Robust Methods for the Automatic Quantification and Prediction of Affect in Spoken Interactions

by

Zakaria Aldeneh

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Computer Science and Engineering)
in the University of Michigan
2020

Doctoral Committee:

Professor Emily Mower Provost, Chair
Professor Robin Brewer
Professor Joyce Chai
Professor Rada Mihalcea

Zakaria Aldeneh

aldeneh@umich.edu

ORCID iD: 0000-0003-4599-2448

Dedicated to my parents, Yehia and Linda, for their boundless love and support.

ACKNOWLEDGMENTS

First and foremost, I would like to thank my adviser, Emily Mower Provost, for taking me as a student and providing me with the support and guidance that I needed to succeed in graduate school. The experience I gained is worthwhile and invaluable, and this research work would not have been possible if not for the continued encouragement and opportunities that Emily has provided.

Thanks are also due to my doctoral committee members, Dr. Robin Brewer, Dr. Joyce Chai, and Dr. Rada Mihalcea for their insightful feedback and constructive suggestions on the topics of my thesis.

I am also greatly thankful to both former and current members of the CHAI Lab: Yelin, Duc, June, Soheil, John, Didi, Matt, Mimansa, Amrit, Katie, Haley, Barbara, Sandy, Alex, and Noor. They were always happy to discuss and provide suggestions during our lab lunches and coffee breaks. I am especially thankful to Yelin, Duc, Soheil, John, Didi, Matt, and Mimansa for being wonderful research collaborators and co-authors on several pieces of my work.

I am grateful to the friends that I made during my time at Michigan: Mahmoud, Javad, Samer, Noura, Sari, Mohamed, Abhishek, Sai, and Charlie. They were always there to listen to my problems and provide me with the support I needed during difficult times.

Finally, I would like to express my sincere gratitude to my parents and my siblings for their continued support and love. My parents have sacrificed a lot to bring me and my siblings to the United States, so we can have a better future.

TABLE OF CONTENTS

LIST OF FIGURES

Figure

LIST OF TABLES

Table

ABSTRACT

Emotional expression plays a key role in interactions as it communicates the necessary context needed for understanding the behaviors and intentions of individuals. Therefore, a speech-based Artificial Intelligence (AI) system that can recognize and interpret emotional expression has many potential applications with measurable impact to a variety of areas, including human-computer interaction (HCI) and healthcare. However, there are several factors that make *speech emotion recognition* (SER) a difficult task; these factors include: variability in speech data, variability in emotion annotations, and data sparsity.

This dissertation explores methodologies for improving the robustness of the automatic recognition of emotional expression from speech by addressing the impacts of these factors on various aspects of the SER system pipeline. For addressing speech data variability in SER, we propose modeling techniques that improve SER performance by leveraging short-term dynamical properties of speech. Furthermore, we demonstrate how data augmentation improves SER robustness to speaker variations. Lastly, we discover that we can make more accurate predictions of emotion by considering the fine-grained interactions between the acoustic and lexical components of speech. For addressing the variability in emotion annotations, we propose SER modeling techniques that account for the behaviors of annotators (i.e., annotators' reaction delay) to improve time-continuous SER robustness. For addressing data sparsity, we investigate two methods that enable us to learn robust embeddings, which highlight the differences that exist between neutral speech and emotionally expressive speech,

without requiring emotion annotations. In the first method, we demonstrate how emotionally charged vocal expressions change speaker characteristics as captured by embeddings extracted from a speaker identification model, and we propose the use of these embeddings in SER applications. In the second method, we propose a framework for learning emotion embeddings using audio-textual data that is not annotated for emotion.

The unification of the methods and results presented in this thesis helps enable the development of more robust SER systems, making key advancements toward an interactive speech-based AI system that is capable of recognizing and interpreting human behaviors.

CHAPTER I

Introduction

1.1 Motivation

Emotional expression plays a key role in interactions as it communicates the necessary context needed for understanding the behavior of individuals. Providing machines with the ability to recognize and interpret human emotions can impact various fields, ranging from healthcare to Human-Computer Interaction (HCI) [1, 2, 3, 4, 5]. For instance, in healthcare, an emotion-aware system can aid in the diagnoses and management of mental health disorders [6, 3]. In automotive safety, a system can detect levels of driver alertness to determine driver engagement [7, 8]. In advertising, emotion-aware systems can measure consumers' emotional engagement to advertisements and movie trailers [9, 10]. In education, an emotion-aware intelligent tutoring spoken dialogue system can increase student persistence by predicting and adapting to student emotions [11].

Emotional expression is inherently a multimodal phenomenon that is communicated through various channels, including, head and body movements, facial expressions, language, and speech [12, 13, 14, 15]. In addition to being expressed through interactive behavioral cues, emotion manifests itself in physiological signals, and can, for instance, affect the temperature and heart rate of an individual [16, 17, 18]. The automatic detection of emotion from speech, however, has garnered special attention

from the research community due to the prevalence of speech-based devices (e.g., recording devices, virtual personal assistants, smart watches, etc.), which can be used for collecting data in a remote and non-intrusive fashion.

Speech emotion recognition (SER) is a difficult task due to various factors, which include the highly variable nature of speech, the variabilities associated with emotion expression and perception, and the limited access to high-quality labeled data needed for building robust SER systems. As a result of these factors, SER systems that are built in controlled conditions often fail to generalize when deployed in real world settings where the conditions are different. This dissertation presents novel solutions and modeling techniques that improve SER robustness by addressing important challenges in the area.

1.2 Describing Emotion

Speech data need to be labeled with descriptors of emotion before they can be used for building and evaluating SER systems. These descriptors aim to capture the underlying emotional state of a speaker. There are two common views that are used for describing emotion: the categorical view [19, 20, 21, 22] and the dimensional view [23, 24, 25, 26]. In the categorical view, emotion is described using discrete attributes (e.g., excited, happy, angry, sad). The categorical view is inspired by the theory of discrete characterization of emotion, which posits that there exists a set of "basic" emotions that have evolved to aid in the survival and adaptation of organisms [27, 28]. In this view, more complex emotions emerge as a result of combining two or more basic emotions. For example, the emotion of jealousy emerges by combining the basic emotions of anger and sadness [29, 30]. Although several sets of basic emotions have been proposed in previous work, the set proposed by Ekman (anger, disgust, fear, happiness, sadness, and surprise) is the most commonly used in SER research.

One limitation with using the categorical descriptors of emotion in SER systems, however, is that these descriptors often fail to capture the various subtleties and intensities that exist in emotional expression with only a small number of descriptors. For instance, "resentfulness", "anger" and "rage" would require three separate categories in an SER system even though they may represent different intensities of the same emotion. The dimensional view of emotion addresses this limitations by defining emotion based on its primary properties in a continuous space [24, 31, 32]. The most common dimensions used in SER research are defined by Russel, and they include arousal (calm to energetic) and valence (negative to positive) [25].

Both the categorical and dimensional descriptors of emotion have been used in the research community for building and evaluating automatic recognition systems. We use both descriptions of emotion in this dissertation.

1.3 Methods for SER

We provide a general overview of the methods used for SER in this section but defer the descriptions of more specific related works to the corresponding chapters. Early works focused on using generative and discriminative machine learning approaches for building emotion recognition models (e.g., Gaussian mixture models, support vector machines) [33, 34, 35]. Most of the contributions in these early works came from engineering features to reflect emotion variations in speech. Many of these features were borrowed from acoustic analysis studies done on speech utterances collected from individuals displaying different emotions [36, 37, 38]. Some of the popular feature sets include the IS09, ComParE, and the eGeMAPS feature sets [39, 40, 41]. These feature sets typically consist of energy, spectral, and voicing related acoustic features, and are extracted in a two-step process. First, a number of low-level-descriptors (LLDs) are extracted from the content of a short sliding window (e.g., 25 milliseconds Hamming window with a shift rate of 10 milliseconds) that is applied to the acoustic signal.

Then, a set of statistics (e.g., mean, standard deviation) are applied to the extracted LLDs to get a fixed-size feature representation of an utterance.

More recent approaches to SER have focused on using neural networks to build recognition models that rely on Mel-filterbanks (MFBs), spectrograms, or raw waveforms [42, 43, 44, 45, 46, 47]. These approaches exploit neural networks' ability to extract powerful representations (i.e., embeddings) that are tailored for the recognition objective from minimally processed input features. For instance, Ghosh et al. investigated the use of denoising autoencoders for learning paralinguistic attributes from MFBs and spectrograms, and demonstrated how autoencoders yielded features that are discriminative to emotions [42]. Latif et al. proposed a multi-resolution neural model for detecting emotions from raw-waveforms, and showed that their proposed model performed on par with SER models that relied on hand-engineered features [48]. Finally, Trigeorgis et al. introduced a convolutional recurrent model that operated on raw-waveforms, and demonstrated improvements in recognition performance compared to a model that used traditional features [49].

Other contributions to SER in the recent years came from the introduction of novel neural network architectures, the development of more generalizable models that address data and emotion variability challenges discussed below, and the collection of emotion corpora needed for building SER models [50, 51, 52, 47, 53, 54]. For example, Parthasarathy and Busso proposed ladder networks, which employ an unsupervised auxiliary task of reconstructing intermediate features, to allow for the utilization of unlabeled data from a target domain [50]. We investigated the use of progressive neural networks for SER, and found that augmenting the emotion recognition task with speaker, gender, and additional datasets improved performance over baselines that solely used emotion [55]. Albanie et al. showed how one could exploit the correlation between a person's speech and facial expression to learn speech emotion embeddings from unlabeled audio-visual data through cross-modal transfer using

a pre-trained facial expression detector [54].

1.4 Challenges in SER

The detection of emotional expression from speech is a difficult task, as there are many sources of variability that need to be accounted for when building SER systems. In this section, we describe three major challenges that face SER systems: (1) speech data variability, (2) emotion label variability, and (3) data sparsity.

Speech Data Variability. Many factors modulate speech; these factors include the recording environment (e.g., recording device, distance from microphone, noise level), speaker demographics (e.g., gender, accent, dialect), and linguistic content. As a result of these factors, SER systems developed using data collected from one domain typically fail to generalize when deployed to new domains. Several techniques have been developed in the speech processing and SER communities to compensate for these factors. Some of these techniques include feature normalization [56, 57, 58], domain adaptation [59], and adversarial training [60, 61, 62]. For example, Zhang et al. demonstrated the benefits of handling data variability, specifically environmental properties and gender, via the multi-task learning paradigm [63]. Abdelwahab and Busso showed how the adversarial training paradigm can be used to train neural models that extract features which are invariant to domain shifts [59]. Li et al. applied adversarial training to disentangle speaker characteristics and demographics from emotion features [64].

Emotion Label Variability. The subjective nature of emotion expression and perception can lead to a set of challenges that are unique to the SER task. Individuals typically express emotions in ways that are unique to themselves. As a result, an SER system that is built using data collected from certain individuals can unintentionally overfit to those individuals and, consequently, fail to generalize when used for recognizing the emotions of unseen individuals. Although most previous works in SER do

not address subjectivity in expression, few works have focused on personalization in SER to conform to, and exploit, this property (e.g., [57, 65, 66]). Like subjectivity in emotion expression, subjectivity in emotion perception also makes SER a challenging task. Individuals can have varying opinions regarding the emotion expressed in a given speech sample. Thus, the content of speech corpora used for building SER systems is typically evaluated by a number of annotators for emotion content. Previous works addressed this challenge by taking the average (or the mode) of the evaluations obtained from multiple raters (e.g., [53, 67, 68]). Other works have demonstrated that the disagreement between annotators provides useful information about the subtlety of the expressed emotions, and showed how this information can be used for building more robust SER models (e.g., [69, 70, 71, 72]).

Data Sparsity. Finding media sources that provide content with diverse emotional expression needed for building SER models is challenging. In addition, the necessity for having multiple annotators when collecting data makes the process both costly and time consuming [54]. As a result, datasets that are typically used for building and testing SER models remain significantly smaller than those used for building other speech models (e.g., speaker recognition and automatic speech recognition), even though the SER application faces the same data variability challenges that other speech applications face. For instance, the size (in recorded hours of speech) of a modern dataset used for developing speaker recognition systems (e.g., VoxCeleb [73]) is around 2,000 hours. In contrast, the size of the MSP-Podcast dataset, a recently released emotion dataset, is around 100 hours.[1] The challenge of having a small emotion dataset is often compounded by challenges from having labels with low annotator agreement as well as from the lack of balanced presentation of emotional expressions (i.e., emotion label imbalance). In addition, the data collection strategies used in recent works, including MSP-Podcast, relied on pre-trained emotion recognizers, which

[1]The collection of the MSP-Podast dataset is an ongoing project; the authors' goal for the dataset is to reach 400 hours of speech.

themselves are trained with small and subjective data, for retrieving candidate speech samples. The availability of large and diverse datasets for building SER systems can help attenuate many of the speech data variability challenges mentioned above.

1.5 Proposed Methods

This section presents the methods that we introduce in this dissertation document as well as the key challenges that the methods address. Sections 1.5.1, 1.5.2, and 1.5.3 introduce and investigate modeling techniques that address speech data variability and emotion annotation variability challenges. Sections 1.5.4 and 1.5.5 mainly address the data sparsity challenge by introducing methods for extracting embeddings which highlight expressive and emotional content in speech without using emotion labels.

1.5.1 Using Regional Saliency in Speech for SER

A key step in the SER feature extraction process is the application of statistics (e.g., mean, standard deviation) to the sequential frame-level acoustic features (i.e., frames) extracted from the waveform. The benefit of this step is that it allows for a fixed-size description of how properties of the low-level acoustic frames change over the course of an utterance. However, one limitation with using statistics is that all acoustic frames are treated equally regardless of their content. In other words, the content of the frames that carry emotion relevant information can be obfuscated by the contents of frames that do not carry emotion relevant information. In Chapter III, we demonstrate how convolutional neural networks can be directly applied to sequential frame-level acoustic features to identify emotionally salient regions without the need for defining or applying utterance-level statistics. We also show how utterance-level statistics can obfuscate emotional information. This study demonstrates that the current approach of feature extraction might not be the most effective approach when building SER systems as it fails to consider dynamic variations across an utterance.

1.5.2 Pooling Acoustic and Lexical Features for SER

The inherently multimodal nature of emotion perception and expression suggests that SER models can benefit from the use of these various modalities. Individuals rely on both linguistic and paralinguistic attributes to express and perceive emotion during spoken interactions. Thus, supplying SER models with both the acoustic and lexical modalities can provide a more complete signal about behavior. The use of multiple modalities by an SER model entails fusing the different streams of information. Several methods for multimodal fusion exist in the literature [74, 75, 76]. However, it is unclear which pooling technique is most effective for combining acoustic and lexical features for the SER task. In addition, it is not clear how much each modality contributes to the performance of SER models. In Chapter IV, we present an analysis of different multimodal fusion approaches in the context of deep learning, focusing on pooling intermediate representations learned from the acoustic and lexical modalities for SER. We also study the influence of each (i.e., the acoustic and lexical) modality on the overall performance of an SER system. This study demonstrates that a multimodal fusion strategy that considers fine-grained interaction between the acoustic and lexical features is most effective.

1.5.3 Capturing Long-term Dependencies for SER

Emotion can be quantified using categorical classes (e.g., happy, neutral, sad, etc.) or using dimensional values (e.g., valence-arousal). In addition, emotional labels can be quantified statically, over units of speech (e.g., utterances), or continuously in time. Emotion labels that are quantified continuously in time provide fine-grained information about the behavior of an individual as a function of time. However, time-continuous emotion annotations create two challenges that need to be addressed in an SER system for effective modeling. The first challenge is that the reaction delay of the annotators creates a mismatch between the acoustic signal and the emotion

annotations. In other words, the annotations for a given acoustic signal will be shifted in time. The second challenge is that the acoustic signal exhibits more variations in time compared to the annotation signal (i.e., the annotation signal is smooth and has considerable time dependencies). In Chapter V, we introduce neural network architectures that address these two challenges and improve SER performance. Specifically, we address the first challenge by proposing the use of convolutional architectures that have a large receptive fields to allow the networks to implicitly compensate for the reaction delay of annotators. We address the second challenge by proposing the use of a convolutional architecture that models a downsampled (i.e., compressed) version of the input acoustic signal and then generates the output signal through an upsampling operation. We demonstrate how addressing these two challenges improves SER performance compared to baseline methods that do not take these effects into account when modeling. This study demonstrates how the behavior of annotators can guide the design of more effective SER systems.

1.5.4 Speaker Embeddings as Robust Features for SER

The performance of an SER system depends on the features used to represent the acoustic signal. Several features have been introduced in the SER literature (e.g., ComParE, eGeMAPS, etc.) [41, 77, 39]. However, these features can be susceptible to distortions due to changes in the recording conditions or due to the presence of noise. Previous research in other domains (e.g., computer vision) has demonstrated that neural networks trained discriminatively on large and diverse datasets learn to extract generalizable embeddings (e.g., [78, 79]). These embeddings are obtained from intermediate layers that the trained networks extract from the input features. However, the main requirement for learning powerful embeddings using neural networks is the access to large labeled datasets; a requirement that is still missing in the SER community as discussed in Section 1.4. This necessitates an alternative approach

for learning speech embeddings that capture emotion information and suppress any extraneous variabilities that exist in the speech data. In Chapter VI, we propose the use of speaker embeddings, features extracted from networks trained on the closely related task of speaker recognition, as robust SER features. We show that expressive speech disturbs speaker embeddings (i.e., speakers sound less like themselves when they are vocally expressive), and demonstrate how these disturbances can be used for recognizing emotions. This study shows that speaker embeddings can be used as a replacement to traditional emotion features.

1.5.5 Learning Emotion Embeddings using Speech and Text

The data sparsity challenge discussed in Section 1.4 makes it difficult to learn a general emotion embedding that captures expressiveness in speech via the supervised learning paradigm. Chapter VI of this dissertation shows that the speaker recognition task can facilitate an alternative approach for learning such embedding. However, one limitation with that approach is that it still relies on a large number of labeled data (i.e., speaker labels). In Chapter VII, we propose a framework for learning emotion embeddings from large-scale (i.e., 200 hours) audio-textual data without using emotion or speaker labels. The key assumption behind the proposed framework is that *expressive* speech can be considered as a modulation to *neutral* speech. Thus, a neural network can be trained to learn what it means for speech to be expressive if we provide pairs of *expressive* and *unexpressive* (neutral) utterances. To this end, we demonstrate how an off-the-shelf speech synthesizer can be used to generate a neutral version of expressive speech data, and then propose a neural model that leverages this resulting neutral-expressive data pairs to learn emotion embeddings. We show that the learned emotion embeddings highlight emotion characteristics in an utterance by demonstrating how they improve emotion classification performance compared to surface MFB features. This study shows that it is possible to leverage naturally

occurring multimodal data (i.e., speech and text) to learn emotion embeddings while circumventing the data collection and annotation challenges.

1.6 Contributions

This dissertation presents novel solutions for detecting and quantifying emotional expression from speech. Chapters III, IV, and V introduce and investigate novel modeling techniques that improve SER performance by addressing speech data and emotion label variability. Chapters VI and VII introduce methods that extract robust embeddings for SER by addressing data sparsity. The contributions of the works presented in this dissertation are summarized as follows:

- Chapter III:
  - We demonstrate the effectiveness of speed perturbation as a data augmentation technique to increase the amount of data available for training SER models.
  - We show how the traditional two-step feature extraction framework used in paralinguistics is not the most effective for SER tasks because it obfuscates the emotion relevant information in an utterance.
- Chapter V:
  - We propose the use of neural network architectures that incorporate long-term context in time-continuous SER applications. These architectures improve performance over baselines by allowing recognition models to compensate for annotator reaction delay.
  - We propose the use of neural network architectures that account for the non-instantaneous nature of human annotations in time-continuous SER

applications. These architectures improve performance over baselines by generating predictions that mimic human annotations.

