamil languages [45] to evaluate the performance of the proposed approach. This dataset contains crowd-sourced audio recordings in Sinhala and Tamil languages which map to six unique intents used in the banking domain. It consists of 7624 samples in Sinhala language, which is 7.5 hours of audio recordings from 215 speakers and 400 samples in Tamil language, which is 0.5 hours of audio recordings from 40 speakers. Each speech utterance is less than 7 seconds long. The dataset information is shown in Table 5.1.

Six different queries commonly used by customers when interacting with the bank staff are used as the set of intents for creating a banking domain speech dataset. Each intent can be communicated through speech in different ways and these varying verbal descriptions are called “inflections” of an intent.

Table 5.1: Details of the Banking Domain Dataset (Sinhala and Tamil): Number of inflections (I) and Number of samples (S) per intent are listed

Intent	Sinhala		Tamil	
	I	S	I	S
Request Acc. balance	8	1712	7	101
Money deposit	7	1306	7	75
Money withdraw	8	1548	5	62
Bill payments	5	1004	4	46
Money transfer	7	1271	4	49
Credit card payments	4	795	4	67
Total	39	7624	31	400
Size in hours	7.5		0.5	

We also used Fluent speech command (FSC) dataset for evaluation. It consists of 30,043 speech recordings in English recorded by 97 speakers. There is a total of 248 phrases mapping to 31 unique intents. Each recording in the dataset contains an utterance used for controlling smart-home appliances, for example, ”put on the music”.

We collected a new dataset in healthcare domain for use in further evaluation. It consists of Tamil speech recordings from 100 speakers that describe a set of symptoms. We chose 16 general symptoms associated with different conditions and asked the speakers to read predefined sentences describing the corresponding symptoms. The target application of this dataset is to predict the symptom the speaker intends to convey. Details of the dataset is presented in Table 5.1.

We initially identified the common medical symptoms and selected 16 of them as the intents. For a given symptom, we used different verbal descriptions to create 10 inflections. For example, a speaker might say ”I have a dull ache in my head” to describe a headache. Speech recordings were collected via crowd-sourcing using a web application. The speaker was first asked to fill in his/her details (name, age, and gender) before recording the speech utterances, to uniquely identify each speaker. To record a speech utterance, a randomly selected description of a symptom was displayed and the speaker was instructed to read the text phrase in his/her own language. The captured

Table 5.2: Details of the Healthcare Domain Dataset: Number of inflections and Number of samples per symptom are listed

Intent	No. of inflections	No. of samples
Cough	10	97
Blurry vision	10	89
Acne	10	75
Headache	10	102
Chest pain	10	86
Neck pain	10	86
Knee pain	10	100
Shoulder pain	10	87
Hard to breathe	10	78
Skin issue	10	84
Foot ache	10	103
Joint pain	10	88
Stomach ache	10	96
Back pain	10	98
Hair fall	10	91
Ear ache	10	93
Total	160	1453

audio recordings were then validated and the dataset was cleaned. This dataset can be used to train conversational agents in the healthcare domain.

5.5 Summary

In this chapter, we have described the experimental setup where we use four different speech representations namely character probability representation, phoneme probability representation, APC, and NPC to analyse the performance in each model. We employed i-vector based feature augmentation technique to compensate for the speaker variability. We compared the performance between models with intermediate speech representations as the only inputs and models with augmented features. We used Sinhala and Tamil banking domain dataset and Fluent speech command dataset to evaluate the proposed model and created a Healthcare domain dataset in Tamil language for future use as well.

Chapter 6

Results and Analysis

We analysed the performance of the intent classification with four feature representations and their corresponding i-vector augmented feature representations. Initially, we split each dataset into training and testing sets with a train-test split ratio of 4:1. We then employed 5-fold cross-validation and obtained the average classification accuracy on the test sets.

6.1 Comparison between results obtained with and without i-vector normalisation

The results of our experiments on each dataset are summarised in Table 6.1. It compares the average accuracy values obtained on the test set for the four feature representations. Also, under each experiment, results obtained when using i-vectors for speaker normalisation are shown.