- Chapter IV:
  - We investigate approaches for fusing intermediate representations from the acoustic and lexical modalities in SER models. We demonstrate how fusion strategies that consider fine-grained interactions between the modalities are most effective for SER.
- Chapter VI:
  - We investigate the utility of speaker embeddings that are extracted from speaker recognition networks in SER tasks, and show that speaker embeddings can be used as a robust replacement to traditional emotion features in SER tasks.
- Chapter VII:
  - We propose a framework for learning emotion embeddings by leveraging naturally occurring audio-textual data without requiring explicit emotion labels. We demonstrate how the learned emotion embeddings improve performance over baseline acoustic features.

1.7 Outline of Dissertation

This dissertation document is organized as follows. Chapter II introduces the datasets that we use in our studies. Chapter III describes our work using convolutional neural networks to model emotionally salient regions in an utterance. Chapter IV describes our work on pooling acoustic and lexical information to improve SER

performance. Chapter V details our work on capturing long-term contextual information in continuous emotion recognition for handling annotator variability. Chapter VI investigates the utility of speaker embeddings for SER. Chapter VII details our work on learning emotion embeddings from large-scale audio-textual data that are not annotated for emotion. Finally, Chapter VIII summarizes the main findings and highlights possible future directions.

CHAPTER II

Datasets

Several emotion datasets exist in the public domain. Some of the attributes that differentiate one dataset from another include: the media source (e.g., online media, movies, TV shows, laboratory recordings), the emotion elicitation method used (e.g., natural, improvised, acted), the emotion descriptors used (e.g., categorical, dimensional), language, and the number of modalities measured by the dataset collectors. In this dissertation, we use four datasets in total, including the IEMOCAP, MSP-IMPROV, RECOLA, and VESUS. We introduce these datasets in this chapter but defer descriptions about processing and feature extraction to the corresponding chapters.

2.1 IEMOCAP

The interactive emotional dyadic motion capture (IEMOCAP) dataset was collected to study audio-visual emotional expression in dyadic interactions [15]. Interactions in the dataset were recorded from five dyadic sessions, each between a male and a female actor. In each session, the actors perform a series of scripted and improvised scenarios designed to elicit emotion expression. The dataset contains approximately 12 hours of data across four modalities (audio, text, video, and motion-capture) 10 speakers (five males and five females). The recordings from each interaction were

manually segmented into utterances such that each utterance contains a complete sentence or a speaker turn (whichever is shorter). The resulting utterances were annotated for categorical emotions by at least three annotators; and for dimensional emotion by at least two annotators. The annotators assigned categorical emotions for each utterance from the following set: {angry, happy, neutral, sad, frustrated, excited, disgusted, fearful, surprised, other}. The annotators assigned valence, activation, and dominance (dominant vs. submissive) levels using five-point Likert scale. More details about the IEMOCAP corpus can be found in [15]

2.2 MSP-IMPROV

The MSP-IMRPOV dataset was collected to study audio-visual emotional expression in dyadic interactions while maintaining partial control over lexical content [80]. Interactions in the dataset were recorded from six dyadic sessions, each between a male and a female actor. The actors in each dyadic interaction improvise scenarios that lead one of them to utter a target sentence in a specific emotion. This approach of eliciting emotion was designed to maintain the spontaneous nature of the interaction while controlling for lexical content. Overall the dataset contains approximately nine hours of speech from 12 speakers (six males and six females). Similar to the recordings in IEMOCAP, the ones in MSP-IMPROV were manually segmented into utterances such that each utterance contains a complete sentence or a speaker turn (whichever is shorter). The resulting utterances were annotated for categorical and dimensional emotions by at least five annotators. The annotators assigned categorical emotions for each utterance from the following set: {angry, happy, sad, neutral, other}. The annotators assigned valence, activation, dominance (dominant vs. submissive), and naturalness (acted vs. natural) levels using five-point Likert scale. More details about the MSP-Improv corpus can be found in [80].

2.3 RECOLA

The Remote Collaborative and Affective Interactions (RECOLA) database [81] was collected to study affective behaviors in remote dyadic interactions. Multimodal data (i.e., audio, video, electro-cardiogram, and electro-dermal activity) was collected from French speaking participants while they complete collaborative task. A portion of each interaction was annotated in a temporal fashion by six annotators for activation and valence using a slider with values ranging from -1 to $+1$. The annotations from each annotators were normalized before being averaged to produce the ground-truth activation and valence signal for each interaction. The RECOLA corpus, as introduced in the 2016 audio/visual emotion challenge (AVEC), contains 27 five minute-recordings from 27 speakers (16 female and 11 male speakers). More details about the RECOLA corpus can be found in [81] and [82].

2.4 VESUS

The Varied Emotion in Syntactically Uniform Speech (VESUS) dataset was collected to provide the research community with a lexically controlled emotional dataset [83]. Over 250 distinct phrases were uttered by 10 actors (five males and five females) while portraying five emotional states (*Neutral*, *Angry*, *Happy*, *Sad*, and *Fear*). The phrases were chosen such that they are semantically neutral, i.e., they don't carry any emotional connotation. Overall the dataset contains approximately six hours of speech. The utterances in the dataset contain labels assigned based on the intended emotion by the actors and labels assigned based on the perceived emotion collected from 10 crowd-sourced annotators.

CHAPTER III

Using Regional Saliency in Speech for SER

3.1 Introduction

Traditional SER systems follow one of three major approaches. In the first approach, utterance-level statistics are applied to sequential low-level descriptors (LLDs) extracted from utterances of variable lengths to obtain fixed-length features that describes the global characteristics of the given utterances. These fixed-length features can then be used to train machine learning classifiers (e.g., [84, 85]). While popular, we hypothesize that this approach dilutes important regional information by combining it with potentially irrelevant information from neighboring frames.

Two recent papers [86, 87] showed that one can train classifiers using only a portion of the information contained within utterances and still achieve competitive results. In particular, Le et al. [86] showed that state-of-the-art results can be obtained on the FAU Aibo 2-class problem using less than 50% of the data contained within an utterance. Kim et al. [87] showed that emotional information in an utterance is regionalized and follows specific patterns. Echoing the findings of Le et al. they showed that, in some cases, systems that use only 59% of the data within an utterance can achieve performance that is similar to that achieved by systems that use 100% of the data. This suggests that traditional SER approaches inadvertently include irrelevant information when creating fixed-length features.

In the second approach, statistical functions are applied to windowed segments of utterances to create statistical descriptions of the segments. These statistics are then classified to create sequences of emotion confidences. Given this sequence of emotion confidences, the problem becomes a time series classification problem (e.g., [88]). This approach assumes that all segments take the same emotional label as their parent utterance and thus assumes that all regions of utterances contain relevant emotional information.

Finally in the third approach, frameworks that are capable of directly modeling temporal LLDs are used to build SER systems. Many of these approaches were inspired by approaches proposed in the automatic speech recognition (ASR) community. Notable approaches include HMM-DNN hybrids [65] and deep end-to-end systems [49]. Such approaches require modeling the dynamics of emotion.

We hypothesize that focusing on emotionally salient regions of utterances can allow us to build robust SER systems that do not require defining statistical functions or making any assumptions about frame-level emotional labels. In this work, we use convolutional neural networks (CNNs) to learn emotion classifiers from speech. CNNs have shown tremendous success in the fields of ASR [89], computer vision [90], and sentence classification [91]. CNNs allow multiple regions of the input to share the same weights; overcoming the scalability problem of regular neural networks. In addition, CNNs can be applied to inputs of variable sizes, thus easing one of the challenges of dealing with variable length speech data.

The contributions of this chapter are as follows: (1) we show how a simple CNN that uses minimally hand-engineered features can yield competitive results when compared to results obtained from systems trained on popular emotion feature sets; (2) we show how applying statistical functions to temporal LLDs can washout information causing loss of performance; (3) we show how using speed augmentation can improve the performance of SER systems.

3.2 Related Work

CNNs have been used for SER. Most notably, Mao et al. [92] used CNNs to learn salient features to be used by an SVM for classification. The authors followed three steps to build their SER system. First, they used sparse auto-encoders to learn filters from spectrogram segments. The authors convolved the learned filters with spectrogram fragments to produce feature vectors. Second, the authors mapped the feature vectors into two smaller feature vectors using a semi-supervised objective function. The objective function disentangled affect-salient features from other non-salient features. Third, the authors used the affect-salient features to train SVMs. The authors finally compared the discriminative performance of features obtained from different stages of the CNN.

Other works used neural networks and recurrent neural networks for SER. Le et al. [65] followed an approach that is similar to the ones followed in ASR literature and used a HMM-DNN hybrid approach [93] to train an SER system. The authors investigated different ways to model emotion as an HMM and finally drew a contrast between the fields of emotion and speech recognition.

Han et al. [94] and Lee et al. [95] both took a multi-step approach to the problem of SER. In the first step, Han et al. [94] trained a neural network using frame-level features (along with contextual information) while Lee et al. [95] trained a 2-layer bidirectional long short-term memory (BLSTM) network. The trained models were used to produce frame-level emotional predictions (four channel time-series). Both authors applied statistical functions to the time-series data before feeding the results into another simple neural network for utterance-level classification.

Xia et al. [96] used denoising autoencoders to to build SER models that take gender into account. The authors train gender-specific models using neutral speech obtained from a large ASR dataset. The results suggested that modeling gender variability can be useful for emotion recognition. In other work, Xia et al. [85] used

a multi-task learning approach to leverage additional data with continuous labels (as opposed to categorical labels) to train a network. The authors showed that using regression as a secondary task can improve the overall performance of the system when compared to a single-task system that only relies on examples with categorical labels.

Finally, motivated by a recent trend in deep learning where raw data is used with minimal feature pre-processing, Trigeorgis et al. [49] devised an end-to-end deep network that worked on raw time-domain signals. The authors first applied convolutions to extract features before they fed the extracted features into a LSTM structure for prediction in the valence-activation space.

All of the cited related work does at least one of the following: (1) makes assumptions about the length of utterances and labels [49]; (2) relies on manual feature engineering [96, 85, 94, 95]; (3) applies statistical functions on top of temporal LLDs [96, 85]; (4) follows a multi-step process for building the emotion recognition system [94, 95, 92]; (5) makes assumptions about frame-level emotional labels and/or dynamics of emotion [94, 95, 65, 92]. In contrast, the approach that we describe in this chapter does not do any of the aforementioned points.

3.3 Model

Motivated by architectures used in the the field of sentence classification (e.g., [91]), where the goal is to predict the class of a given variable length sentence (e.g., positive/negative review), we build a simple four-layer CNN for SER (Figure 3.1). Our model has four major components: (1) convolutional layer; (2) max-pooling over time layer; (3) dense layer; and (4) softmax layer. The convolutional layer identifies emotionally salient regions within variable length utterances and creates a sequence of feature maps. The max-pooling over time layer propagates features with the highest value to the dense layer. The max-pooling over time layer induces time invariance and

creates a fixed-size feature vector from a variable length input. Finally, the dense and softmax layers provide further modeling and prediction. We describe each component in more detail in this section.

Let $\mathbf{x}_i^u \in \mathbb{R}^d$ be a d dimensional feature vector available at frame i of an utterance u. Then, we represent an utterance u with T frames as:

$$\mathbf{X}^u = [\mathbf{x}_1^u, \mathbf{x}_2^u, \ldots, \mathbf{x}_T^u]$$

note that d is fixed while T varies across utterances. A temporal convolution operation applies a filter $\mathbf{w} \in \mathbb{R}^{d \times s}$, where s is the width of the filter, to produce a new feature set of length $T - s + 1$. So convolving filter $\mathbf{w}$ with $\mathbf{X}^u$ yields:

$$\mathbf{c}^u = \left[c_1^u, c_2^u, \ldots, c_{T-s+1}^u\right]$$

where each $c_i^u \in \mathbb{R}$ is obtained using the following operation:

$$c_i^u = \sum_{m=1}^{s} \sum_{n=1}^{d} \left(\left[\mathbf{x}_i^u, \ldots, \mathbf{x}_{i+s-1}^u\right] \odot \mathbf{w}\right)_{m,n}$$

where $\odot$ denotes the element-wise multiplication operation. We leave out the bias term in the above equation for simplicity.

The convolution operation allows the network to extract local features from an utterance. The width of the convolutional filters dictates the size of the region from which we create the feature maps. Wider filters capture long-term interactions while narrower filters capture short-term interactions. We can apply multiple filters, each with different weights, to extract different information from the same region. It is customary to apply a non-linearity activation function to the outputs of the convolution operation. We use the rectified linear unit (ReLU) in this work [97].

We follow the convolutional layer by a max-pooling over time operation. Given

a sequence of features, the max-pooling over time operation returns the maximum feature within that sequence. This ensures that only emotionally salient information is propagated. We follow the max-pooling layer by a dense layer and then by a softmax layer for prediction. The softmax layer takes a C-dimensional feature vector and outputs a C-dimensional probability distribution.

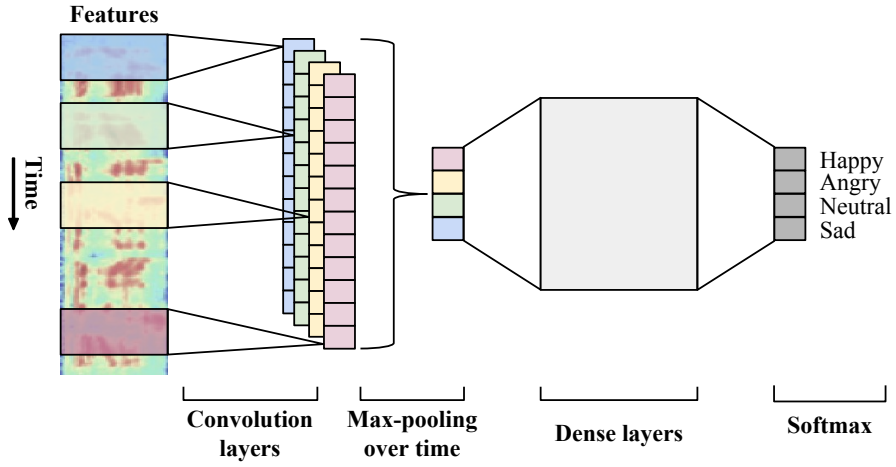


Figure 3.1: Network architecture used (four filters shown). The model takes in filter-bank representations of a variable-length utterance and predicts the emotion of that utterance.

3.4 Datasets and Recipe

3.4.1 Datasets

We evaluate our system on two emotion datasets: IEMOCAP [15] and MSP-IMPROV [80]. A description of the two datasets can be found in Chapter II. We use categorical evaluations with majority agreement for both datasets; and focus on four emotional categories: *Happy*, *Sad*, *Angry*, and *Neutral*. We include *excitement* utterances with *happiness* utterance for the IEMOCAP dataset to be consistent with previous work [85]. The final IEMOCAP dataset that we use in this chapter contains a total of 5531 utterances (1103 *Angry*, 1708 *Neutral*, 1084 *Sad*, 1636 *Happy*). The final MSP-IMPROV dataset that we use in this chapter contains a total of 7798

utterances (792 *Angry*, 3477 *Neutral*, 885 *Sad*, 2644 *Happy*).

3.4.2 Feature Extraction and Data Augmentation

We use the openSmile toolkit [98] to extract 40-dimensional log Mel-filterbank features (MFBs) from each utterance. We create our initial segments by sliding a Hamming window of width 25 milliseconds with an overlap of 10 milliseconds. We perform speaker-specific z-normalization on all features. Note that this normalization method assumes that we have access to enough samples, which represent all the emotion classes, from each speaker to compute the normalization parameters.

We increase the size of our training data by creating two different copies of each utterance following the approach described in [99]. In particular, for a given training utterance, We apply the *speed* effect found in the *Sox*[1] audio manipulation tool at factors of 0.9 and 1.1 to create two versions of the original utterance. We report the performance with and without augmentation in the results section.

3.4.3 Experimental Recipe

We follow a leave-one-speaker-out evaluation scheme for both datasets. In each session, we use utterances from one speaker for testing and utterances from the other speaker for validation and early stopping. We use utterances from all other speakers for training. This scheme allows using a validation speaker who has similar acoustic and recording conditions to those of the test speaker. We report the mean and standard deviation of the unweighted average recall (UAR) from all speakers. UAR is a popular metric used in SER because of imbalanced datasets.

We implement the network using the Keras deep learning library. In our experiments, we fix the dense network to have three layers with shape 1024:1024:4; and regularize the network using early stopping. The weights of the network are randomly

[1]http://sox.sourceforge.net

initialize following recommendation by He et al. [100]. We minimize the cross-entropy loss function using RMSprop [101] with an initial learning rate of 1e-4 and batches of samples with up to 50 samples. To create batches, we first edge-pad utterances so that they have lengths that are integer multiples of 32, and then group the resulting same-length utterances for batch training. To deal with class-imbalance, we scale the loss function using weights that are inversely proportional to class frequencies. For a given sample i, assume that $\mathbf{y}_i$ is the true label vector (all zeros but with a one at the correct class) and $\hat{\mathbf{y}}_i$ is the predicted probability distribution from the softmax layer, then the loss function takes the following form:

$$L_i = -w_i \sum_{j=0}^{C-1} y_{i,j} \log(\hat{y}_{i,j})$$

where C is the total number of classes and w_i is the scaling factor associated with sample i.

We compute the UAR on the validation set at the end of each epoch. If the UAR does not improve, then we restore the learned weights to their initial values at the beginning of the epoch and reduce the learning rate by 1.4. The process stops if the UAR does not improve for 10 consecutive epochs. For each setup, We train 10 models and average their predictions to account for randomness in initialization and training.

3.5 Experiments

We attempt to answer the following questions in our experiments: (1) does capturing regional information using CNNs provide an advantage over computing utterance-level statistics? (2) how does the performance of a system that focuses on emotionally salient regions compare to those of systems trained with popular large feature sets?

To answer the first question, we capture utterance-level features by applying the 12 IS09 statistical functions [39] to 40 MFBs to get fixed-length feature vector of size

480. We remove the convolutional component of the CNN and train the dense layers directly using the captured statistical features. The first row of Table 3.1 shows the results we obtain from training a dense network on utterance-level statistical functions.

Next, we train a CNN directly on temporal MFBs without applying any statistical functions. We vary the width of the filters from 8 to 128. To ensure a fair comparison, we adjust the number of filters in each setup such that the total number of learnable parameters are equal to those used in the dense network trained on utterance-level statistical features. Table 3.1 shows the results we obtain for different filter widths.

To answer the second question, we train a set of SVMs using popular feature sets. We extract IS09 [39], IS13 [40], GeMAPS and eGeMAPS [41] features. We apply the same 12 statistical functions to IS09, and IS13 LLDs. We use an RBF kernel and do a grid search using validation data to pick the optimal hyper-parameters in $C \in \{2^0, 2^2, \ldots, 2^{12}\}$, and $\gamma \in \{2^{-15}, 2^{-13}, \ldots, 2^{-3}\}$. We scale the SVM cost parameter to take class-imbalance into account. We use augmented data for all SVM experiments to ensure a fair comparison. Table 3.2 shows the results we obtain using different sets of features.

Next, we train a CNN that uses multi-width filters (8, 16, 32, 64) directly on temporal MFBs. Combining multiple widths allows the network to consider multiple contextual dependencies simultaneously. This approach showed promise in some sentence classification applications [102]. We use 384 filters for each width to set the total number of inputs to the dense layers to be equal to the total number of features we obtain from IS13 features. The first two rows of Table 3.2 shows the results we obtain from this setup.