From the results presented in Table 6.1, the following observations can be made. i-vector based speaker normalisation technique was able to slightly increase the classification accuracy of the intent classification model in all four experiments. Despite the fact that the improvement is minor, it can be clearly seen in all four models. These

Table 6.1: Average accuracy values obtained for 4 different intent classification models trained with Sinhala and Tamil speech dataset

Dataset	Character Prob		Phoneme Prob		APC		NPC	
	without i-vectors	with i-vectors	without i-vectors	with i-vectors	without i-vectors	with i-vectors	without i-vectors	with i-vectors
Sinhala speech intent dataset	0.81	0.83	0.92	0.93	0.97	**0.98**	0.93	0.94
Tamil speech intent dataset	0.32	0.44	0.49	**0.51**	0.25	0.43	0.25	0.46
Fluent speech command dataset	0.41	0.46	0.79	**0.81**	0.61	0.62	0.67	0.71

results show that the proposed feature augmentation method is more effective than the previous approaches in spoken intent prediction. For the Sinhala dataset, NPC features augmented with the i-vectors yielded the highest accuracy of 98 %. This model outperformed the approach presented in the previous study [7]. It indicates that the use of pre-trained self-supervised speech models contributes to improved performance as well. Phoneme probability features, on the other hand, achieved the best performance for Tamil and English datasets with the accuracy values of 51 % and 81 % respectively.

Plot of average test accuracy values from the the four experiments conducted on the Sinhala speech dataset is shown on Figure 6.1. The standard deviation obtained with 5-fold cross validation is marked with error bar. As we can see in the figure, when acoustic features are augmented using i-vectors, there is an increase in accuracy with little variance .

In Table 6.2, we present the detailed results from the experiments with Sinhala dataset. The table shows the intent-wise F1-score values obtained for the four intent classification models and their respective comparison when speaker normalisation technique is applied. APC features outperforms the other three features and all the classes achieved an F1-score of over 0.96. The highest F1-score of 0.99 is achieved by Type 1 intent (Request Acc. balance). Even with a smaller number of training samples, type 6 intent (Credit card payments) achieved an f1-score of 0.97 . The efficacy of our proposed approach of applying i-vector based speaker normalisation to self-supervised speech representations in speech-to-intent systems is clearly demonstrated by these results.

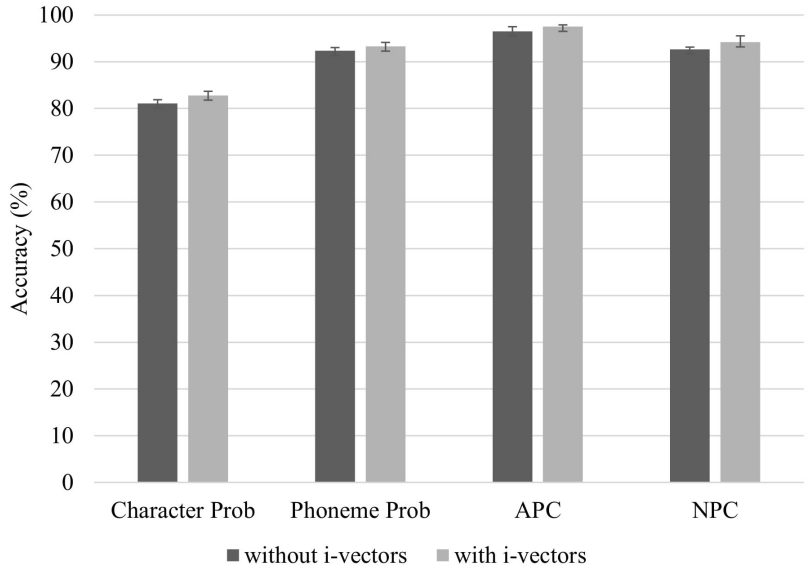

Figure 6.1: Plot of average accuracy values for the four speech-to-intent models with standard deviation (Sinhala dataset)

Table 6.2: Comparison of F1 scores obtained for each intent in Sinhala speech dataset