Table 3.1: Regions vs. utterance-level statistics (40 MFBs) ("*" indicates $p < 0.05$ under paired t-test with first row)

Filter Width	UAR (%)	
	IEMOCAP	MSP-IMPROV
statistics	58.7 ± 3.0	49.8 ± 4.7
8	60.3 ± 3.8	50.2 ± 3.7
16	60.9 ± 3.4*	50.5 ± 3.5
32	60.5 ± 3.1*	50.4 ± 2.9
64	61.0 ± 3.4*	50.2 ± 3.9
128	59.2 ± 2.8	48.0 ± 3.7

Table 3.2: System performance comparison ("*" indicates $p < 0.05$ under paired t-test with first row)

Method	UAR (%)	
	IEMOCAP	MSP-IMPROV
CNN + 40 MFBs	61.9 ± 2.7	52.6 ± 3.8
CNN + 40 MFBs (no aug)	60.7 ± 3.0	49.8 ± 2.9*
SVM + IS09	61.2 ± 3.6	53.3 ± 5.0
SVM + IS13	62.0 ± 3.7	53.8 ± 6.0
SVM + GeMAPS	59.2 ± 4.0*	52.1 ± 4.7
SVM + eGeMAPS	60.0 ± 3.7	52.4 ± 5.0

3.6 Results

Table 3.1 shows that focusing on regional information when training a network is better than training a network using features obtained from statistical functions. When focusing on regional content, We see a significant improvement ($p < 0.05$) of 2.3% on IEMOCAP and a minor improvement of 0.7% on MSP-IMPROV over results of networks that relies on utterance-level statistics.

Table 3.2 shows that a network that combines multi-width filters that is trained using temporal MFBs yields UARs that are statistically comparable ($p \geqslant 0.05$) to

those obtained from SVMs trained using current widely used feature sets (with the exception of SVM + GeMAPS for IEMOCAP). Our results suggest that CNNs with MFBs can be used as replacement for traditional SVMs with hand-engineered features for SER. Table 3.2 also shows that augmenting the dataset using speed perturbation gives an improvement of 1.2% on IEMOCAP and a significant improvement ($p < 0.05$) of 2.8% on MSP-IMPROV over UARs obtained from non-augmented data.

The SVM + IS13 setup yields the highest UARs for both datasets (though not significantly higher than our results). IS13 contains a total of 1560 (130×12) features. These features include spectral, energy, and voicing features. In contrast, our system only uses 40 MFBs as features.

Xia et al. [85] obtained a UAR of 62.4% on IEMOCAP after training a deep neural network using 1582 hand-engineered features and utilizing a multi-task learning approach to incorporate more data. In contrast, our system is simpler, requires minimal feature engineering, and is trained in an end-to-end fashion.

3.7 Conclusion

This chapter demonstrated how taking utterance-level statistics (i.e., the temporal pooling operation during the feature extraction process) can obfuscate the emotion content of an utterance by washing out information from emotion-discriminative frames with information from frames that do not carry emotion information. In addition, this chapter demonstrated that speed perturbation can be an effective data augmentation technique for alleviating the data sparsity challenge in SER tasks.

CHAPTER IV

Pooling Acoustic and Lexical Features for SER

4.1 Introduction

In this chapter, we explore deep learning architectures for multimodal speech emotion recognition (SER) that use both linguistic and paralinguistic features. Conventionally, multimodal fusion in deep learning uses pooling techniques to combine representations from different modalities to form a joint multimodal representation [103]. However, it is unclear which pooling technique is most effective for combining acoustic and lexical feature for the task of valence prediction. To this end, we investigate different pooling strategies that can be used to combine information from these two modalities.

Previous work showed that systems that incorporate both acoustic and lexical features are more accurate than those that only incorporate features from one modality [74, 75]. Traditionally, these multimodal approaches rely upon either early-fusion or late-fusion [76]. In late-fusion, a model is independently built for each modality, and decisions are generated from these independent models. These decisions are then combined to make a final decision. In early-fusion, multimodal feature vectors are created by combining the feature vectors from each modality. These augmented feature vectors are then used to learn a model. Early-fusion allows a model to consider low-level interactions between features from multiple modalities when making a pre-

diction. However, these approaches assume a level of temporal synchrony between the individual modalities, which may not be valid. This is in contrast to late-fusion, where individual models consider features from only one modality, obscuring time-varying properties but alleviating the assumption of time-synchrony.

In this chapter, we investigate approaches for pooling representations from the acoustic and lexical modalities in neural networks for the end goal of making valence predictions. The pooling strategies that we investigate include element-wise summation, element-wise multiplication, concatenation, and outer-product. In addition, we also experiment with the multimodal compact bilinear pooling (CBP) approach [103], which provides a method for reducing the number of parameters obtained from a regular outer-product. Outer-product-based methods for pooling features allow the model to consider more expressive interactions between the features from the two modalities [103]. This is due to the fact that taking the outer-product allows all pairs of features from the two vectors to interact. Such methods showed success in computer vision applications [104, 103], but their use has not been investigated in linguistic and paralinguistic tasks.

4.2 Related Work

Li et al. [105] and Poria et al. [76] used models that were trained independently on different modalities as feature extractors. Li et al. applied a maximum entropy classifier to predict the speakers' stance in idealogical debates given lexical and acoustic features extracted from separately trained models. Poria et al. used lexical features extracted from a convolutional neural network along with manually extracted acoustic and visual features to perform multimodal sentiment predictions using a multiple kernel learning (MKL) classifier. Poria et al. experimented with both early-fusion and late-fusion methods and showed that early-fusion was more effective. Both works showed that models that used multimodal features performed better than those that

only used unimodal features. In contrast, the model presented in this chapter is trained in an end-to-end fashion, avoiding the need for training different parts separately. The model is trained to jointly extract representations from the different modalities under one loss function.

Perez-Rosas et al. [106], Jin et al. [74], and Brilman et al. [75] all extracted high-level knowledge-based features to be used in a support vector machine (SVM) classifier. Perez-Rosas et al. looked at the problem of multimodal sentiment analysis in YouTube video reviews using acoustic, visual, and bag-of-words textual features to find that multimodal systems outperform unimodal ones. Jin et al. used OpenSmile [107] and bag-of-words features to recognize emotions and compare the early- and late-fusion methods to find that late-fusion performs best. Finally, Brilman et al. extracted a comprehensive set of multimodal features, and then performed an analysis to identify features that are most indicative of successful debate performance. Brilman et al. showed that the audio modality was most predictive and a multimodal system, via late-fusion, outperforms unimodal systems.

In contrast, in the work we present in this chapter, we do not rely on high-level features, the development of which requires expert-knowledge in speech and language processing. In addition, we consider neural approaches to multimodal modelings instead of SVM-based ones. The inputs to our model consist of frequency-domain representation of speech signals and `word2vec` feature representations. We also investigate different pooling strategies and their impact on overall performance.

4.3 Dataset and Features

Dataset. we use the IEMOCAP dataset in this study [15]. A description of the IEMOCAP dataset is provided in Chapter II of this dissertation document. Each utterance in IEMOCAP was labeled for both valence and arousal on a 5-point Likert scale by at least two distinct annotators. We use the 10,032 utterances that have

both the acoustic and lexical content. The IEMOCAP corpus is used in this study because: (1) at the time this study was conducted, IEMPCAP was one of the largest emotion datasets; (2) it provides both the `.wav` files and their associated transcripts; and (3) all utterances were recorded in English.

Labels. We convert the 5-point scale used for describing valence values to a 3-point scale following the approach described by Chang et al. [108]. This is done by pooling valence levels 1 and 2 into a single "low" value and pooling valence levels 4 and 5 into a single "high" value. We generate fuzzy labels for each utterance by representing the labels from each annotator as one-hot vectors and computing the mean over the vectors. For instance, if three annotators labeled an utterance $[0, 0, 1]$, $[0, 0, 1]$, and $[0, 1, 0]$ each, then the final label vector representation would be $[0, 0.3, 0.7]$ and the correct class label would be 2 (where the possible options are $\{0, 1, 2\}$). We treat the problem as a three-way classification problem, where the goal is to assign a label from {0, 1, 2} to a given utterance.

Acoustic Features. We extract 40 Mel-filterbank (MFB) features by sliding a 25 millisecond Hamming window with a step-size of 10 milliseconds. As a result, each utterance is represented as a sequence of 40-dimensional feature vectors. MFBs have shown success in many speech processing applications, including automatic speech recognition and SER [109, 110].

Lexical Features. We represent each word in the dataset as a 300-dimensional vector using a pre-trained `word2vec` model.[1] `word2vec` representations have shown success in sentiment analysis tasks [91], which is closely related to the task of predicting valence in emotional speech. Thus, we expect `word2vec` representations to be useful for our task.

[1] `https://code.google.com/archive/p/word2vec/`

4.4 Methods

4.4.1 Architecture

The multimodal architecture that we use is shown in Figure 4.1. The hyper-parameters that we consider are shown in Table 4.1. The network architecture accepts two input streams, one for each modality. The acoustic input stream takes a sequence of 40-dimensional vectors, while the lexical input stream takes a sequence of 300-dimensional vectors.

Acoustic Input. We pass the sequence of acoustic features through five layers of 1D convolution and 1D max-pooling to reduce the temporal resolution of the acoustic input sequence by 2^5, in order to make training faster (since the acoustic features have a temporal resolution of 10 milliseconds). We then pass the resulting sequence to bidirectional gated recurrent unit (GRU) layers [111] for temporal modeling. Previous work showed that GRUs can have comparable performance to that of long short-term memory (LSTM) units while using fewer parameters [111]. One of the main differences between a GRU and an LSTM unit is that a GRU has only two gates (as opposed to three) and it does not contain internal memory cells. Given the output sequence representation from the GRU layers, we induce a fixed-length feature vector by averaging the sequential outputs as described in [42], since it was shown that this can result in better discrimination between emotions when compared to only considering the output of the last layer.

Lexical Input. we pass the sequence of lexical feature vectors through bidirectional GRU layers and then induce a fixed-length representation by taking the average as we did for the acoustic features. We do not pass the lexical features through initial convolution or pooling because sequences of lexical features are much shorter than those of acoustic features.

Multimodal Pooling. For the unimodal systems, we feed the output from the

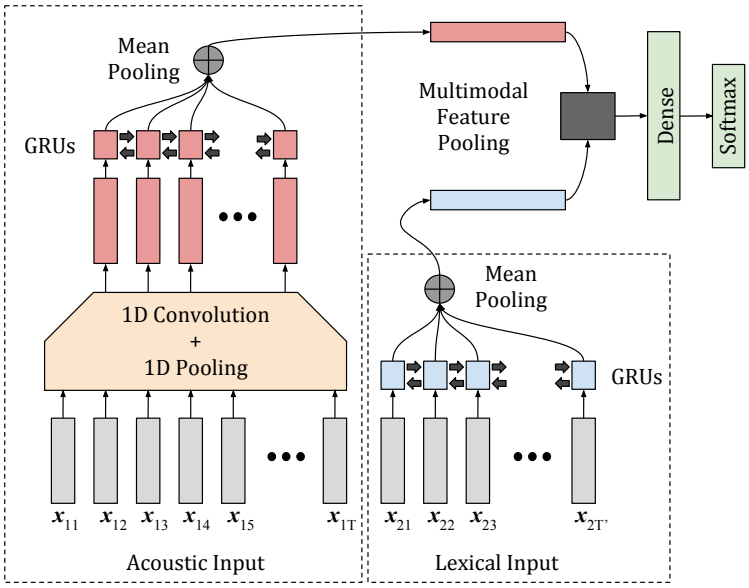


Figure 4.1: Overall network architecture. The network takes two input streams, one for each modality, and consists of three main components. One component for extracting features from acoustic features, another for extracting features from lexical features, and finally one for pooling the representations obtained from the two modalities.

average pooling layer to fully-connected layers before feeding them into a softmax layer (i.e., we skip the multimodal feature pooling step in Figure 4.1). For the multimodal systems, we pool the features obtained from the two modalities using the strategies described below and then feed the resulting features into fully-connected layers followed by a softmax layer.

4.4.2 Pooling Strategies

Given the representations for each modality, the next step is to pool these two representations to form a shared multimodal representation to be used for further modeling and prediction. We consider the following pooling strategies to combine the lexical and acoustic intermediate representations: (1) concatenation; (2) element-wise addition; (3) element-wise multiplication (Hadamard product); (4) outer-product; (5) compact bilinear pooling (CBP). Unlike traditional pooling methods, outer-product

Table 4.1: Hyper-parameters used in the validation process.

Hyper-parameter	Values
number of conv. kernels	$\{64, 128\}$
conv. kernel width	$\{2\}$
number of conv. layers	$\{5\}$
1D max-pooling kernel width	$\{2\}$
number of GRU layers	$\{1, 2\}$
GRU layers width	$\{32, 64\}$
number of dense layers	$\{0, 1\}$
dense layers width	$\{0, 128\}$
CBP output width	$\{256, 1024, 2048\}$

and CBP provide a more expressive way to consider the interactions between features from the two modalities. Taking the outer-product of two feature vectors considers the interactions between each pair of features from the two vectors. The problem with taking the outer-product, however, is the quadratic increase in the number of parameters. CBP [112] can be used to compress the results obtained from an outer-product. In particular, we utilize the multimodal variant of CBP [103], which makes taking the outer-product between multimodal vectors more feasible.

4.4.3 Compact Bilinear Pooling (CBP)

Given two input vectors, x and y, bilinear pooling is simply a linear transformation that considers all pairs of features from the two input vectors. Bilinear pooling can be obtained by first taking the outer-product of the two input vectors, $(x \otimes y)$, and then following it by a dense layer. CBP can be thought of as a sampling based approximation to bilinear pooling. The approximation is done using Tensor Sketch Projection [113, 114], and utilizes the property that $\Psi(x \otimes y, h, s) = \Psi(x, h, s) * \Psi(y, h, s)$, where Ψ is the projection function, h and s are vectors of randomly sampled parameters, and $*$ is the convolution operation. This property obviates the need for

computing outer-products of the two input vectors directly. The projection function is computed as follows: $\Psi(x, h, s)_i = \sum_{j:h_j=i}(s_j \cdot x_j)$, where $x, h, s \in \mathbb{R}^n$, h_j is sampled from $\{1, \ldots, d\}$, s_j is sampled from $\{-1, 1\}$, and d is the desired output dimension. In this work I use the CBP implementation by Ronghang Hu[2].

4.5 Experiments

4.5.1 Recipe

We follow a leave-one-speaker-out evaluation scheme. The dataset contains a total of five sessions, where each session has data from a male and a female speaker. This results in 10 unique speakers in total. For each fold, we use one speaker for testing and the other speaker within the same session for validation and early stopping. We use the remaining eight speakers for training.

We use unweighted average recall (UAR) and Pearson correlation (ρ) as our evaluation metrics. UAR is a popular metric used when dealing with imbalanced classes [115]. In cases where ground-truth labels have a tie, we accept predictions for either position as a correct answer. So if $[0, 0.5, 0.5]$ is the ground-truth label, then class labels 1 and 2 are considered correct predictions in the evaluation process. To compute Pearson correlation, we convert the network's output to numerical values by taking the expected value, similar to [108].

We implement the models using Keras [116] with a TensorFlow back-end [117]. We use RMSprop [101] to train the models and use a weighted cross-entropy loss function to account for class imbalance. We use fuzzy labels in the training process similar to [108], and run each experiment three times to account for random initialization of the parameters and report the ensemble performance. We sweep through hyper-parameters values shown in Table 4.1 and pick the combination that maximizes the

[2] `https://github.com/ronghanghu/tensorflow_compact_bilinear_pooling`

Table 4.2: Performance obtained using different pooling strategies. We assert significance when $p < 0.05$ under a paired t-test.

Method	UAR	ρ
unimodal—acoustic	.590	.320
unimodal—lexical	.648‡	.540‡
concatenation	.680†	.581†
summation	.683†	.578†
multiplication	.687†	.588†
outer-product	**.694**†	.601$^{\dagger *}$
CBP	.693†	**.605**$^{\dagger *}$

‡: significantly better than unimodal—acoustic
†: significantly better than unimodal—lexical and —acoustic
*: significantly better than concatenation

validation performance for each fold. We use an initial learning rate of 0.001. Starting from epoch five, we reduce the learning rate by half whenever the validation UAR does not improve at the end of each epoch.

4.5.2 Results

Table 4.2 shows the results for the different pooling strategies that we considered. The results show that the lexical modality yields significantly ($p < 0.05$) better performance than the acoustic modality does in terms of both UAR and ρ. This suggests that lexical cues are better for predicting valence than acoustic cues. The results show that multimodal systems significantly ($p < 0.05$) outperform the unimodal lexical systems, suggesting that adding the acoustic modality can still be beneficial. Pooling through element-wise multiplication provided a non-significant improvement in performance over element-wise summation and concatenation approaches. Outer-product methods provided significant improvement ($p < 0.05$) in ρ when compared to results from concatenation method. Finally, our results suggest that a CBP strategy does not provide an advantage over simple outer-product strategy. This is probably

Table 4.3: Confusion matrices comparison. Columns represent predictions while rows represent ground-truth.

(a) Lexical modality

	neg	neu	pos
neg	**.607**	.140	.253
neu	.233	**.572**	.194
pos	.128	.115	**.757**

(b) Acoustic modality

	neg	neu	pos
neg	**.509**	.181	.311
neu	.239	**.621**	.140
pos	.198	.164	**.638**

(c) Multimodal

	neg	neu	pos
neg	**.705**	.144	.151
neu	.247	**.648**	.104
pos	.148	.128	**.724**

due to the relatively low dimensionality of our multimodal representations required for each modality ($32 - 64$ for each modality).

Table 4.3 shows the confusion matrices obtained from the two unimodal systems and the CBP model. The results in Table 4.3 suggest that the acoustic modality is better for predicting neutral valence than the lexical modality. On the other hand, our results suggest that the lexical modality is better for predicting positive/negative valence than the acoustic modality. Finally, Table 4.3 shows that the significantly improved performance of CBP over that of the unimodal systems is due to more accurate negative and neutral valence predictions.

4.5.3 Analysis

The model that we use in this work abstracts the influence of individual modalities on the final decision. To further analyze the influence of each modality on the overall performance of our multimodal system, we study the effect of perturbing the individual input streams by adding white Gaussian noise (with zero mean and varying standard deviation) to the input features with different signal-to-noise-ratio (SNR) levels. We run this analysis on our best performing system, the CBP multimodal

system, and vary the SNR levels from -18 dB to 6 dB. We also include SNR values of -Inf dB and +Inf dB in our analysis. The idea is that if an input modality is less important, then perturbing its values with noise will have minimal effect on the overall performance.

Figure 4.2 shows the results that we obtain for this analysis. The figure shows that adding more noise to the lexical modality (dashed line) results in a rapid drop in performance compared to the performance drop due to adding noise to the acoustic modality (solid line). This suggests that the lexical modality has larger influence on the overall performance of the system. The figure shows that a multimodal system would still result in $> 60\%$ UAR even when SNR is zero for the acoustic modality.

4.6 Conclusion

There are several strategies that can be used to pool representations learned for acoustic and lexical modalities in neural networks. In this chapter, we presented a comparison between different multimodal feature pooling strategies for the task of predicting valence in emotional speech. Our results on the IEMOCAP dataset suggest the following: (1) multimodal methods that combine acoustic and lexical features are better than unimodal for predicting valence; (2) lexical modality is better for predicting valence than the acoustic modality; and (3) outer-product-based pooling strategies outperform other pooling techniques.

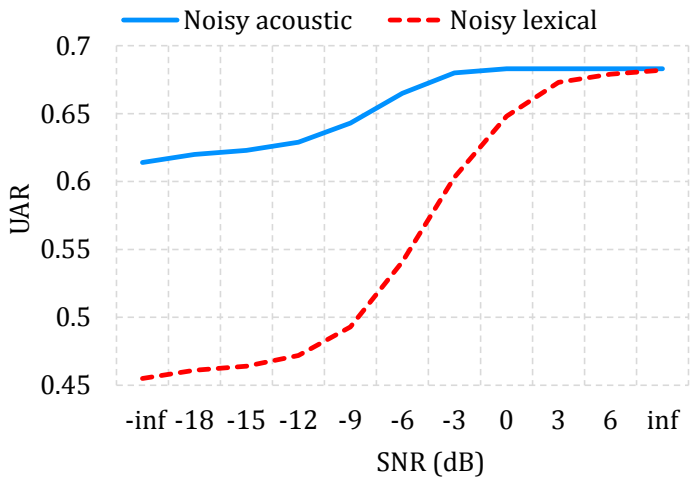


Figure 4.2: Effect of adding noise to each modality (while keeping the other modality clean) on the performance of CBP multimodal system.