Intent	Character Prob		Phoneme Prob		APC		NPC	
	without i-vectors	with i-vectors	without i-vectors	with i-vectors	without i-vectors	with i-vectors	without i-vectors	with i-vectors
Request Acc. balance	0.91	0.95	0.98	0.98	0.99	0.99	0.95	0.97
Money deposit	0.77	0.77	0.89	0.92	0.96	0.97	0.94	0.93
Money withdraw	0.76	0.79	0.87	0.90	0.96	0.97	0.91	0.90
Bill payments	0.74	0.78	0.90	0.91	0.94	0.96	0.90	0.89
Money transfer	0.86	0.88	0.95	0.97	0.99	0.98	0.96	0.98
Credit card payments	0.81	0.77	0.91	0.89	0.96	0.97	0.92	0.93
Average	**0.810**	**0.824**	**0.917**	**0.930**	**0.966**	**0.974**	**0.930**	**0.934**

Table 6.4 show the confusion matrices for the best performing model without and with i-vectors for Sinhala dataset. With the incorporation of i-vectors, the number of correct predictions increases for the the first 4 intents and largest improvement is observed for the Type 3 intent where the number of correct predictions increases by 4.8 %. However, the number of correctly classified speech utterances slightly decrease for the last two intents. Further investigation is needed to identify the reason behind this.

Table 6.3: Confusion matrix for the model using APC features without i-vectors - Sinhala dataset

1	**337**	0	2	1	1	1
2	0	**250**	3	4	1	2
3	3	8	**291**	4	0	4
4	2	3	1	**189**	0	5
5	0	2	0	0	**252**	0
6	0	0	0	2	0	**157**
	1	2	3	4	5	6

Table 6.4: Confusion matrix for the model using APC features with i-vectors - Sinhala dataset

1	**337**	0	2	2	1	0
2	1	**251**	5	2	1	0
3	0	2	**305**	3	0	0
4	1	0	4	**195**	0	0
5	1	0	1	0	**246**	6
6	0	1	0	4	0	**154**
	1	2	3	4	5	6

6.2 Performance under different sizes of training data

In addition to the experiments mentioned above, we analysed the performance of the best performing model in the three datasets by varying the training data size. The plots of accuracy values obtained on Sinhala, Tamil, and English datasets with varying sizes of training data are illustrated in Figure 6.2, Figure 6.3, and Figure 6.4 respectively.

From these plots, we can see that the classification accuracy obtained in the experiment with i-vector based speaker normalisation is always higher than that in the experiment without i-vector based speaker normalisation irrespective of the size of the training data. Furthermore, a large increase in accuracy values obtained with a small number of training samples clearly shows that the incorporation of i-vector based feature augmentation technique is more helpful under low-resource conditions.

6.3 Discussion

In our experiments, we investigated the effectiveness of i-vector based speaker normalisation in enhancing the performance of speech-to-intent classification models. We used a transfer learning approach with various pre-trained speech models to exploit the

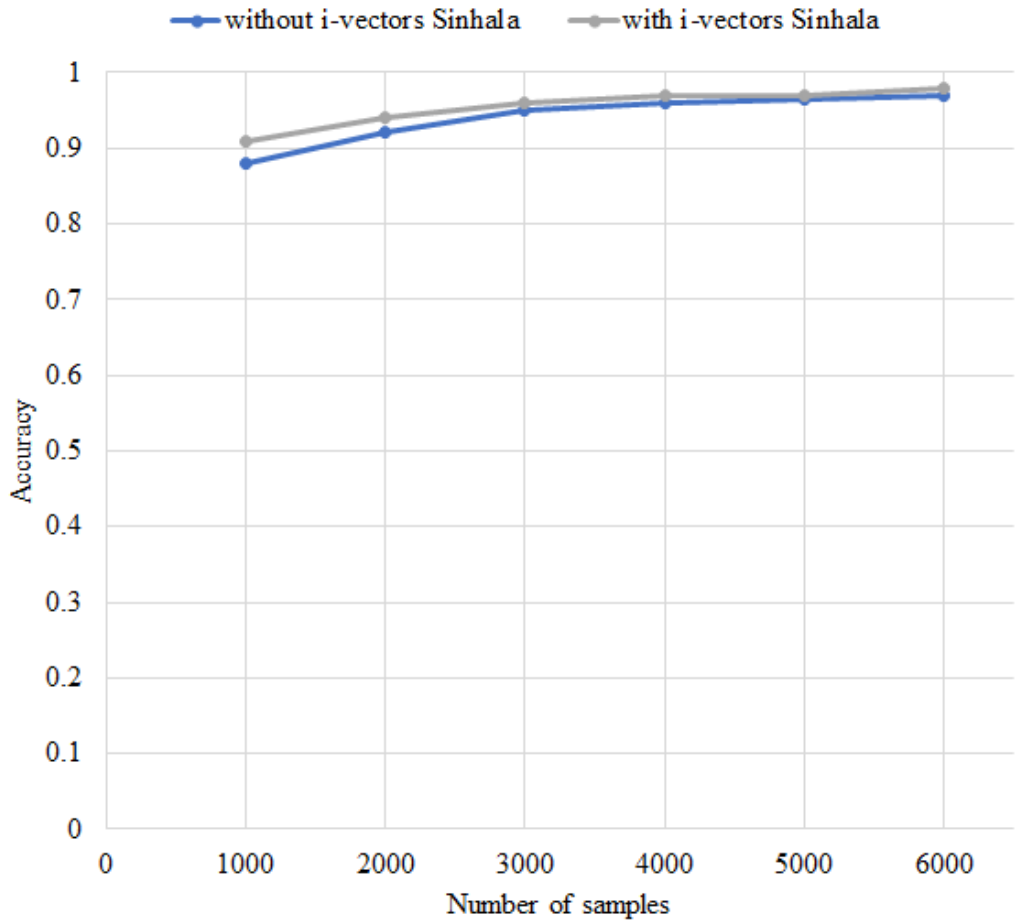

Figure 6.2: Plot showing the variation of accuracy with the size of the training data when APC features are used in the Sinhala speech-to-intent model

knowledge learnt from ASR and self-supervised models. We can see that the models with i-vectors showed better results than the conventional ones without i-vectors. APC features achieved the best performance for Sinhala dataset (98 %) while the phoneme probability features achieved the best performance for Tamil (51 %) and English (81 %) datasets. The performance of the proposed model for Tamil dataset was not significant since the dataset size is small. The Tamil dataset consists of only 400 samples which were not enough to achieve better performance. However, even with a small number of training samples, we were able to observe improvement in performance when i-vector based speaker normalisation technique was used.