CHAPTER V

Capturing Long-term Dependencies for SER

5.1 Introduction

In this chapter, we focus on problems where the goal is to recognize emotions in the valence-arousal space, continuously in time. The valence-arousal space is a psychologically grounded method for describing emotions [118]. Valence ranges from negative to positive, while activation ranges from calm to excited. Previous research has demonstrated that it is critical to incorporate long-term temporal information for making accurate emotion predictions. For instance, Valstar et al. [82] showed that it was necessary to consider larger windows when making frame-level emotion predictions (four seconds for arousal and six seconds for valence). Le et al. [65] and Cardinal et al. [119] found that increasing the number of contextual frames when training a deep neural network (DNN) for making frame-level emotion predictions is helpful but only to a certain point. Bidirectional long short-term memory networks (BLSTMs) can naturally incorporate long-term temporal dependencies between features; explaining their success in continuous emotion recognition tasks (e.g., [120]).

In this chapter, we investigate two convolutional network architectures, dilated convolutional networks and downsampling/upsampling networks, that capture long-term temporal dependencies. We interpret the two architectures in the context of continuous emotion recognition and show that these architectures can be used to

build accurate continuous emotion recognition systems.

5.2 Related Work

Even though the problem of emotion recognition has been extensively studied in the literature, we only focus on works that predicted dimensional values, continuously in time. Successful attempts to solving the continuous emotion recognition problem relied on DNNs [119], BLSTMs [120], and more commonly, support vector regression (SVR) classifiers [121]. With the exception of BLSTMs, such approaches do not incorporate long-term dependencies unless coupled with feature engineering. In this chapter, we show that purely convolutional neural networks can be used to incorporate long-term dependencies and achieve good emotion recognition performance, and are more efficient to train than their recurrent counterparts.

In their winning submission to the AVEC 2016 challenge, Brady et al. [121] extracted a set of audio features (Mel-frequency cepstral coefficients, shifted delta cepstral, prosody) and then learned high-level representations of the features using sparse coding. The high-level audio features were used to train linear SVRs. Povolny et al. [122] used eGeMAPS [41] features along with a set of high-level bottleneck features extracted from a DNN trained for automatic speech recognition (ASR) to train linear regressors. The high-level features were produced from an initial set of 24 Mel-filterbank (MFB) features and four different estimates of the fundamental frequency (F0). Povolny et al. used all features to train linear regressors to predict a value for each frame, and considered two methods for incorporating contextual information: simple frame stacking and temporal content summarization by applying statistics to local windows. In contrast, in this chapter we show that considering temporal dependencies that are longer than those presented in [121, 122] is critical to improve continuous emotion recognition performance.

He et al. [120] extracted a comprehensive set of $4,684$ features, which included

energy, spectral, and voicing-related features, and used them to train BLSTMs. The authors introduced delay to the input to compensate for human evaluation lag and then applied feature selection. The authors ran the predicted time series through a Gaussian smoothing filter to produce the final output. In this chapter, we show that it is sufficient to use 40 MFBs to achieve state-of-the-art performance, without the need for special handling of human evaluation lag.

Trigeorgis et al. [49] trained a convolutional recurrent network for continuous emotion recognition using the time domain signal directly. The authors split the utterances into five-second segments for batch training. Given an output from a the trained model, the authors applied a chain of post-processing steps (median filtering, centering, scaling, time shifting) to get the final output. In contrast, we show that convolutional networks make it possible to efficiently process full utterances without the need for segmenting. Further, since the proposed models work on full-length utterances, we show that it is not necessary to apply any post-processing steps as described in [49].

On the ASR end, Sercu et al. [123] proposed viewing ASR problems as dense prediction tasks, where the goal is to assign a label to every frame in a given sequence, and showed that this view provides a set of tools (e.g., dilated convolutions, batch normalization, efficient processing) that can improve ASR performance. The authors argued that ASR approaches required practitioners to splice their input sequences into independent windows, making the training and evaluation procedures cumbersome and computationally inefficient. In contrast, the authors' proposed approach allows practitioners to efficiently process full sequences without requiring splicing or processing frames independently. The authors showed that their approach obtained the best published single model results on the switchboard-2000 benchmark dataset.

In this chapter, we treat the problem of continuous emotion recognition as a dense prediction task and show that, given this view of the problem, we can utilize convolu-

tional architectures that can efficiently incorporate long-term temporal dependencies and provide accurate emotion predictions.

5.3 Problem Setup

We focus on the RECOLA database [81] following the AVEC 2016 guidelines [82]. A description of the RECOLA database is provided in Chapter II of this dissertation. The RECOLA database consists of spontaneous interactions in French and provides continuous, dimensional (valence and arousal) ground-truth descriptions of emotions. Even though the AVEC 2016 challenge is multimodal in nature, we only focus on the speech modality in this chapter. The RECOLA database contains a total of 27 five-minute utterances, each from a distinct speaker (9 train; 9 validation; 9 test). Ground-truth continuous annotations were computed, using audio-visual cues, on a temporal granularity of 40 milliseconds from six annotators (three females).

Features. We use the Kaldi toolkit [124] to extract 40-dimensional log MFB features, using a window length of 25 milliseconds with a hop size of 10 milliseconds. Previous work showed that MFB features are better than conventional Mel-frequency cepstral coefficients (MFCCs) for predicting emotions [110]. We perform speaker-specific z-normalization on all extracted features. RECOLA provides continuous labels at a granularity of 40 milliseconds. Thus, we stack four subsequent MFB frames to ensure correspondence between hop sizes in the input and output sequences.

Problem Setup. Given a sequence of stacked acoustic features $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_T]$, where $\boldsymbol{x}_t \in \mathbb{R}^d$, the goal is to produce a sequence of continuous emotion labels $\boldsymbol{y} = [y_1, y_2, \ldots, y_T]$, where $y_t \in \mathbb{R}$.

Evaluation Metrics. Given a sequence of ground-truth labels $\boldsymbol{y} = [y_1, y_2, \ldots, y_T]$ and a sequence of predicted labels $\hat{\boldsymbol{y}} = [\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_T]$, we evaluate the performance using the root mean squared error (RMSE) and the Concordance Correlation Coefficient (CCC) to be consistent with previous work. The CCC is computed as follows:

$$\text{CCC} = \frac{2\sigma_{y\hat{y}}^2}{(\sigma_y^2 + \sigma_{\hat{y}}^2 + (\mu_y - \mu_{\hat{y}})^2)}$$

where $\mu_y = \mathbb{E}(\boldsymbol{y})$, $\mu_{\hat{y}} = \mathbb{E}(\hat{\boldsymbol{y}})$, $\sigma_y^2 = \text{var}(\boldsymbol{y})$, $\sigma_{\hat{y}}^2 = \text{var}(\hat{\boldsymbol{y}})$, and $\sigma_{y\hat{y}}^2 = \text{cov}(\boldsymbol{y}, \hat{\boldsymbol{y}})$.

5.4 Preliminary Experiment

We first study the effect of incorporating temporal dependencies of different lengths. The network that we use in the preliminary experiments consists of a convolutional layer with one filter of variable length from 2 to 2048 frames, followed by a $Tanh$ non-linearity, followed by a linear regression layer. We vary the length of the filter and validate the performance using CCC. We train the model on the training partition and evaluate on the development partition. We report the results of this preliminary experiment in Figure 5.1. The results show that incorporating long-term temporal dependencies improves the performance on the validation set up to a point.

The observed diminishing gains in performance past 512 (20.48 seconds) frames may occur either due to the increased number of parameters or because contextual information becomes irrelevant after 512 frames. Covering contexts as large as 512 frames still provided improvements in performance compared to results obtained from covering smaller contexts. The utility of contexts spanning 512 frames (20.48 seconds) is contrary to previous work that considered much smaller time scales. For instance, Valstar et al. [82] only covered six seconds worth of features and Povolny et al. [122] considered a maximum of eight seconds worth of features. Results from the preliminary experiment suggest that continuous emotion prediction systems could benefit from incorporating long-term temporal dependencies. This acts as a motivation for using architectures that are specifically designed for considering long-term dependencies.

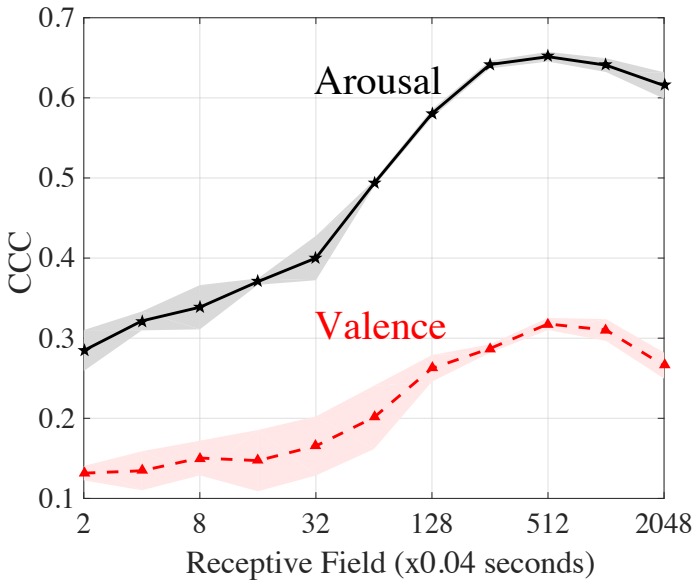


Figure 5.1: Increasing the size of the receptive field improves performance for both arousal and valence. Solid lines represent mean CCC from 10 runs and shaded area represents standard deviation from the runs.

5.5 Methods

In this section, we describe the two architectures that we propose to use to capture long-term temporal dependencies in continuous emotion prediction tasks.

5.5.1 Dilated Convolutions

Dilated convolutions provide an efficient way to increase the receptive field without causing the number of learnable parameters to vastly increase. Networks that use dilated convolutions have shown success in a number of tasks, including image segmentation [125], speech synthesis [126], and ASR [123].

van den Oord et al. [126] recently showed that it is possible to use convolutions with various dilation factors to allow the receptive field of a generative model to grow exponentially in order to cover thousands of time steps and synthesize high-quality speech. Sercu et al. [123] showed that ASR could benefit from dilated convolutions since they allow larger regions to be covered without disrupting the length of the

input signals. Continuous emotion recognition could benefit from such properties.

When compared to filters of regular convolutions, those of dilated convolutions touch the input signal every k time steps, where k is the *dilation factor*. If $[w_1, w_2, w_3]$ is a filter with a dilation factor of zero, then $[w_1, 0, w_2, 0, w_3]$ is the filter with a dilation factor of one and $[w_1, 0, 0, w_2, 0, 0, w_3]$ is the filter with a dilation factor of two, and so on. We build a network that consists of stacked convolution layers, where the convolution functions in each layer use a dilation factor of 2^n, where n is the layer number. This causes the dilation factors to grow exponentially with depth while the number of parameters grows linearly with depth. Figure 5.2 shows a diagram of the dilated convolution network.

5.5.2 Downsampling/Upsampling

The emotion targets in the RECOLA database are sampled at a frequency of 25 Hz. Using Fourier analysis, we find that more than 95 percent of the power of these trajectories lies in frequency bands that are lower than 1 Hz. In other words, the output signals are smooth and they have considerable time dependencies. This finding is not surprising because we do not expect rapid reactions from human annotators. Networks that use dilated convolutions do not take this fact into account while making predictions, causing them to generate output signals whose variance is not consistent with the continuous ground truth contours (Section 5.6.2). To deal with this problem, we propose the use of a network architecture that compresses the input signal into a low-resolution signal through downsampling and then reconstructs the output signal through upsampling. Not only does the downsampling/upsampling architecture capture long-term temporal dependencies, it also generates a smooth output trajectory.

We conduct an experiment to investigate the effect of downsampling/upsampling on continuous emotion labels. First, we convert the ground truth signals to low-

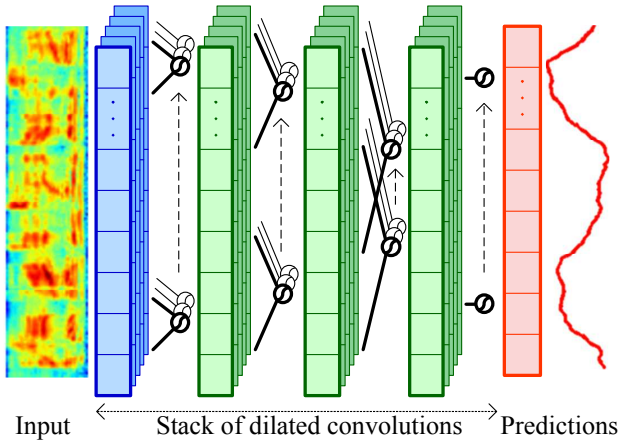


Figure 5.2: A visualization of the dilated convolution network. We use convolutions with a different dilation factor for different layers. We use a 1×1 convolution for the last layer to produce the final output.

resolution signals using standard uniform downsampling. Given the downsampled signals, we then generate the original signals using spline interpolation. We vary the downsampling factor exponentially from 2 to 128 and compute the CCC between the original signals and the reconstructed ones. The results that we show in Figure 5.4 demonstrate that distortions caused by downsampling with factors up to 64 are minor ($< 5\%$ loss in CCC relative to original).

The network that we use contains two subnetworks: (1) a downsampling network; (2) an upsampling network. The downsampling network consists of a series of convolutions and max-pooling operations. The max-pooling layers reduce the resolution of the signal and increase the effective receptive field of the convolution layers. Initial experiments showed that max-pooling was more effective than other pooling techniques.

The upsampling function can be implemented in a number of ways [127]. In this chapter we use the transposed convolution[1] [128, 129] operation to perform upsampling. Transposed convolutions provide a learnable map that can upsample a low-resolution signal to a high-resolution one. In contrast to standard convolution

[1]Other names in literature include deconvolution, upconvolution, backward strided convolution and fractionally strided convolution.

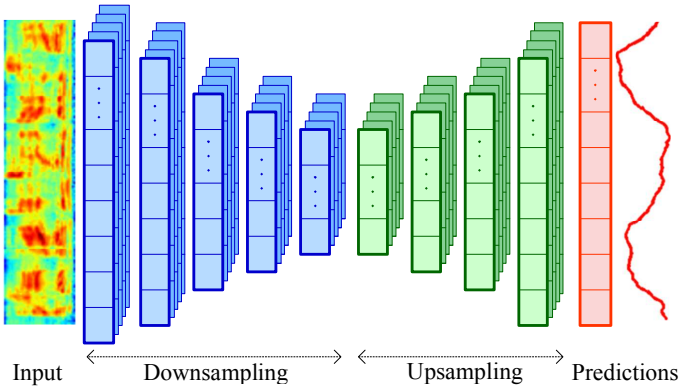


Figure 5.3: A visualization of the downsampling/upsampling network. Downsampling compresses the input signal into shorter signal which is then used to reconstruct a signal of the same length by the upsampling sub-network. We use the transpose convolution operation to perform upsampling.

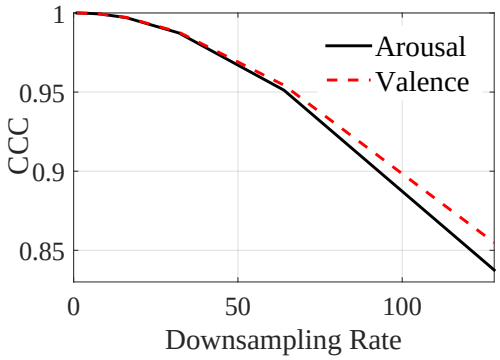


Figure 5.4: Effect of downsampling/upsampling on CCC.

filters that connect multiple input samples to a single output sample, transposed convolution filters generate multiple outputs samples from just one input sample. Since it generates multiple outputs simultaneously, the transposed convolution can be thought of as a learnable interpolation function.

Downsampling/upsampling architectures have been used in many computer vision tasks (e.g., [127, 130, 131]). For instance, Noh et al. [127] showed that transposed convolution operations can be effectively applied to image segmentation tasks. In addition to vision applications, downsampling/upsampling architectures have been successfully applied to speech enhancement problems [132], where the goal is to learn a mapping between noisy speech spectra and their clean counterparts. Park

et al. [132] demonstrated that downsampling/upsampling convolutional networks can be 12× smaller (in terms of the number of learnable parameters) than their recurrent counterparts and yet yield better performance on speech enhancement tasks.

The main goal of a transposed convolution is to take an n_x-dimensional low-resolution vector $\boldsymbol{x}$ and generate an n_y-dimensional high-resolution vector $\boldsymbol{y}$ using an n_w-dimensional filter $\boldsymbol{w}$ (where $n_y > n_x$). Similar to other linear transforms, $\boldsymbol{y}$ can be expressed as:

$$\boldsymbol{y} = \boldsymbol{T}\boldsymbol{x}$$

where $\boldsymbol{T}$ is the linear n_y-by-n_x transform matrix that is given by

$$\boldsymbol{T} = [\boldsymbol{T}_1, \boldsymbol{T}_2, ..., \boldsymbol{T}_{n_x}]$$

$\boldsymbol{T}_i$ is the i-th column of $\boldsymbol{T}$ and can be written as:

$$\boldsymbol{T}_i = [\underbrace{0, ..., 0}_{s(i-1)}, \underbrace{\boldsymbol{w}^T}_{n_w}, \underbrace{0, ..., 0}_{s(n_x-i)}]^T$$

where s is the upsampling factor. This linear interpolator is able to expand the input vector $\boldsymbol{x}$ to the output vector $\boldsymbol{y}$ with the length of $n_y = s(n_x - 1) + n_w$. Note that the matrix $\boldsymbol{T}$ is nothing but the transposed version of the standard strided convolution transform matrix. our experiments confirm that the proposed downsampling/upsampling network generates smooth trajectories.

5.6 Results and Discussion

5.6.1 Experimental Setup

We build the proposed models using the Keras library [116] with a Theano backend [133]. We train our models on the training partition of the dataset and use the

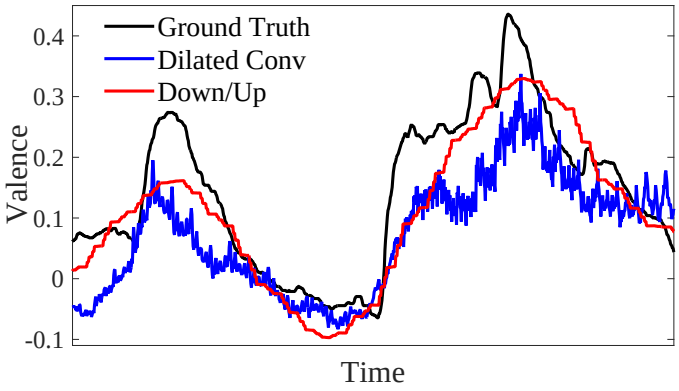


Figure 5.5: A visualization of the predictions produced by the two models plotted against ground-truth for a 40-second segment.

development partition for early stopping and hyper-parameter selection (e.g., learning rate, number of layers layer size, filter width, $L2$ regularization, dilation factors, downsampling factors). We optimize the CCC objective directly in all setups. We repeat each experiment five times to account for the effect of initialization. The final test evaluation is done by the AVEC 2016 organizers (i.e., the test set is withheld by the competition organizers). Test submissions were created by averaging the predictions produced from the five runs to account for randomness from initialization and training.

We report published results from the literature as baselines. Almost all previous works only report their final test results based on multimodal features. We only show results that are reported on the audio modality in the results tables. We also compare our performance to that of an optimized BLSTM regression model, described in [134]. our final dilated convolution structure has a depth of 10 layers, each having a width of 32. Our final downsampling/upsampling network contains four downsampling layers, one intermediate layer, and four transposed convolution layers, each having width of 32 for arousal and 128 for valence. We use a downsampling factor of three. We do not splice the input utterances into segments; instead, we train on full length utterances and use a batch size of one.

5.6.2 Results

Tables 5.1 and 5.2 show the development and test results for arousal and valence, respectively. Each row shows the results for one setup. We only include results from the literature that are based on the speech modality and use "–" to show unreported results.