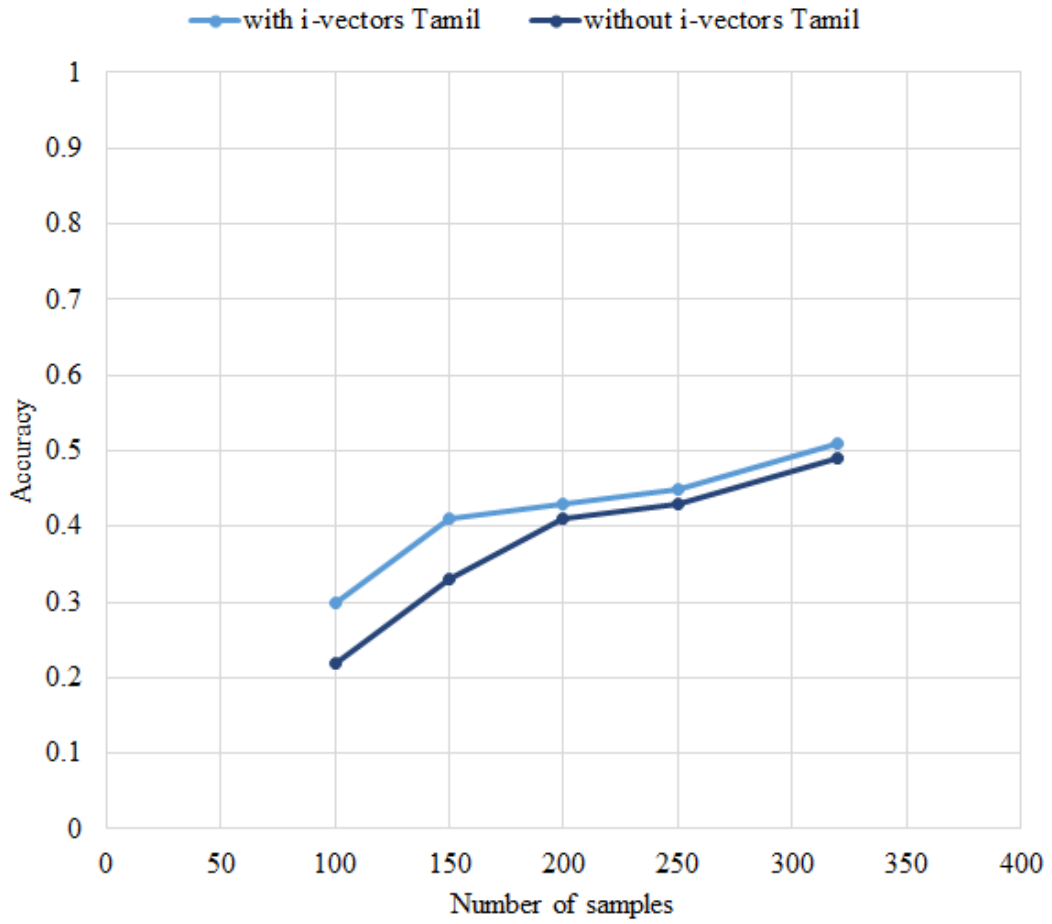

Figure 6.3: Plot showing the variation of accuracy with the size of the training data when phoneme probability features are used in the Tamil speech-to-intent model

The proposed Sinhala speech-to-intent model outperforming the previous work [7] shows the effectiveness of self-supervised features combined with i-vectors. Since we were able to observe only a slight improvement with the i-vector based speaker normalisation technique, we carried out experiments with varying training data sizes. The experimental results proved that our approach is more beneficial when there is only a small number of training samples. Thus, it has to be further evaluated on datasets from different domains and different languages to prove the generalizability of the model.