Both proposed systems show improvements over baseline results by Valstar et al. [82]. The proposed dilated convolution based system provides improvements of 5.6% and 19.5% over baseline systems for arousal and valence, respectively. The proposed downsampling/upsampling system provides improvements of 5.1% and 33.9% over baseline systems for arousal and valence, respectively. We report the results we obtain from the proposed BLSTM system to provide a reference point. The proposed BLSTM system performs well when compared to the baseline results.

The proposed methods outperform BLSTMs and are more efficient to train on long utterances. For instance, given a convolutional network and a BLSTM network with approximately equal number of learnable parameters, one epoch of training on the AVEC dataset takes about 13 seconds on the convolutional network while one epoch of training takes about 10 minutes on the BLSTM network. This suggests that convolutional architectures can act as replacement for recurrent ones for continuous emotion recognition problems.

We show an example 40-second segment of the predictions made by our two networks along with the ground-truth predictions in Figure 5.5. The figure shows that the predictions produced by the downsampling/upsampling network are much smoother than those produced by the dilated convolution networks. We believe that the structure of the downsampling/upsampling network forces the output to be smooth by generating the output from a compressed signal. The compressed signal only stores essential information that is necessary for generating trajectories, removing any noise components.

Table 5.1: Arousal results.

Method	Dev.		Test	
	RMSE	CCC	RMSE	CCC
Valstar et al. [82]	–	.796	–	.648
Brady et al. [121]	.107	.846	–	–
Povolny et al. [122]*	.114	.832	.141	.682
BLSTM [134]	.103	.853	.143	.664
Dilated	.102	.857	**.137**	**.684**
Down/Up	**.100**	**.867**	**.137**	.681

Table 5.2: Valence results.

Method	Dev.		Test	
	RMSE	CCC	RMSE	CCC
Valstar et al. [82]	–	.455	–	.375
Brady et al. [121]	.132	.450	–	–
Povolny et al. [122]*	.142	.489	.355	.349
BLSTM [134]	.113	.518	**.116**	.499
Dilated	.117	.538	.121	.486
Down/Up	**.107**	**.592**	.117	**.502**

5.7 Conclusion

We investigated two architectures that provide different means for capturing long-term temporal dependencies in a given sequence of acoustic features. Dilated convolutions provides a method for incorporating long-term temporal information without disrupting the length of the input signal by using filters with varying dilation factors. Downsampling/upsampling networks incorporate long-term dependencies by applying a series of convolution and max-pooling operations to downsample the signal and

*Unpublished test results, courtesy of the authors.

get a global view of the features. The downsampled signal is then used to reconstruct an output with a length that is equal to the uncompressed input. When the experiment were conducted, the proposed methods achieved the best known audio-only performance on the AVEC 2016 challenge.

CHAPTER VI

Speaker Embeddings as Robust Features for SER

6.1 Introduction

In this chapter, we study the utility of speaker embeddings, representations extracted from a trained speaker recognition network, as robust features for detecting emotions. The features used to describe the acoustic signal are a crucial aspect of any emotion recognition model. Consequently, various features have been proposed in the literature for the task of SER (e.g., [39, 40, 41]). However, the extraction of many of these proposed features are susceptible to distortions due to variations in lexical content, the presence of environmental noise, or domain shifts. As a result, there remains a need for robust paralinguistic features that abstract extraneous low-level variations present in the acoustic signal and only capture speaker characteristics that are necessary for predicting emotions.

Previous research has shown that neural networks trained discriminatively on large and diverse datasets learn to extract robust features that are invariant to noise and domain-shifts (e.g., [78, 79]). These features are obtained from intermediate representations that the trained networks extract from the input signal. However, the main requirement for learning powerful features using neural networks is the access to large labeled datasets; a requirement that is still challenging to fulfill in the affective computing community in general, and in the emotion recognition community

in particular. The challenges associated with finding media sources that provide varied emotional data as well as the challenges associated with annotating the data with accurate emotion labels are the driving reasons behind the data sparsity problem in emotion recognition.

Many paralinguistic tasks are closely related, and thus, the representations extracted while solving one paralinguistic task can be used for solving other tasks [135, 55]. Specifically, previous research showed that representations learned while solving the emotion recognition task are useful for solving other paralinguistic tasks, such as gender detection and speaker identification [55]. However, unlike emotion recognition, speaker recognition does not suffer from the problem of data sparsity; there are multiple large-scale datasets with speaker labels (e.g., [136, 137, 138]). In this chapter, we ask if speaker recognition can help emotion recognition by attenuating the challenges that come with having limited amounts of labeled emotion data. We hypothesize that we can improve emotion recognition performance by leveraging speaker embeddings, feature representations trained for the speaker recognition task. Our work complements previous research by demonstrating that speaker embeddings can be used as a replacement to common paralinguistic features in emotion recognition applications.

We propose two experiments designed to study the relationship between emotion and speaker embeddings, and assess their utility as general paralinguistic features. In this experiment, we quantify the effect emotion has on speaker embeddings to examine whether or not the embeddings capture speech characteristics that are changed by emotion. We hypothesize that emotionally charged vocal expressions change speech characteristics that are captured by speaker embeddings (i.e., speakers sound less like themselves when their vocal expressions are emotionally charged). This hypothesis is supported by existing work, which studied the effect of emotion on speaker representations (e.g., i-vector), focusing on changes in the equal error rate (EER) in speaker verification tasks[139, 140, 141, 142]. However, the focus on the EER metric obfus-

cated the utility of speaker embeddings as paralinguistic features because the EER metric, as used in speaker verification tasks, measures the performance as a function of a general population of test speakers. In other words, although emotionally charged vocal expressions might change how identity is encoded for a certain speaker, individual speakers might still sound more like themselves when compared to a general population of other test speakers. In this chapter, we instead quantify the effect of emotion on speaker representations by hypothesizing that representations extracted from neutral speech, as a group, has more intra-group similarity, compared to the similarity between neutral and emotional speech. We test this hypothesis using a novelty detection framework, implemented using autoencoders, with reconstruction error as a proxy for similarity. The benefit to this paradigm is that it allows us to ask not whether the emotional speech belongs to a different speaker, but instead, if the differences in emotional speech are captured by speaker identification features. Our results suggest that emotional speech significantly changes speaker embeddings from their neutral representation, and that these changes can be utilized in a novelty detection framework for detecting non-neutral speech.

In the second experiment, we assess the effectiveness of speaker embeddings for detecting emotions by comparing them to state-of-the-art paralinguistic features. We expect speaker embeddings to be more robust to the variations introduced by domain shifts compared to common paralinguistic features used in the emotion recognition literature. This is because neural networks used for extracting the speaker embeddings are trained on large and in-the-wild datasets, which make the extracted embeddings invariant to changes in recording conditions and background noises. As a result, we expect speaker embeddings to capture high-level speaker characteristics that can be beneficial for recognizing emotions while abstracting any low-level variations present in the acoustic signal. We test this hypothesis by running both within-corpus and cross-corpus emotion recognition experiments. Cross-corpus setups make the emotion

recognition task more challenging as trained models cannot rely on spurious correlations that exist within a dataset to make predictions. Our results demonstrate that emotion recognition models that use speaker embeddings as features outperform those that use state-of-the-art paralinguistic features, especially in cross-corpus settings.

To summarize, the novelty of the work presented in this chapter is three-fold: (1) we demonstrate how speaker embeddings highlight the differences that exist between neutral speech and emotionally expressive speech; (2) we show how speaker embeddings can be used in a novelty detection framework for establishing a baseline of how a speaker sounds in the neutral state, and for detecting deviations from this baseline neutral state; and (3) we demonstrate how speaker embeddings provide a robust replacement to general paralinguistic features for recognizing emotional expression. The remainder of this chapter is organized as follows. Section 6.2 covers related work. Section 6.3 covers the proposed approach. Section 6.4 covers the datasets used in our work. Sections 6.5 and 6.6 cover the experiments and results. Finally, Section 6.7 includes concluding remarks and proposed future directions.

6.2 Related Works

6.2.1 Speaker Representations and Emotional Speech

There are several works that studied the relationship between speaker representations and emotional speech. In this section we cover works that looked at this relationship as it relates to traditional (e.g., i-vectors) and neural speaker representations. i-vectors are common representations used in speaker identification and verification applications [143]. They capture several sources of variation (e.g., identity, age, gender) present in the acoustic signal as represented by the Gaussian mixture model (GMM) mean supervector. More recently, neural representations have outperformed their i-vector counterparts (e.g., [144, 145, 146]). These representations are extracted

from the intermediate layers of a neural network that was discriminatively trained to classify speakers. Some common neural representations introduced in the literature include the d-vector and x-vector representations [145, 137, 146, 147, 145].

One question with these representations is how other modulations (e.g., emotion) change their ability to recognize speakers. Previous research used degradation in the EER metric in a speaker verification task as a proxy for quantifying the effect of emotion on speaker representations [141, 140, 142, 148]. However, one limitation with the use of the EER metric for this purpose is that it measures both the inter- and intra-speaker variations in the representations. In other words, the negative samples used when evaluating the EER for a speaker always came from a different speaker (i.e., a speaker is always compared to other speakers). So the metric will only be affected if the variations due to emotions are bigger than those due to changes in speaker identity. In contrast, we study how emotion modulates speaker representations by treating neutral speech from a given speaker as a group, and determining if this group has more intra-group similarity, compared to the similarity between neutral and emotional speech. The similarity measure is used as a proxy for the amount of modulation that emotion incurs on the speaker representations. The benefit of this approach is that it allows us to determine if differences in emotional speech are captured by speaker recognition features.

6.2.2 Speech Representations for Emotion Recognition

One of the most common paradigms for extracting acoustic features for emotion recognition involves two steps. First, low-level-descriptors (LLDs) are extracted using a short sliding window (e.g., extracted every 25 milliseconds) applied to the acoustic signal. Then, a set of statistical functions are applied to these LLDs to get a feature representation of an utterance. Some popular feature sets that were developed include the INTERSPEECH 2009 (IS09) Emotion Challenge features, the INTERSPEECH

2013 Computational Paralinguistics ChallengeE (ComParE), the Geneva Minimalistic Acoustic Parameter Set (GeMAPS), and the extended Geneva Minimalistic Acoustic Parameter Set (eGeMAPS) [39, 149, 40, 41]. The benefit of this paradigm is that it allows for a description of how properties of the low-level acoustic features change over the course of an utterance, obviating the need for a detailed focus on the short-time dynamical properties of the features.

Yet, the short-time dynamical properties of acoustic features convey critical cues into an individual's emotions. The work in Chapter III showed how the application of statistical functionals in the feature extraction process can obfuscate these cues, and has shown that modeling the acoustic features directly in neural networks can alleviate this problem. However, one challenge with using low-level acoustic features (e.g., MFCCs, pitch, etc.) to directly predict emotion is that their extraction can be significantly affected by variations in the recording conditions or variations in the lexical content of the utterance [150]. In other words, the features extracted from an utterance can look different depending on what the speaker said or depending on the environment of the speaker during the recording.

Representation learning, through the use of neural networks, has been shown to be an effective way to learn powerful features that are invariant to lexical content and recording conditions (e.g., [151, 152]). As a result, more recent approaches to emotion recognition from speech have focused on using neural networks to train recognition models that rely on minimally engineered features. These works have used spectrograms, filterbanks, or raw-waveforms for building emotion recognition models [42, 43, 44]. However, datasets used for building SER models remain significantly smaller than those used for building other speech models (e.g., speaker recognition). This hinders the ability of neural networks trained for the task of emotion detection to extract robust representations from the acoustic signal to be used in other domains or applications.

In this chapter, we show that features extracted from a neural network that was trained for speaker recognition can be used as general features for detecting emotions. We demonstrate how emotional expression modulates speaker embeddings from their neutral representation, and demonstrate how these modulations can be used for detecting emotional expression. Finally, we show that speaker embeddings can outperform traditional state-of-the-art features in challenging cross-corpus emotion recognition tasks.

6.3 Method

We propose the use of speaker embeddings as a replacement to traditional paralinguistic acoustic features for the recognition of emotions. We introduce speaker embeddings and the model used to extract them in this section.

6.3.1 Speaker Embeddings

Speaker embeddings are fixed-size vector representations of variable-length utterances. They are typically used in speaker recognition and diarization tasks [137, 145, 146], and can also be used for adapting acoustic models in automatic speech recognition systems [153]. The current standard for extracting robust speaker embeddings is by taking the outputs from an intermediate layer of a neural network that was discriminatively trained to identify speakers from a large set of individuals. Common speaker embeddings from the literature include d-vectors, x-vectors, and embeddings extracted from the VGG-M speaker identification network [137, 146, 145].

Speaker recognition neural networks map low-level acoustic features (e.g., Mel-filterbanks, MFCCs, spectrograms) extracted from utterances to speaker identities present in the training set. The representations (i.e., transformation) that such networks learn in the process can be used for extracting general embeddings to represent utterances from new speakers not seen in the training phase. These representations

encode speech characteristics needed for recognizing speakers but abstract low-level variations that are not needed for recognizing speakers.

6.3.2 x-vector Model

We focus our work on speaker embeddings extracted from the x-vector model as described in [151, 146]. We choose to work with the x-vector system because it has been demonstrated that it provides state-of-the-art embeddings for speaker recognition and diarization applications [146, 154, 155, 156], and because it is built on top of the open-source Kaldi toolkit [124]. The network used for extracting x-vectors is summarized in Table 6.1, and it consists of three parts: (1) frame-level feature extraction sub-network; (2) statistics pooling layer; and (3) and utterance-level classification sub-network.

The frame-level feature extraction sub-network takes in a sequence of 30-dimensional MFCC frames, where each frame represents 25 millisecond, and outputs a sequence of 512-dimensional features. It consists of five layers with a time-delay architecture. The first layer stacks the current frame at t with context frames from the previous two and the next two time steps. The second and third layers stack the current frame at t with context frames $t \pm 2$ and $t \pm 3$, respectively. The fourth and fifth layers do not add any context frames and only transform the representations at the current frame. The statistics pooling layer summarizes the frame-level features by taking the mean and standard deviation across the time dimension. Finally, the utterance-level classification sub-network consists of two fully-connected layers, and a softmax layer for classifying speakers.

Given a variable-length utterance by a speaker that was not seen in the training phase, a fixed-size representation for this utterance can be obtained by taking the output of the "segment6" layer (before the non-linearity) from the neural network summarized in Table 6.1. We use the outputs of "segment6" layer as our embeddings

for two reasons. First, previous research has suggested that they encode information relating to emotion, speaking style, and speaking rate [157, 154]. Second, previous research found that they are better equipped than other outputs for capturing speaker characteristics in speaker verification tasks [151].

6.4 Datasets

We use three emotion datasets in this study: IEMOCAP [15], MSP-IMPROV [80], and VESUS [83]. A description of the three datasets can be found in Chapter II. One limitation with both the IEMOCAP and MSP-IMPROV datasets is that they have very few utterances where the lexical content is the same but the emotion varies; making it difficult to study the influence of emotion and lexical variations on the embeddings independently. Studying these two variables independently is necessary since emotions modulate not only speech acoustics, but also language [52]. These modulations can influence the sequence of phonemes that are uttered, which can then affect the extracted speaker embeddings. Thus, we use the VESUS dataset to study the relationship between emotion and speaker embeddings (Section 6.5.1), and use the IEMOCAP and MSP-IMPROV datasets when running within-corpus and cross-corpus emotion recognition tasks (Section 6.5.2).

6.5 Experiments

This section describes the experiments used to assess the utility of speaker embeddings in emotion recognition tasks. The first experiment quantifies the effect of emotion variation on speaker embeddings; teasing out the effects on the embeddings due to emotion variations from those due to lexical variations. The second experiment compares the performance of an emotion recognition model trained and evaluated with speaker embeddings as features to the performance of recognition models

Table 6.1: The network architecture used in the speaker identification task taken from [151]. Speaker embeddings are extracted from the segment6 layer. N is the total number of speakers used in the training phase. T is the total number of frames in an utterances. The input size of 150 for the frame1 layer is the result of stacking five context frames, each with a size of 30. The input sizes of 1536 for the frame2 and frame3 layers are a result of stacking three context frames, each with a size of 512.

Layer	**Layer context**	**Total context**	**Input $\times$ Output**
frame1	$[t-2, t+2]$	5	150×512
frame2	$\{t-2, t, t+2\}$	9	1536×512
frame3	$\{t-3, t, t+3\}$	15	1536×512
frame4	$\{t\}$	15	512×512
frame5	$\{t\}$	15	512×1500
stats. pooling	$[0, T)$	T	$1500T \times 3000$
segment6	$\{0\}$	T	3000×512
segment7	$\{0\}$	T	512×512
softmax	$\{0\}$	T	$512 \times N$

trained and evaluated with state-of-the-art features used in the emotion literature.

The speaker embeddings that we use in all of our experiments were extracted using a pre-trained[1] x-vector model that was discriminatively trained to identify speakers in the combined VoxCeleb1 and VoxCeleb2 datasets [137, 138]. The combined VoxCeleb datasets contain more than 2,000 hours of speech (more than 1 million utterances) from more than 7,000 speaker identities. The x-vector model, summarized in Table 6.1, takes in the voiced frames of an utterance as an input and gives a speaker identity as an output. The input features to the x-vector model are 30-dimensional Mel-frequency cepstral coefficients (MFCCs) extracted from 16kHz utterances using a 25 millisecond sliding window. All utterances are mean normalized using a three-second window before being fed into the speaker identification network. A more detailed training recipe for the speaker identification network can be found in [146].

[1] `https://kaldi-asr.org/models/m7`

6.5.1 Experiment 1: Speaker Embeddings and Emotions

In this experiment, we quantify the effect emotion has on speaker embeddings to examine whether or not the embeddings are well-equipped for the emotion recognition task. We hypothesize that embeddings are modulated by emotional speech, allowing us to either preserve or enhance the differences that exist between a neutral expression and the expression of emotion. We formulate this problem by asserting that neutral speech, as a group, has more intra-group similarity, compared to the similarity between neutral and emotional speech. We test this hypothesis using a novelty detection framework, implemented using autoencoders, with reconstruction error as a proxy for similarity. The use of the reconstruction error of an autoencoder for novelty detection tasks has been studied for other applications by several works in the literature (e.g., [158, 159, 160, 161]). To the best of our knowledge, we are the first to propose the use of autoencoders to analyze the effect of the variations in emotion and in lexical content on speaker embeddings.

We address the following two questions about the relationship between speaker embeddings and emotion in our experiments:

- Q1: Do variations in emotion significantly modulate speaker embeddings from their neutral representation?
- Q2: Are the modulations on the neutral embeddings due to emotion variation larger or smaller than those due to lexical variation?

Answering these two questions is necessary for understanding how emotions affect speaker embeddings, and for assessing the embeddings' utility as general paralinguistic features in emotion recognition tasks.

We rely on two datasets to pre-train and evaluate our networks. All of the autoencoders that we use in this analysis were first trained on embeddings extracted from the 100-hour clean version of the LibriSpeech dataset and validated on the clean

development set of LibriSpeech [162]. The 100-hour clean version of LibriSpeech contains a total of 28,539 utterances from 585 speakers (284 males and 301 females). The pre-training was performed to ensure that the parameters of our autoencoders are properly tuned for encoding and decoding speaker embeddings for general *neutral* speaker population and to provide the same starting point for all speaker-specific autoencoders. We use the VESUS dataset to test emotional similarity because it provides us with the means to control for both emotion and lexical content without compromising the total number of samples available for each speaker [83].

The autoencoders that we use consist of five hierarchical down-sampling stages and five hierarchical up-sampling stages. The hierarchical architecture of our autoencoders is similar to models used in [42]. Each down-sampling layer in our autoencoders reduces the dimensionality of its input by two while each up-sampling layer increases the dimensionality of its inputs by two. This reduces the effective size of speaker embeddings to 16 features from their original 512 features before being up-sampled. Each block (but the last) in our autoencoders consist of a fully-connected layer followed by a Tanh activation. The last block only includes a fully-connected layer with no activation units. We use the mean squared error (MSE) loss function and train our autoencoders using the ADAM optimizer with a learning rate of 0.001 and batch sizes of 256. We run the training for a total of 100 epochs and apply early stopping once the validation loss does not improve for five consecutive epochs. For fine-tuning, we use a batch size of 32 and use the same learning rate and loss functions used for training the autoencoders. We run the fine-tuning for a total of 50 epochs and apply early stopping once the loss on a held-out validation set does not improve for five consecutive epochs.

6.5.1.1 Question 1 Experimental Setup

We first pre-train the autoencoders using neutral speech from the LibriSpeech corpus. Then, for each speaker in our test corpus (VESUS), we partition their data into three categories: (1) neutral training, (2) neutral testing, and (3) emotional testing. We use the neutral training data (which consist of 70% of a speaker's total neutral data) to fine-tune the autoencoder for each speaker. This allows us to construct a *baseline* model for each speaker. We then create a distribution using the reconstruction error associated with the neutral testing data and compare the reconstruction errors obtained from the emotional testing data to this distribution. If, in general, the reconstruction error on the neutral speech is lower than that of the emotional speech, this will support the hypothesis that embeddings are modulated by emotion.

We analyze the effect emotion has on the reconstruction errors using linear mixed effect models (LMEMs), implemented via the *lme*4 package [163] in R [164]. We set the reconstruction error as a response variable in our linear models, and set the emotion (*neutral* vs. *non-neutral*) and the gender as dependent binary variables. We set random intercepts for speaker_ids and utterance_duration (discretized into 3-quantiles), as well as per-speaker random slopes. In case the linear model fails to converge, we simplify the model by removing the per-speaker random slope and only retain the random intercepts, as suggested in [165]. We use likelihood ratio tests to test for statistical significance and test a full model (with the emotion fixed effect) against a null model (without the emotion fixed effect).

6.5.1.2 Question 2 Experimental Setup

We first pre-train the autoencoders using neutral speech from the LibriSpeech corpus. Then, for each speaker in our test corpus (VESUS), we partition their data into four categories: (1) neutral training, (2) neutral testing-a, (3) neutral testing-b, and (4) emotional testing. Further, we filter utterances in partition (4) such that we

only retain those that can be matched based on lexical content with utterances in partition (2). Note that due to the lexically controlled nature of VESUS, the lexical content of each utterance is unique. As a result, each neutral partition contains utterances with unique content. We use the data in the neutral training partition to fine-tune the autoencoder for each speaker. This allows us to construct a *baseline* model for each speaker. We then create a distribution using the reconstruction error associated with the neutral testing-a data, and compare the reconstruction error of the neutral testing-b and emotional testing data to this distribution. If the difference in reconstruction errors between partition (2) and partition (4) is bigger than the error between and partition (2) and partition (3), then this will support the hypothesis that modulation on the speaker embeddings due to variations in emotion are larger than those due to variations in lexical content.

We run a series of LMEMs to analyze the effect of emotion variation and lexical content variation on reconstruction errors. We set the reconstruction error as a response variable in our linear models, and set a binary value (i.e., *neutral* vs. *non-neutral with same content* or *neutral* vs. *neutral with different content*) and the gender as dependent binary variables. We set random intercepts for speaker_ids and utterance_duration (discretized into 3-quantiles), as well as per-speaker random slopes. We follow the same process described in Section 6.5.1.1 to fit the LMEMs and test for significance.

6.5.2 Experiment 2: Speaker Embeddings as General Paralinguistic Features

While the previous experiment investigates whether or not speaker embeddings capture speech characteristics that are changed by variations in emotion, this experiment investigates whether or not these disturbances can be used for recognizing emotions. We compare the emotion recognition performance obtained with speaker

embeddings to the performance obtained with state-of-the-art features used in the literature. We hypothesize that emotion recognizers that use speaker embeddings as features will outperform those that use common features from the paralinguistics literature. Our hypothesis is based on the fact that speaker embeddings are extracted from models that were trained on much bigger and diverse datasets compared to the commonly used emotion features. Specifically, the large and in-the-wild nature of the datasets used for training the speaker recognition models encourages the models to extract robust representations that capture speaker characteristics from a given audio signal, regardless of the acoustic conditions or environmental noise present. This experiment allows us to understand the relationship between the speaker and emotion recognition tasks, and helps us assess the prospects of replacing low-level paralinguistic features with speaker embeddings in emotion recognition models.

We compare the performance obtained using the extracted speaker embeddings to baselines obtained from common paralinguistics feature sets. The first category are the same 30-dimensional MFCCs used by the speaker identification model to extract the speaker embeddings. This allows us to ask how the transformation introduced by the speaker embeddings improves our ability to recognize emotion. The second category include feature sets broadly grouped based on their use of statistics to characterize the patterns in low-level acoustic features. These feature sets include: the INTERSPEECH 2009 (IS09) Emotion Challenge features [39] (384 parameters), the INTERSPEECH 2013 Computational Paralinguistics ChallengE (ComParE) [40] features (6,373 parameters), the Geneva Minimalistic Acoustic Parameter Set (GeMAPS) [41] (62 parameters), and the extended Geneva Minimalistic Acoustic Parameter Set (eGeMAPS) [41] (88 parameters). The features for all categories were z-normalized using the training set statistics while the utterances for the speaker embeddings were mean-normalized using a three-second window applied to the MFCC features of each utterance.

We assess the utility of the features in emotion recognition by running both within-corpus and cross-corpus recognition experiments with the IEMOCAP and MSP-IMPROV datasets. For the within-corpus experiments, we follow a leave-one-speaker-out evaluation scheme. For the cross-corpus experiments, we train our models using the labeled samples from one dataset and evaluate on the other dataset (and vice versa). The cross-corpus setup limits the effect of spurious correlations that a trained model can use in the evaluation process. We use unweighted average recall (UAR), which takes an average of the recall of each emotion class, as our evaluation metric in this experiment. This metric allows us to account for the class imbalance in the two datasets we use in this experiment.

The emotion recognition model that we use is based on deep neural networks (DNNs) as previous research has demonstrated their effectiveness when used with state-of-the-art feature sets [67, 166, 167]. For the within-corpus experiments, we follow a leave-one-speaker-out evaluation scheme, where for each test speaker, we use the opposite gender speaker from the test speaker's session as our validation speaker. For the cross-corpus experiments, we use the two speakers from the last session (i.e., session five for IEMOCAP and session six for MSP-IMPROV) as our validation speakers. The hyper-parameters for our DNNs include the number of hidden layers $\{1, 2, 3\}$ and the width of each hidden layer $\{128, 256, 512\}$. We use ReLU activation units in all of our experiments. The networks were trained using the ADAM[2] optimizer with a learning rate 10^{-4} on batches with 32 samples. We assign weights to our training samples according to the inverse of their respective frequencies in the training sets and train the models using a weighted cross-entropy loss function for a total of 100 epochs. We use the held-out validation set for hyper-parameter selection and early stopping. We apply the model that yields the highest validation performance to the unseen test data and report the test performance. Finally, we run

[2]Default parameters were used ($\alpha = 0.0001, \beta_1 = 0.9, \beta_2 = 0.999$)

each setup 30 times to account for variance from random initialization and training.

6.6 Results

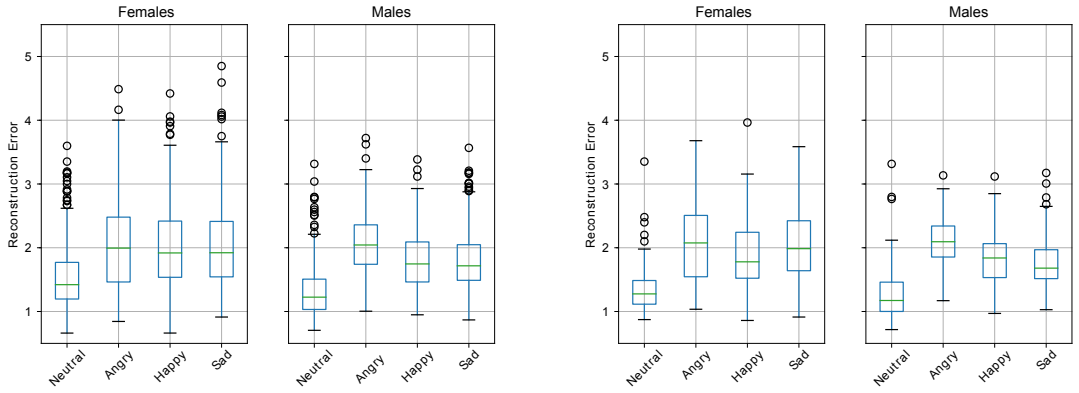
6.6.1 Experiment 1: Speaker Embeddings and Emotions

In this experiment, we study how emotion modulates speaker embeddings, measured in terms of reconstruction error. Smaller reconstruction errors indicate that the samples are more similar to the baseline distribution of *neutral* utterances while bigger reconstruction errors indicate that the samples are different from the baseline distribution. We will treat evidence of emotion-centric modulation, measured by reconstruction error, as evidence of the utility of embeddings for emotion recognition.

6.6.1.1 Question 1 Results

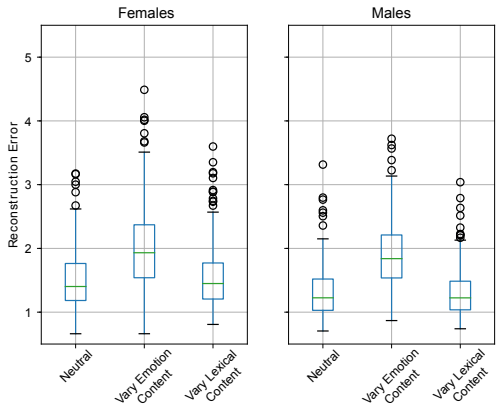
Our first question asked whether or not variations in emotion significantly modulate speaker embeddings from their neutral representation. We find that variations in emotion significantly modulate speaker embeddings from their neutral representation. In addition, we find that these modulations are consistent across male and female speakers. Figure 6.1(a) shows the reconstruction errors associated with 3,032 utterances, grouped by *intended* emotions (758 neutral, 758 *happy*, 758 *angry*, 758 *sad*). Figure 6.1(b) shows the reconstruction errors associated with 752 utterances, grouped by *perceived* emotions (188 neutral, 188 *happy*, 188 *angry*, 188 *sad*). The perceived emotions group includes utterances whose labels achieved at least 50% agreement between the intended and perceived emotions.

Figure 6.1(a). We find that the reconstruction error was significantly increased by 0.667±0.103 when moving from neutral speech to angry speech ($\chi^2(1)$=16.475, p=4.929e-05). Similarly, we find that the reconstruction error was significantly increased by 0.458±0.070 when moving from neutral speech to happy speech ($\chi^2(1)$=16.760,

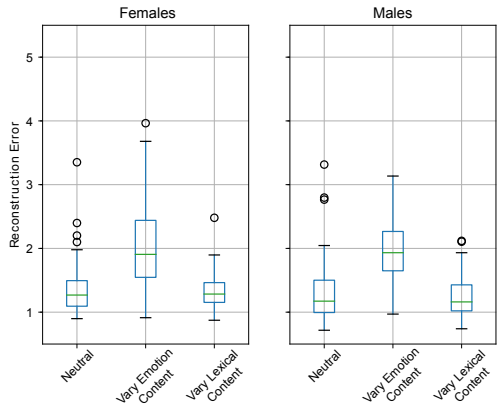


(a) Intended Emotion

(b) Perceived Emotion



(c) Intended Emotion



(d) Perceived Emotion

Figure 6.1: Reconstruction errors obtained from autoencoders trained with embeddings extracted from *neutral* utterances. Sub-figures (a) and (b) show the reconstruction errors grouped by emotion (*neutral*, *angry*, *happy*, *sad*) and gender (females, males). Sub-figures (c) and (d) compare the reconstruction errors obtained from *neutral* utterances to those obtained from *emotional* utterances with lexical content fixed, and to those obtained from *neutral* utterances but with different lexical content.

p=4.242e-05). Finally, we find that the reconstruction error was significantly increased by 0.490±0.071 when moving from neutral speech to sad speech ($\chi^2(1)$=17.560, p=2.784e-05). If we use the reconstruction error as a feature and apply a threshold to separate neutral and non-neutral speech, then we obtain an Area Under the Receiver Operating Characteristic curve (AUC) of 0.782. This indicates that the reconstruction errors obtained from an autoencoder that was exclusively trained on neutral speech can be used for detecting non-neutral speech.

Figure 6.1(b). We find that the reconstruction error was significantly increased by 0.645±0.126 when moving from neutral speech to angry speech ($\chi^2(1)$=11.683, p=6.307e-4). Similarly, we find that the reconstruction error was significantly increased by 0.458±0.070 when moving from neutral speech to happy speech ($\chi^2(1)$=11.871, p=5.703e-4). Finally, we find that the reconstruction error was significantly increased by 0.548±0.080 when moving from neutral speech to sad speech ($\chi^2(1)$=14.606, p=1.325e-4). If we use the reconstruction error as a feature and apply a threshold to separate neutral and non-neutral speech, then we obtain an AUC of 0.850. Again, demonstrating the utility of this setup for speech novelty detection applications.

6.6.1.2 Question 2 Results

Our second question asked whether the modulations due to emotion are larger or smaller compared to to those due to lexical variation. We find that variations in the lexical content have a non-significant effect on neutral embeddings, compared to the significant effect observed in the emotional utterances. We compare the reconstruction errors in three partitions of data as described in Section 6.5.1: (1) control neutral; (2) non-neutral with fixed lexical content; and (3) neutral with varying lexical content.

Figures 6.1(c) and 6.1(d) show the reconstruction errors, grouped by the aforementioned three partitions, associated with 1,892 and 458 utterances, respectively. Figure 6.1(c) displays the results obtained with intended emotion labels while Fig-

ure 6.1(d) displays the results obtained with perceived emotion labels. As before, the perceived emotions group includes utterances whose labels achieved at least 50% agreement between the intended and perceived emotions.

Figures 6.1(c). We find that the reconstruction error was significantly increased by 0.529±0.0253 when moving from neutral speech to non-neutral speech while keeping content fixed ($\chi^2(1)$=384.450, p¡2.2e-16). If we use the reconstruction error as a feature and apply a threshold to separate neutral and non-neutral speech, we obtain an AUC of 0.785. In contrast, we find that the reconstruction error does not significantly change when varying lexical content while keeping emotion fixed.

Figures 6.1(d). We find that the reconstruction error was significantly increased by 0.649±0.091 when moving from neutral speech to non-neutral speech while keeping content fixed ($\chi^2(1)$=14.398, p=1.480e-4). If we use the reconstruction error as a feature and apply a threshold to separate neutral and non-neutral speech, then we obtain an AUC of 0.840. We again find that the reconstruction error does not significantly change when varying lexical content but fixing emotion to neutral.

6.6.1.3 Experiment 1 Discussion

The findings from this experiment suggest that while neutral speaker embeddings may be invariant to modulations due to variations in lexical content, they are significantly changed by variations in emotions. We find that using utterances with majority emotion agreement yields smaller overlaps between the interquartile ranges (IQRs) of reconstruction errors obtained from the neutral and emotional utterances, for both female and male speakers, compared to those obtained when using all utterances (i.e., intended emotions). One explanation for this is that the intended emotion labels are more subtle than the perceived emotion labels that we use in this chapter. As a result, we see more pronounced modulations with the perceived labels compared to the modulations we see when using the intended labels. Finally, we note that none

of the models we ran yielded significant interaction between emotion and gender, suggesting that the increases in reconstruction error per emotion (for both intended and perceived) are consistent across female and male speakers.

The findings suggest that speaker embeddings can be used for establishing a baseline of how an individual sounds in their neutral state (i.e., normal behavior). Then, disturbances to this speaker model can be used as a proxy for measuring deviations from this normal behavior. This property of speaker embeddings can be beneficial in applications where we have ample baseline data from a speaker in the neutral state, but have limited or no access to outlier or novel data points from the speaker in certain states (e.g., road rage detection applications in vehicles). In the next experiment, we test if we can utilize these observed modulations in speaker embeddings for detecting emotions in challenging settings.

6.6.2 Experiment 2: Speaker Embeddings as General Paralinguistic Features

In the first experiment, we studied the relationship between emotion and speaker embeddings. In this section, we compare speaker embeddings to state-of-the-art paralinguistic features on the task of emotion recognition. We first demonstrate the relative ability of embeddings, compared to conventional speech emotion features, in a within-corpus experiment. We then repeat the analysis cross-corpus. In both cases, we assess the efficacy of the feature sets on the IEMOCAP and MSP-IMPROV datasets.

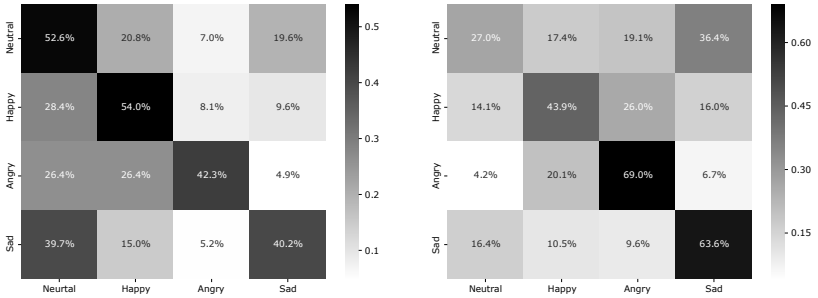
We first compare the emotion recognition performance of different feature sets within-domain. Overall, we find that speaker embeddings, when used as paralinguistic features, significantly outperform or perform comparably to the baseline features described in Section 6.5.1. We find that speaker embeddings significantly outperform MFCCs, IS09, and GeMAPs; and perform comparably to eGeMAPs and ComParE for

Table 6.2: The unweighted average recall (UAR) obtained for each setup in the within-corpus and cross-corpus experiments. MSP and IEM denote the MSP-IMPROV and IEMOCAP dataset, respectively. Models in the within-corpus experiments are evaluated following a leave-one-speaker-out evaluation scheme. MSP under cross-corpus indicates the performance of a model that is trained on IEMOCAP and evaluated on MSP-IMPROV; IEM under cross-corpus indicates the performance of a model that is trained on MSP-IMPROV and evaluated on IEMOCAP. The results shown are averages (± 1 standard deviation) from 30 runs with different random seeds. The best result in each experiment is **bolded**. ‡ indicates that the marked performance is significantly higher than all baselines; $*$ indicates that the marked performance is significantly higher than MFCCs; † indicates that the marked performance is significantly higher than all but eGeMAPS and ComParE. Significance is assessed at $p < 0.05$ using the Tukey's honest test on the ANOVA statistics.

Features	**Within-corpus UAR (%)**		**Cross-corpus UAR (%)**	
	MSP	**IEM**	**MSP**	**IEM**
Chance	25.0	25.0	25.0	25.0
MFCCs	40.7 ($\pm$1.8)	51.6 ($\pm$1.6)	39.2 ($\pm$2.7)	43.7 ($\pm$3.1)
IS09	45.6 ($\pm$2.3)	55.9 ($\pm$1.6)	42.1 ($\pm$0.9)	43.7 ($\pm$2.7)
ComParE	47.1 ($\pm$3.4)	56.0 ($\pm$1.9)	42.0 ($\pm$1.1)	48.6 ($\pm$3.0)
GeMAPS	42.5 ($\pm$3.6)	56.2 ($\pm$1.9)	42.2 ($\pm$1.1)	38.7 ($\pm$2.2)
eGeMAPS	45.7 ($\pm$3.0)	56.6 ($\pm$1.9)	39.9 ($\pm$1.3)	35.9 ($\pm$3.1)
Embeddings	**47.7 ($\pm$1.8)***†	**57.3 ($\pm$3.1)***	**47.3 ($\pm$2.1)**‡	**50.9 ($\pm$2.1)**‡

the within-corpus experiments on the MSP-IMPROV dataset. We find that speaker embeddings only significantly outperform MFCCs and perform comparably to other baseline features for the within-corpus experiments on the IEMOCAP dataset (Table 6.2).