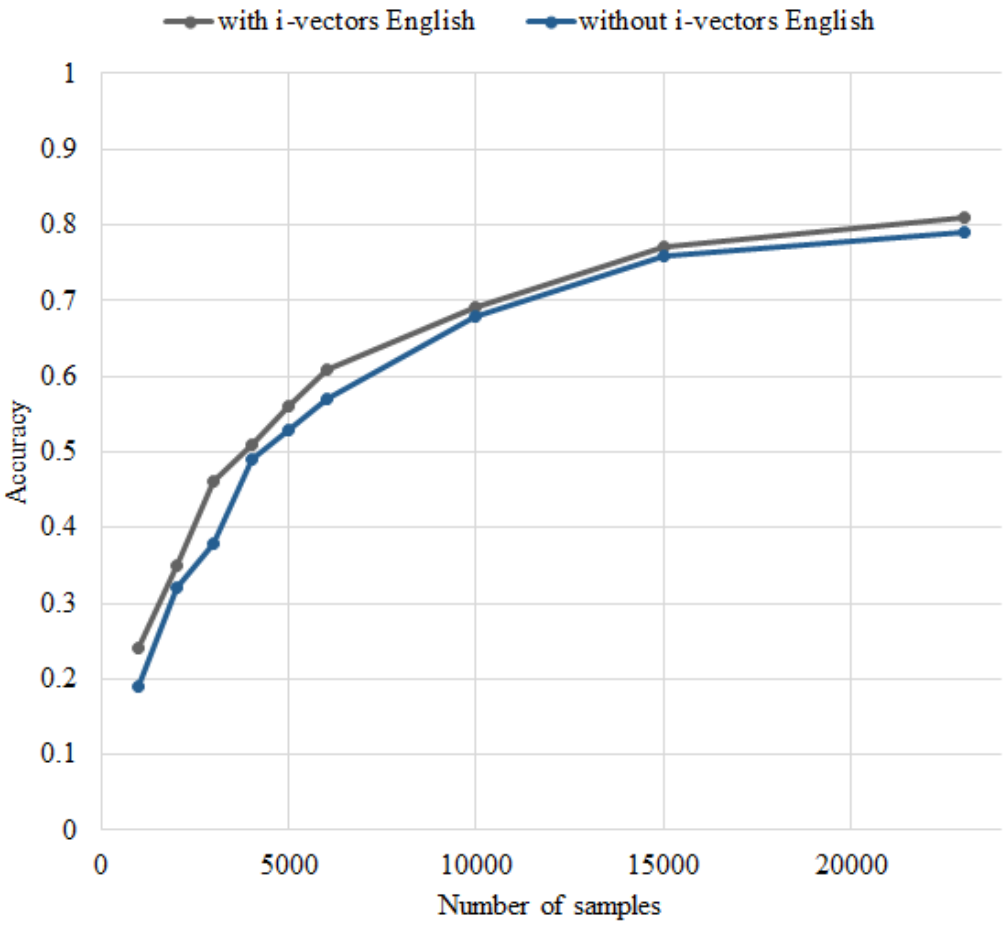

Figure 6.4: Plot showing the variation of accuracy with the size of the training data when phoneme probability features are used in the English speech-to-intent model

Chapter 7

Conclusion and Future work

In this study, we presented an speaker-invariant speech-to-intent model using i-vector based speaker normalisation technique. Speaker normalisation is an efficacious way to mitigate the discrepancy between testing and training conditions caused by the variations in the manner of speaking. We propose an i-vector based feature augmentation technique which incorporates i-vector as an additional input to the DNN model to compensate for the speaker effects in the speech signal. It helps DNN remove the underlying speaker information while retaining the required linguistic content by learning meaningful representations from acoustic features augmented with i-vectors.

We used knowledge transfer from pre-trained ASR systems and self-supervised speech representation models under low resource settings to improve the performance. To assess the proposed technique, we used a banking domain speech intent dataset in Sinhala and Tamil languages. We used Fluent speech command dataset consisting of English speech recordings for further comparison. The experimental results indicate that the integration of i-vectors improves the performance of the intent classification model by learning better representations. For Sinhala dataset, APC features yielded the highest classification accuracy of 98 %, outperforming the approach presented in the previous study. For Tamil and English datasets, phoneme probability features yielded the best results with accuracy values of 51 % and 81 % respectively.

In the future, we plan to investigate the effeciveness of more self-supervised acoustic features including wav2vec, Mockingjay, audio Albert, and Tera. In addition, we intend to extend this work by carrying out experiments with speech datasets in various languages and domains to demonstrate the model's ability to generalize.

Bibliography

[1] D. Amodei, S. Ananthanarayanan, R. Anubhai, J. Bai, E. Battenberg, C. Case, J. Casper, B. Catanzaro, Q. Cheng, G. Chen, J. Chen, J. Chen, Z. Chen, M. Chrzanowski, A. Coates, G. Diamos, K. Ding, N. Du, E. Elsen, and Z. Zhu, "Deep speech 2: End-to-end speech recognition in english and mandarin," in *ArXiv*, 12 2015.

[2] L. Lugosch, M. Ravanelli, P. Ignoto, V. Tomar, and Y. Bengio, "Speech model pre-training for end-to-end spoken language understanding," *ArXiv*, vol. abs/1904.03670, 2019.

[3] Y.-A. Chung, H. Tang, and J. Glass, "Vector-quantized autoregressive predictive coding," in *INTERSPEECH*, 2020.

[4] A. H. Liu, Y.-A. Chung, and J. Glass, "Non-autoregressive predictive coding for learning speech representations from local dependencies," *ArXiv*, vol. abs/2011.00406, 2020.