Next, we analyze the performance of these feature sets in a more challenging cross-domain task. We find that speaker embeddings significantly outperform all other features when evaluating the models on the IEMOCAP and MSP-IMPROV datasets (Table 6.2). In addition, we observe a higher test performance when we test on the IEMOCAP dataset than we do when we test on the MSP-IMPROV dataset. Among the baselines, we find that the ComParE feature set outperforms all other baselines on the IEMOCAP dataset but performs comparably to IS09 and GeMAPS on the MSP-IMPROV dataset. The results suggest that the embeddings are more



(a) MSP-IMPROV (b) IEMOCAP

Figure 6.2: Confusion matrices obtained using speaker embeddings in the cross-corpus setting when (a) training on IEMOCAP and testing on MSP-IMPROV; (b) training on MSP-IMPROV and testing on IEMOCAP.

robust to domain-shifts than baseline features.

Figure 6.2 shows the confusion matrices obtained when using speaker embedding in cross-corpus emotion recognition settings. When testing on the MSP-IMPROV corpus, we find that the performance of detecting the neutral and happy emotions is higher than the performance of detecting the angry and sad emotions. In contrast, when testing on the IEMOCAP corpus, we find that the performance of detecting the Angry and Sad emotions is higher than the performance of detecting the neutral and happy emotions. The trends displayed by the confusion matrix in Figure 6.2(b) agree with the trends we saw in Figures 6.1(a) and 6.1(b). Specifically, the confusion matrix in Figure 6.2(b) shows that we obtain the highest performance when detecting the angry emotion, followed by both the sad and happy emotions. However, the confusion matrix in Figure 6.2(a) shows that the happy emotion is the easiest to detect, followed by the angry and sad emotion. Finally, we find that the improvements gained by using speaker embeddings over ComParE features cannot be attributed to the improvement in recognizing a specific emotion, but instead, can be attributed to a consistent improvement across all emotions.

6.6.2.1 Experiment 2 Discussion

To the best of our knowledge, this is the first study to compare both within-corpus and cross-corpus emotion recognition performance obtained using speaker embeddings to the performance obtained using general features commonly used in the emotion recognition community. Our results suggest that speaker embeddings are highly versatile, and can easily be adapted to other paralinguistic applications such as emotion recognition. We note that speaker embeddings also provide a more compact alternative to some of the features sets (e.g., ComParE). For example, the ComParE feature set contains 6,373 parameters representing energy, spectral, and voicing features [149]. In contrast, speaker embeddings only contain 512 parameters and are extracted from 30-dimensional MFCCs (i.e., spectral).

6.7 Discussion and Conclusion

In this chapter, we proposed the use of speaker embeddings, representations extracted from neural networks trained on a speaker identification task, as paralinguistic features to be used in emotion recognition applications. Speaker embeddings capture high-level speaker characteristics and abstract extraneous low-level variations in the acoustic signal that are not needed for recognizing speakers. The hypothesis that drove our study is that emotionally charged vocal expressions make speakers sound different from how they typically sound.

We first used autoencoders to quantify the effect of emotion on speaker embeddings. We trained our auto-encoders on *neutral* utterances from each speaker, and used the reconstruction errors obtained for test utterances as a proxy for measuring the effect emotion has on speaker embeddings. Our analysis showed that embeddings extracted from expressive speech resulted in significantly increased reconstruction error compared to neutral speech. In addition, our analysis showed that lexical

variation had a non-significant effect on the reconstruction errors obtained from the utterances. Our experiments also demonstrated how the reconstruction errors obtained from the autoencoders can be used as features for detecting deviations from the neutral state. Future work will study techniques for making changes in emotions more pronounced while maintaining speaker discriminative properties in speaker embeddings (e.g., emotion-invariant x-vectors).

We then showed that speaker embeddings can be used as a replacement to common paralinguistic features used in emotion recognition tasks. We demonstrated this by showing not only that speaker embeddings outperform baseline features in cross-corpus emotion recognition tasks, but also that they are more compact (i.e., fewer parameters) than state-of-the-art paralinguistic features. Speaker embeddings outperformed other features despite being extracted from spectral representations (i.e., MFCCs) alone. In contrast, other features used a combination of energy, voicing, and spectral representations. MFCCs used for extracting speaker embeddings were originally designed based on observations from perceptual experiments and thus, may not be optimal for all speech applications. For example, MFCC features smooth the speech spectrum and make it difficult to extract other narrow-band information that is known to be predictive of emotion (e.g., pitch, formants). One extension to the current approach is to train the speaker identification models with representations from which this fine-grained information is easily extractable (e.g., spectrograms, raw waveform). Another extension to the current approach is to combine speaker embeddings with common emotion features to provide the recognizer access to the fine-grained information present in the speech signal.

In conclusion, this chapter further contributed to our understanding of the relationship between emotions and speaker representations, and demonstrated how variations in emotion manifest themselves in speaker embeddings. These manifestations not only can impact the performance of a verification system, but also can be lever-

aged for detecting emotions.

CHAPTER VII

Distilling Emotional Expression in Speech Through Voice Conversion

7.1 Introduction

One major challenge with building robust paralinguistic models is the limited access to large-scale datasets (i.e., several hundreds of hours) with accurate paralinguistic labels. In addition, the highly subjective nature of many paralinguistic tasks, such as emotion expression and perception, exacerbates the data sparsity challenge by making the data collection and annotation process both costly and time consuming. Consequently, datasets used for building paralinguistic models, specifically emotion models, are significantly smaller than those used for developing other speech applications. For instance, a typical emotion dataset (e.g., IEMOCAP) that is used for building paralinguistic models contains around 12 hours of speech while a modern dataset used for building speaker recognition models (e.g., VoxCeleb) contains around 2000 hours of speech. New solutions are needed to address the effects of the data size discrepancy for paralinguistic tasks.

We introduce the Expressive Voice Conversion Autoencoder (EVoCA), an unsupervised framework for learning features that distills paralinguistic attributes from speech without relying on explicit emotion or style labels. EVoCA learns what it

means for speech to be expressive by treating *expressive* speech as a modulation of *neutral* speech. The goal is to then train a style encoder that learns a style embedding, a compact representation of the expressivity of an utterance, that can be used to transform speech from neutral to expressive. EVoCA achieves this goal using parallel speech inputs: one expressive and one neutral. However, these types of parallel paralinguistic corpora are not available at scale. Instead, we use a large audiobook corpus (i.e., 200 hours) composed of expressive speech and artificially generate the parallel neutral speech using the available transcriptions (see Figure 7.1). We train the EVoCA model to convert between the synthetic neutral speech signal and the real expressive speech, and demonstrate how this conversion yields a style embedding that captures paralinguistic attributes (see Figure 7.2). We then show that the learned style embeddings can be used in downstream emotion recognition and speaking style classification tasks. The benefit is that a pre-trained style encoder could be independently used in future applications to generate embeddings that highlight paralinguistic content.

In summary, the contributions of this work are as follows:

- We present the EVoCA framework for learning speech emotion and style features from audiobooks without relying on manual annotations for those attributes.
- We demonstrate the that the transformed features are more effective than surface acoustic features for recognizing emotions and speaking style.
- We show that EVoCA learns embeddings that outperform those obtained using other unsupervised and self-supervised speech feature learning methods from the literature.

To the best of our knowledge, ours is the first work to demonstrate how one can learn paralinguistic features by training a neural model to convert between non-expressive synthetic speech and expressive real speech.

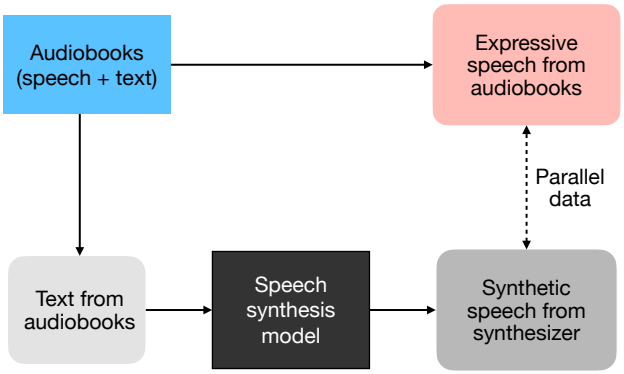


Figure 7.1: An overview of the parallel data generation process. We use a speech synthesis model to generate a synthetic version of each audio sample in the original audiobook corpus. Synthesized samples lose paralinguistic attributes present in the original samples but retain linguistic information. Our goal is to leverage the resulting real/synthetic sample pairs to learn to extract paralinguistic features.

7.2 Related Work

Speech emotion recognition applications rely on an extensive set of acoustic features that has evolved over the years [39, 168, 77, 40, 169]. Spectral features are a crucial component of any emotion feature set, and are included in the widely used ComParE and eGeMAPs feature sets [40, 169]. Common surface features that are derived from the speech spectrum include Mel-frequency cepstral coefficients (MFCCs) and Mel-filterbanks (MFBs). In this work, we propose a framework for learning an MFB transformation that highlights the paralinguistic content of an utterance, and we demonstrate the effectiveness of the learned transformation over using surface MFB features on emotion and speaking style classification tasks.

Our work also explores the utility of using both synthetic and real speech to learn paralinguistic information. Lotfian and Busso have previously demonstrated how speech synthesizers can be used to remove emotion from speech, and provide trained emotion recognizers with a neutral reference to aid in the recognition of expressive speech [170]. They found that providing emotion recognizers with both real and synthesized speech led to improved emotion recognition performance. One limitation

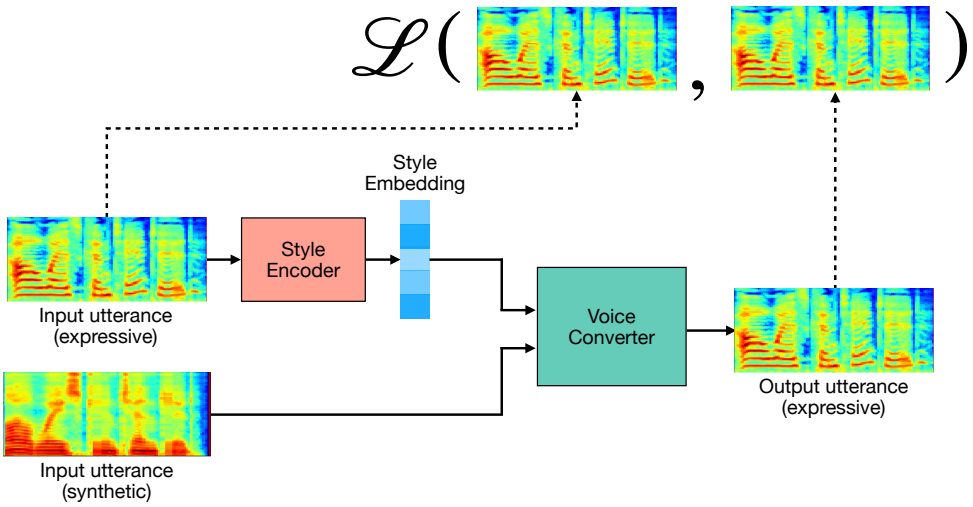


Figure 7.2: An overview of the proposed Expressive Voice Conversion Autoencoder (EVoCA). The model takes the expressive and synthetic speech samples as inputs; and outputs the reconstructed expressive speech sample. The Style Encoder extracts an embedding from the expressive speech sample such that it can be used by the Voice Converter to insert paralinguistics into the synthetic speech input sample. The network is trained with an $L2$ loss between the generated expressive sample and the original expressive sample. Once the full model is trained, the Style Encoder is disconnected and used as a general purpose paralinguistic feature extractor.

with their approach is that it relies on having access to a real-time speech synthesizer to generate a neutral version of the input utterance for use by the emotion recognizer. In contrast, we use the speech synthesizer only during the data preparation process (Figure 7.1) and not during test time; which makes our approach more efficient to run during test time.

Our approach is also related to works that focused on unsupervised and self-supervised speech representation learning. Chung et al. introduced two auto-regressive methods to learn MFB transformations for speech applications without relying on explicit labels [171]. Both of the proposed models were trained to predict future frames of the input speech sequence in order to learn global structures represented in the speech signal. They showed that the resulting transformation improved performance over surface features on speaker verification and phone recognition tasks. Hsu et al. devised a variational autoencoder that is capable of learning hierarchical informa-

tion present in speech data [172]. Their approach disentangled frame-level features from utterance-level features in order to provide robust embeddings for both speaker recognition and automatic speech recognition tasks. Although many unsupervised learning strategies exist for learning speech transformations, ours is the only approach that is targeted at learning transformations that highlight expressive characteristics in speech.

Recent works in voice conversion have also inspired our proposed approach. The goal of voice conversion is to convert an utterance from one speaker so that it sounds as if it was spoken by another speaker [173]. In other words, a voice converter retains all linguistic content and only modulates the paralinguistics of speech. Previous works demonstrated that voice conversion techniques can be used to convert between emotional states [174, 175, 176]. However, to the best of our knowledge, our work is the first to show that the voice conversion task can be adapted and incorporated into a framework that enables a neural network to learn compact embeddings that capture emotional expression in speech.

7.3 Approach

7.3.1 Creating Parallel Data using Speech Synthesis

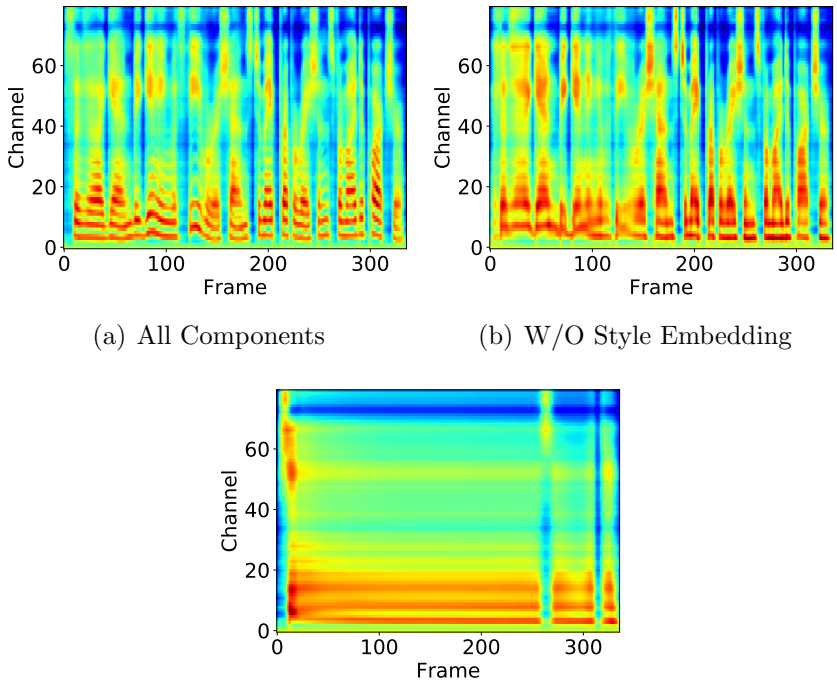
A sketch of our data generation setup is shown in Figure 7.1. Given an audiobook corpus, where both speech and text modalities are available, we use the text to create synthetic speech samples using a speech synthesizer. The created synthetic speech should lack expressiveness. This provides our system with the opportunity to learn how to characterize expressiveness and imbue the non-expressive speech with expressive characteristics. We use the open-source Festival toolkit[1], as previous research has demonstrated its utility to generate neutral non-emotionally expressive speech [53].

[1] `http://festvox.org/festival/`

Once the speech synthesis process finishes, our data now contain pairs of real (expressive) speech and synthetic (neutral) speech. Our EVoCA model then leverages the resulting parallel data to learn an embedding transformation that facilitates the conversion from synthetic to real speech without relying on any manual emotion or style labels.

7.3.2 Expressive Voice Conversion Autoencoder Setup

A sketch of EVoCA is shown in Figure 7.2. The key idea behind our proposed framework is that expressive speech is a modulation to neutral speech. Thus, a model that converts between neutral and expressive speech learns a quantification, in the form of an embedding, which characterizes expressiveness. The proposed EVoCA model consists of two components: a style encoder and a voice converter. The style encoder condenses the paralinguistic attributes of the original expressive speech into a fixed-size feature vector, which we refer to as the style embedding. The style embedding and the paired synthetic speech sample are fed into the voice converter, which produces expressive speech. A reconstruction loss ($L2$) between the generated expressive speech and the original expressive speech is computed and used to train the style autoencoder in an end-to-end fashion. The style embedding can be used to do more than transform speech from non-expressive to expressive, it can also be viewed as a numeric quantification of the expressive characteristics within a given speech sample. Therefore, once trained, the style encoder can be used to transform the original surface features into representations that highlight the emotional components of the input data.



(a) All Components

(b) W/O Style Embedding

(c) W/O Synthetic Reference

Figure 7.3: Sample converted test utterance with three model setups.

7.4 Datasets, Features, and Metrics

7.4.1 Datasets

We use four datasets in this chapter: Blizzard2013, IEMOCAP, MSP-IMPROV, and VESUS. Blizzard2013 is used to train the EVoCA model while the other three datasets are used to test the effectiveness of the learned embeddings on the speech emotion recognition and speaking style detection tasks. The Blizzard2013 dataset contains around 200 hours from 55 American English audiobooks read by Catherine Byers. Although other audiobook-based datasets are publicly available, we choose the Blizzard2013 corpus due to its highly expressive and animated nature. This corpus was used in previous research to model style and prosody in speech synthesis applications [177, 178].. We use a segmented version of the corpus which we obtained from the 2013 Blizzard Challenge website.[2]

[2] `http://www.cstr.ed.ac.uk/projects/blizzard/`

A description of the IEMOCAP, MSP-IMPROV, and VESUS datasets can be found in Chapter II. For IEMOCAP and MSP-IMPROV, we only consider utterances that had majority agreement among the annotators and focus on four basic categorical emotions: happy (merged with excited for IEMOCAP), angry, neutral, and sad. In addition to emotion labels, the IEMOCAP dataset provides spontaneity labels, which we use in our speaking style detection experiments. For VESUS, we focus on utterances that achieved at least 50% agreement among the crowd-sourced annotators with respect to the actor's intended emotion.

7.4.2 Features

We first pre-process speech samples from all datasets such that they have a sampling rate of 16 kHz and then extract 80-dimensional MFB features using the `Librosa` toolkit [179] with a 50 ms Hanning window and a step size of 12.5 ms, consistent with previous research in voice conversion [180]. We z-normalize the frequency bins per utterance for the voice converter and mean-normalize the bins per-utterance for the style encoder; consistent with normalization methods used in previous works [146, 178]. Normalization ensures that the features are robust to variations that could arise from having different recording conditions [181].

7.4.3 Tasks

The voice conversion task is a regression task in which the goal is to output the MFB features of an expressive speech utterance given the MFB features of the synthesized speech utterance. The emotion recognition task is a multi-class classification task in which the goal is to recognize the target emotion from the set available in the dataset. Speaking style detection is a binary classification task in which the goal is to recognize if the target data is acted or read.

7.4.4 Metrics

We use Mel-cepstral distortion (MCD) and root mean square error (RMSE) of F0 for evaluating the quality of the converted speech [180] when training the end-to-end model. MCD and F0 RMSE cannot be directly extracted from the MFB acoustic features used by our conversion model. Thus, we use `Librosa` to invert the MFB features to audio by first approximating the Short-time Fourier transform (STFT) magnitude and then using the Griffin-Lim algorithm to reconstruct the phase. We extract the F0 and 24-dimensional mel cepstral coefficients from the waveform using the WORLD vocoder [182] following [180, 178].

We use unweighted average recall (UAR) and accuracy for evaluating the performance on the emotion recognition and speaking style detection tasks. The UAR metric is used to account for the class imbalance that is inherent in the emotion data [115]. Chance UAR is 25% and 50% for the emotion recognition and speaking style detection tasks, respectively.

7.5 Experiments

7.5.1 Experimental Questions

We design our experiments to address the following four questions regarding the proposed framework shown in Figure 7.2:

1. Is the proposed framework capable of inserting expressiveness into synthetic speech?

2. Can the learned style embeddings be used for emotion and style classification?

3. How do changes to the structure of the proposed framework affect both the quality of the converted speech and the effectiveness of the extracted embeddings for emotion and speaking style detection tasks?