[5] Y.-P. Chen, R. Price, and S. Bangalore, "Spoken language understanding without speech recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 6189–6193.

[6] J. Poncelet and H. Van hamme, "Multitask learning with capsule networks for speech-to-intent applications," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 05 2020, pp. 8494–8498.

[7] Y. Karunanayake, U. Thayasivam, and S. Ranathunga, “Sinhala and tamil speech intent identification from english phoneme based asr,” *2019 International Conference on Asian Language Processing (IALP)*, pp. 234–239, 2019.

[8] B. Zhen, X. Wu, Z. Liu, and H. Chi, “On the importance of components of the mfcc in speech and speaker recognition,” in *INTERSPEECH*, vol. 37, 01 2000, pp. 487–490.

[9] V. Këpuska and H. Elharati, “Robust speech recognition system using conventional and hybrid features of mfcc, lpcc, plp, rasta-plp and hidden markov model classifier in noisy conditions,” *Journal of Computer and Communications*, vol. 03, pp. 1–9, 01 2015.

[10] K. Aida-zade, C. Ardil, and S. Rustamov, “Investigation of combined use of mfcc and lpc features in speech recognition systems,” *Signal Processing*, 01 2007.

[11] S. Shum, N. Dehak, R. Dehak, and J. Glass, “Unsupervised speaker adaptation based on the cosine similarity for text-independent speaker verification,” *Odyssey*, 01 2010.

[12] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, “X-vectors: Robust dnn embeddings for speaker recognition,” in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 04 2018, pp. 5329–5333.

[13] Y. Shi, Q. Huang, and T. Hain, “H-vectors: Utterance-level speaker embedding using a hierarchical attention model,” in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 05 2020, pp. 7579–7583.

[14] A. Senior and I. Lopez-Moreno, “Improving dnn speaker independence with i-vector inputs,” in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 05 2014, pp. 225–229.

[15] C. yu, A. Ogawa, M. Delcroix, T. Yoshioka, T. Nakatani, and J. Hansen, "Robust i-vector extraction for neural network adaptation in noisy environment," in *INTERSPEECH*, 09 2015.

[16] S. Garimella, A. Mandal, N. Strom, B. Hoffmeister, S. Matsoukas, and S. H. K. Parthasarathi, "Robust i-vector based adaptation of dnn acoustic model for speech recognition," in *INTERSPEECH*, 2015.

[17] Y. Miao, H. Zhang, and F. Metze, "Towards speaker adaptive training of deep neural network acoustic models," in *INTERSPEECH*, 2014.

[18] X. Cui, V. Goel, and G. Saon, "Embedding-based speaker adaptive training of deep neural networks," in *INTERSPEECH*, 08 2017, pp. 122–126.

[19] N. Tomashenko, A. Caubrière, and Y. Estève, "Investigating Adaptation and Transfer Learning for End-to-End Spoken Language Understanding from Speech," in *Proc. Interspeech 2019*, 2019, pp. 824–828. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2019-2158

[20] Y.-A. Chung, W.-N. Hsu, H. Tang, and J. R. Glass, "An unsupervised autoregressive model for speech representation learning," *ArXiv*, vol. abs/1904.03240, 2019.

[21] Y.-A. Chung and J. R. Glass, "Generative pre-training for speech with autoregressive predictive coding," *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3497–3501, 2020.

[22] B. Schuller, S. Steidl, A. Batliner, F. Burkhardt, L. Devillers, C. Müller, and S. Narayan, "Paralinguistics in speech and language - state-of-the-art and the challenge," *Computer Speech and Language, Special Issue on Paralinguistics in Naturalistic Speech and Language*, 01 2013.

[23] B. Belean, "Comparison of formant detection methods used in speech processing applications," *AIP Conference Proceedings*, vol. 1565, pp. 85–89, 11 2013.

[24] J. P. Teixeira and A. Gonçalves, “Algorithm for jitter and shimmer measurement in pathologic voices,” *Procedia Computer Science*, vol. 100, pp. 271 – 279, 2016.

[25] X. Li and X. Wu, “Modeling speaker variability using long short-term memory networks for speech recognition,” in *INTERSPEECH*, 2015.

[26] Y. zhao, J. Li, X. Wang, and Y. Li, “The speechtransformer for large-scale mandarin chinese speech recognition,” in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 05 2019, pp. 7095–7099.

[27] Z. Fan, J. Li, S. Zhou, and B. Xu, “Speaker-aware speech-transformer,” *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pp. 222–229, 2019.

[28] J. Pan, D. Liu, G. Wan, J. Du, Q. Liu, and Z. Ye, “Online speaker adaptation for lvcsr based on attention mechanism,” *2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pp. 183–186, 2018.

[29] W.-N. Hsu, Y. Zhang, and J. R. Glass, “Unsupervised learning of disentangled and interpretable representations from sequential data,” in *NIPS*, 2017.

[30] W.-N. Hsu and J. R. Glass, “Extracting domain invariant features by unsupervised learning for robust automatic speech recognition,” *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5614–5618, 2018.

[31] S. Feng and T. Lee, “Improving unsupervised subword modeling via disentangled speech representation learning and transformation,” in *INTERSPEECH*, 2019.

[32] J. Chorowski, R. J. Weiss, S. Bengio, and A. van den Oord, “Unsupervised speech representation learning using wavenet autoencoders,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, pp. 2041–2053, 2019.

[33] S. Schneider, A. Baevski, R. Collobert, and M. Auli, “wav2vec: Unsupervised pre-training for speech recognition,” in *INTERSPEECH*, 2019.

[34] A. T. Liu, S.-W. Li, and H. yi Lee, “Tera: Self-supervised learning of transformer encoder representation for speech,” *ArXiv*, vol. abs/2007.06028, 2020.

[35] P.-H. Chi, P.-H. Chung, T. Wu, C.-C. Hsieh, S.-W. Li, and H. yi Lee, “Audio albert: A lite bert for self-supervised learning of audio representation,” *ArXiv*, vol. abs/2005.08575, 2020.

[36] A. T. Liu, S. Yang, P.-H. Chi, P.-C. Hsu, and H. yi Lee, “Mockingjay: Unsupervised speech representation learning with deep bidirectional transformer encoders,” *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6419–6423, 2020.

[37] E. Morais, H. Kuo, S. Thomas, Z. Tuske, and B. Kingsbury, “End-to-end spoken language understanding using transformer networks and self-supervised pre-trained features,” *ArXiv*, vol. abs/2011.08238, 2020.

[38] E. Palogiannidi, I. Gkinis, G. Mastrapas, P. Mizera, and T. Stafylakis, “End-to-end architectures for asr-free spoken language understanding,” *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7974–7978, 2020.

[39] M. Radfar, A. Mouchtaris, and S. Kunzmann, “End-to-end neural transformer based spoken language understanding,” in *INTERSPEECH*, 2020.

[40] Y. Karunanayake, U. Thayasivam, and S. Ranathunga, “Transfer learning based free-form speech command classification for low-resource languages,” in *ACL*, 2019.

[41] A. Larcher, K.-A. Lee, and S. Meignier, “An extensible speaker identification sidekit in python,” *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5095–5099, 2016.

[42] A. Hannun, C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Satheesh, S. Sengupta, A. Coates, and A. Ng, "Deepspeech: Scaling up end-to-end speech recognition," *ArXiv*, 12 2014.

[43] R. Ardila, M. Branson, K. Davis, M. Henretty, M. Kohler, J. Meyer, R. Morais, L. Saunders, F. M. Tyers, and G. Weber, "Common voice: A massively-multilingual speech corpus," in *LREC*, 2020.

[44] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An asr corpus based on public domain audio books," *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5206–5210, 2015.

[45] D. Buddhika, R. Liyadipita, S. Nadeeshan, H. Witharana, S. Jayasena, and U. Thayasivam, "Voicer: A crowd sourcing tool for speech data collection," in *2018 18th International Conference on Advances in ICT for Emerging Regions (ICTer)*, 2018, pp. 174–181.