4. How does the performance of style embeddings in emotion and speaking style detection tasks compare to those of feature transformations learned using other unsupervised and supervised methods?

The first question studies if the style encoder provides an embedding that the voice converter uses for inserting paralinguistics into a neutral speech signal. The second question looks at whether or not these compact embeddings are more useful for emotion and style classification tasks than surface-level MFB features. The third question asks how the capacities (i.e., number of hidden units) of both the style encoder and the voice converter affect the performance of both the voice conversion task and the downstream classification tasks. The fourth question aims to compare the utility of the learned feature transformation to those learned via other unsupervised and supervised methods from the literature. Next, we provide more details regarding the experimental setup.

7.5.2 Expressive Voice Conversion Autoencoder (EVoCA)

The proposed EVoCA consists of two components: the voice converter and the style encoder. The voice converter consists of a stack of four Bidirectional Long Short-Term Memory (BLSTM) layers, each with a hidden size of 256, followed by a 1D convolution layer with 80 channels and a kernel size of one. The style encoder we use consists of a stack of two BLSTM layers, each with a hidden size of 256. The fixed-size embeddings from the style encoder are induced by taking the mean of the hidden representations from the last BLSTM layer and then passing the outputs through a linear layer, which reduces the size by half. The reasoning for this linear layer is to counteract the bidirectional property of BLSTM which outputs hidden representations that are twice the size of the hidden layer. Our voice converter is inspired by the one used in [183]. However, in this work we utilize a basic version of the model that does not include a two-layer fully connected PreNet, a five-layer 1D

convolution PostNet, nor an attention module. We opt to use a simple implementation for voice conversion since our problem does not follow the sequence-to-sequence learning paradigm as our input features are pre-aligned using dynamic time warping (DTW). Our final style autoencoder model has approximately 2.2 million parameters.

We investigate how changes to the structure of the proposed EVoCA affect not only the quality of the converted speech, but also the quality of the extracted embeddings. We study the impact that the style embedding and synthetic speech has on the voice converter by comparing the voice conversion performance when only one component is present. We also investigate the effect of reducing the capacity (i.e., the number of hidden units) of the style encoder and the voice converter on the converted speech as well as on the extracted embeddings for downstream classification tasks. Specifically, we keep the voice converter fixed and reduce the hidden size of the BLSTM style encoder gradually from 256 units to 32 units (reducing the number of parameters from 2.2 million to 1.5 million), noting performance changes on the two tasks. Then, we keep the style encoder fixed and reduce the hidden size of the BLSTM voice converter from 256 units to 32 units (reducing the number of parameters from 2.2 million to 0.7 million), again noting performance changes on the two tasks. Note that these hyper-parameters are not and should not be tuned based on the performance of the downstream task as the goal of this experiment is to analyze how these parameters affect the qualities of the transformed features and the converted speech.

We split the Blizzard2013 data into training, validation, and test partitions following a random 90%-5%-5% split rule. We train our style autoencoder on the training partition, and use the validation partition for loss monitoring and early stopping. Conversion performance is reported on the test partition of the data. We construct the network in PyTorch and train it from scratch with batches of size 128 using the ADAM optimizer for a total of 80 epochs. We use an initial learning rate of 10^{-4} and decrease it exponentially using a decay factor of 0.95 after each epoch starting

from epoch 30. We monitor the validation loss after each epoch and perform early stopping if the validation loss does not improve for 15 consecutive epochs.

7.5.3 Unsupervised and Supervised Baselines

The first unsupervised baseline that we consider is a convolutional autoencoder that is applied to fixed-length MFB segments of 128 frames. The autoencoder is similar to the one used in [184]. The encoder consists of three 2D convolution layers, of shape: $[32 \times 9 \times 9]$, $[64 \times 7 \times 7]$, and $[128 \times 5 \times 5]$, followed by a linear layer with 256 units. A $[2 \times 2]$ max pooling operation is applied after each layer to reduce the dimensionality of the input by two. The decoder consists of a linear layer with 256 units followed by four 2D convolution layers of shape: $[32 \times 9 \times 9]$, $[64 \times 7 \times 7]$, $[128 \times 5 \times 5]$, and $[1 \times 1 \times 1]$. A $[2 \times 2]$ nearest neighbor up-sampling operation is applied after each layer to get back the original size of the input. Both the encoder and the decoder use Leaky ReLU activation units and the autoencoder has approximately 3.9 million parameters.

The second unsupervised baseline that we consider is the Autoregressive Predictive Coding (APC) model that was introduced in [171]. Given an input of MFB features, the APC model is trained to predict the features n time-steps in the future. The APC model that we use is similar to the one used by Chung et al. and it consists of three LSTM layers, each with a width of 512. We run our experiments with three values for n: 5, 10, and 20. Once trained, the outputs from the last LSTM layer are averaged to obtain fixed-size features for downstream tasks. The APC model that we use has approximately 5.5 million parameters.

We train both the autoencoder and the APC baselines on the Blizzard2013 dataset. We use the same protocol we use for training EVoCA when training the autoencoder baseline. However, we train the APC baselines for 100 epochs following the authors' recommendation.

The supervised baseline is the x-vector embeddings used for speaker identification [146]. The x-vectors model is a supervised neural model that is trained to identify speakers in a large corpus given MFCC features as input. Previous research demonstrated that x-vectors encode various paralinguistic attributes relating to speaking rate, style, and emotion [157, 154].

It is not possible to train the x-vector model on the same data because the dataset only has one speaker. Thus, we use a pre-trained x-vector model that was trained on an augmented version of the combined VoxCeleb datasets.[3] Due to the large size (2000 hrs) and diversity (7000+ speakers) of VoxCeleb, we consider the performance we get from the x-vector embeddings as a strong benchmark for feature transformation.

7.5.4 Emotion and Speaking Style Recognition

We test the utility of the learned style encoder for transforming MFB features to highlight their paralinguistic attributes in emotion recognition and speaking style detection tasks. First, we assess if transforming MFB features provides any advantage over using surface MFB features on the two tasks. Then, we compare the learned feature transformation to those obtained using the unsupervised and supervised baselines.

We follow a leave-one-speaker-out evaluation scheme and report the average performance across all test speakers on all four downstream tasks. For each test speaker, we pick the model that gives the best performance on a held-out validation set. The hyper-parameters that we optimize on the validation set include the number of hidden layers $\{1, 2, 3\}$, the width of each hidden layer $\{64, 128, 256\}$, and the activation unit {Tanh, ReLU}. We construct the networks in PyTorch and train them with batches of size 32 using the ADAM optimizer with learning rate of 10^{-4} and a cross-entropy loss function. We train each model for a maximum of 100 epochs and apply early

[3] `https://kaldi-asr.org/models/m7`

stopping if the validation loss does not improve for five consecutive epochs. We repeat each experiment with 30 different random seeds and report the average and standard deviation to account for performance fluctuation due to random initialization and training.

7.6 Results

In this section, we provide the results of our four experiments (see Section 7.5.1 for an overview).

1. Is the proposed framework capable of inserting expressiveness into synthetic speech? Table 7.1 shows that we obtain an MCD of 24.01 and an F0 RMSE of 146.20 when computing the performance using the synthetic reference speech and ground-truth expressive speech. In comparison, we obtain an MCD of 10.71 and an F0 RMSE of 64.36 when computing the performance using the converted speech and the ground-truth expressive speech. This suggests that that proposed EVoCA framework converts the synthetic speech so that its closer to the expressive speech. We note that it is possible to obtain better conversion performance if we increase the capacity of the model and utilize a more sophisticated vocoder. However, as the results for question 3 will suggest, increasing the capacity of the voice converter might not necessarily yield better embeddings for downstream classification tasks.

2. Can the learned style embeddings be used for emotion and style classification? Table 7.2 shows that our style embeddings significantly outperform MFB surface features on both the emotion recognition and the speaking style detection tasks. This suggests that the style encoder learns a feature transformation that highlights paralinguistic attributes that are obfuscated in surface acoustic features.

3. How do changes to the structure of the proposed framework affect both the quality of the converted speech and the effectiveness of the extracted embeddings for emotion and speaking style detection tasks? Fig-

Table 7.1: Objective performance measures for the style voice conversion task with different setups. The base EVoCA consists of a 256-dimensional style encoder and a 256-dimensional voice converter. Reference numbers are computed using the synthetic speech and ground-truth expressive speech. All other numbers are computed using converted speech and ground-truth expressive speech.

Setup	**MCD (dB)**	F_0 **RMSE (Hz)**
Reference	24.01	146.20
Base EVoCA	10.71	64.36
w/o synth. ref.	+8.33	+106.23
w/o style enc.	+1.90	+79.50
w/ 128-dim style enc.	+0.31	+6.14
w/ 64-dim style enc.	+0.69	+19.41
w/ 32-dim style enc.	+0.97	+31.06
w/ 128-dim converter	+1.03	+15.60
w/ 64-dim converter	+1.77	+31.73
w/ 32-dim converter	+2.61	+61.82

ure 7.3 visually demonstrates the effect of each input on the quality of a converted utterance. Figure 7.3a shows that the converted speech has higher quality when the style embedding is provided as an input compared to Figure 7.3b. Specifically, the harmonic structure in Figure 7.3a is well defined and dynamic while that in Figure 7.3b is relatively static and not well separated. Figure 7.3c shows that the model is unable to generate speech solely from style embeddings. We hypothesize that this is due to the style embeddings' limited capacity to encode both linguistic and paralinguistic information present in the original signal to allow for accurate reconstruction. Additionally, we believe style embeddings struggle to model time-varying phenomena like rhythm and speech activity because they are computed using a global average over LSTM outputs.

Table 7.1 quantitatively shows the effect of each of these two inputs on the conversion performance. We find that the synthesized reference input is more important to the conversion task than the style embedding. In other words, reducing the capacity of the style encoder has a less detrimental effect on the quality of the converted

speech, as measured by the performance metrics, than reducing the capacity of the voice converter does. This can be due to the fact that the style embeddings do not have enough capacity to encode the linguistic attributes in speech that are necessary for obtaining good voice conversion performance.

Tables 7.1 and 7.2 show the results obtained on the voice conversion task and the downstream classification tasks, respectively. We find that while a high capacity voice converter improves the quality of the converted speech, it can also degrade the quality of the extracted embeddings as measured on the classification tasks. For instance, we find that reducing the capacity of the voice converter from 256 to 128 decreases the conversion performance on the voice conversion task but improves the classification performance on two out of the four downstream tasks. We believe this is because using a high-capacity voice converter can reduce EVoCA's reliance on the style encoder for providing paralinguistic information; which causes the style encoder to perform poorly when used to transform features for downstream applications.

4. How does the performance of style embeddings in emotion and speaking style detection tasks compare to those of feature transformations learned using other unsupervised and supervised methods? Table 7.2 shows that style embeddings encode information that is more suited to paralinguistic tasks than those extracted from other unsupervised methods, namely APC and a traditional autoencoder. The APC model provides improvements over surface features on all four downstream tasks when using the 20-step setup, and shows improvements over surface features on three downstream tasks when using the 10-step setup. In contrast, a standard autoencoder fails to provide any improvements over surface features on all tasks. We believe that the success of the extracted embeddings from EVoCA demonstrate the importance of targeted unsupervised tasks.

EVoCA was also able to close a large margin of the performance gap between the x-vector system, which represents a strong model that was trained on a bigger

dataset with more than $7,000$ speakers. In contrast, EVoCA utilizes a smaller dataset without any speaker labels, which highlights how a carefully devised unsupervised training task can offset the requirement for having access to a large number of labels in a related task.

7.7 Concluding Remarks

In this work we proposed EVoCA, a framework for learning a surface feature transformation that highlights paralinguistic content needed for detecting emotion and speaking style. We first showed that speech synthesizers can be used to strip away paralinguistic attributes from speech while retaining linguistic information. We demonstrated how a neural voice conversion model can be adapted to facilitate the extraction of paralinguistic features by converting synthetic neutral speech to real expressive speech. Finally, we showed that these extracted embeddings improve performance over surface features and can outperform other embeddings extracted from existing unsupervised and self-supervised methods on emotion recognition and speaking style detection tasks. Future work will consider how the choice of the synthesis model, the number of speakers in the training set, and the architecture used for the speaker encoder affect the quality of the extracted embeddings.

Table 7.2: Performance obtained using different features for emotion recognition and speaking style classification. The performance on the emotion recognition task is measured using the unweighted average recall (UAR) while the performance on the speaking style detection task is measured using accuracy (Acc.). IEM, MSP, and VES denote the IEMOCAP, MSP-IMPROV, and the VESUS datasets, respectively. Performance is evaluated using a leave-one-speaker-out scheme and the numbers reported are averages (± 1 standard deviation) from 30 runs to account for randomness in initialization and training. $*$ indicates that the marked performance is significantly higher than MFBs. † indicates that the marked performance is significantly higher than best APC model. Significance is assessed at $p < 0.05$ using the Tukey's honest test on the ANOVA statistics.

	Emotion (UAR)			**Style (Acc.)**
Features	**IEM**	**MSP**	**VES**	**IEM**
Baseline – Surface Features				
Chance	25.0	25.0	20.0	52.3
MFBs	53.0 ± 0.6	43.6 ± 1.2	36.0 ± 1.4	67.0 ± 0.7
Baseline – Unsupervised				
Autoencoder	50.6 ± 0.9	38.7 ± 1.0	33.6 ± 1.1	64.2 ± 0.6
APC (5-steps)	51.7 ± 0.8	42.2 ± 0.8	33.5 ± 1.2	68.3 ± 0.6
APC (10-steps)	53.9 ± 0.9	44.6 ± 0.9	35.5 ± 1.6	69.7 ± 0.6
APC (20-steps)	54.3 ± 0.9	44.1 ± 0.9	36.1 ± 1.5	69.7 ± 0.6
Style Embeddings (ours)				
Base EVoCA	$56.4 \pm 0.6^{*\dagger}$	$46.0 \pm 0.6^{*\dagger}$	$44.2 \pm 0.9^{*\dagger}$	$71.7 \pm 0.5^{*\dagger}$
w/ 128-dim style enc.	$55.4 \pm 0.8^{*\dagger}$	$45.3 \pm 0.9^{*}$	$42.6 \pm 1.4^{*\dagger}$	$69.6 \pm 0.5^{*}$
w/ 64-dim style enc.	53.0 ± 0.6	42.9 ± 0.8	$38.2 \pm 0.9^{*\dagger}$	67.2 ± 0.5
w/ 32-dim style enc.	51.7 ± 0.6	41.0 ± 0.4	36.0 ± 1.3	65.7 ± 0.5
w/ 128-dim converter	$57.1 \pm 0.5^{*\dagger}$	$46.3 \pm 0.9^{*\dagger}$	$43.5 \pm 1.3^{*\dagger}$	$70.4 \pm 0.5^{*\dagger}$
w/ 64-dim converter	$57.0 \pm 0.7^{*\dagger}$	$44.9 \pm 0.9^{*}$	$41.0 \pm 0.9^{*\dagger}$	$69.6 \pm 0.6^{*}$
w/ 32-dim converter	$54.9 \pm 0.6^{*}$	$44.6 \pm 0.7^{*}$	$38.1 \pm 1.0^{*\dagger}$	$68.8 \pm 0.5^{*}$
Supervised				
x-vectors	$57.9 \pm 1.0^{*\dagger}$	$47.7 \pm 1.8^{*\dagger}$	$44.0 \pm 1.5^{*\dagger}$	$72.0 \pm 1.1^{*\dagger}$

CHAPTER VIII

Concluding Remarks

This dissertation presented novel solutions for detecting and quantifying emotional expression from speech. Chapters III, IV, and V focused on the introduction and investigation of modeling techniques that improve SER performance by addressing speech data variability and emotion annotation variability; Chapters VI and VII addressed data sparsity by introducing methods that learn to extract embeddings that highlight emotion content in speech using data that are not annotated for emotion. This chapter highlights the key findings and contributions of these works, and provides potential future directions.

8.1 Summary of Contributions

In Chapter III, we explored one limitation of the current two-step feature extraction pipeline that is commonly employed by SER systems, and proposed a modeling approach to address this limitation. The first step in this pipeline is to extract a sequence of low-level features that represent the speech utterance; the second step is to take statistics (over time) to induce a fixed-size feature vector from the sequence. We showed that the second step of this feature extraction pipeline can obfuscate the short-time dynamical properties of the acoustic features, which provide critical cues into an individual's emotions. We proposed to address this limitation by forgoing

the second step in the feature extraction pipeline, and instead, utilizing classification models that work directly on the sequential low-level acoustic features (e.g., convolutional neural networks). Finally, we proposed the use of speed perturbation as an effective data augmentation technique for improving the robustness of these models that work directly on the sequential low-level acoustic features.

In Chapter IV, we explored various methods for fusing (i.e., combining) multimodal speech data (i.e., acoustics and lexical) for predicting emotions, specifically predicting valence, from speech. A multimodal recognition system that is provided with multiple input streams needs to fuse the features obtained from these multiple streams before making a prediction. We focused our study on intermediate fusion methods, which take place in the context of neural networks, to combine feature descriptors from the acoustic and lexical modalities. We first showed that multimodal SER systems that combine acoustic and lexical features are better than unimodal systems for the valence prediction task. Then, we demonstrated how the lexical modality encodes more information about valence than the acoustic modality does. Finally, we showed how fusion strategies that consider fine-grained interactions between the extracted multimodal features (i.e., bilinear pooling based methods) are more effective than traditional fusion methods.

In Chapter V, we proposed the use of two convolutional neural architectures to improve time-continuous SER performance. The proposed architectures address two challenges that are inherent in time-continuous SER problems. The first challenge is that the reaction delay of the annotators introduces a mismatch between the acoustic and the annotation signals. The second challenge is that the acoustic signal exhibits more abrupt variations in time compared to the slow moving annotation signal (i.e., the annotation signal is smooth and has considerable time dependencies). We showed how designing network architectures that explicitly account for these two effects improves the performance of SER over baselines methods. Specifically, we showed that

convolutional network architectures that cover large contexts allow SER models to compensate for the mismatch between the acoustic and the annotation signals. Further, we showed how convolutional network architectures that model a downsampled version of the input signal and then generate the output signal through an upsampling operation can generate outputs that are more in-line with the slow moving nature of human annotations.

In Chapter VI, we explored the use of speaker embeddings, embeddings extracted from speaker recognition models, as robust features for SER systems. Speaker embeddings provide a compact representation that captures the high-level characteristics of a speaker while suppressing extraneous low-level variations that are present in the speech signal. The large and diverse datasets available for the speaker recognition task make the task an attractive alternative to the emotion recognition task, where the data is small and limited, for learning robust embeddings from speech. To this end, we first demonstrated how speaker embeddings can be used for highlighting the differences that exist between neutral speech and emotionally expressive speech. We then showed how speaker embeddings can be used for establishing behavioral baselines for speakers, and we demonstrated how deviations from this baseline can be used for detecting emotions. Finally, we showed that speaker embeddings can replace general paralinguistic features for recognizing emotions as they are more robust to changes in domains and recording environments.

In Chapter VII, we introduced a framework that enables a neural network to learn embeddings that capture expressiveness in speech by using audio-textual data that are not annotated for emotion. The proposed framework allowed us to learn embeddings using datasets that are larger than those typically used in SER research since we are no longer limited by the sizes of the labeled emotion datasets available. We showed how a neural network that is trained to convert between synthesized *neutral* speech (obtained by passing the text through an off-the-shelf speech synthesizer) and

natural *expressive* speech learns to extract embeddings that capture expressiveness. We demonstrated that the resulting embeddings improve emotion recognition performance over surface acoustic features (e.g., MFBs). Further, we showed that the resulting embeddings outperform other embeddings obtained from unsupervised and self-supervised baselines.

8.2 Future Work

Data sparsity is an overwhelming challenge in SER research. One consequence of data sparsity is that it becomes difficult to train models to extract emotion from speech. Chapters VI and VII of this dissertation introduced methods that alleviate some of the problems that arise from the data sparsity challenge. Future work will expand on, and combine, these methods.

First, we propose to use data augmentation methods to further increase the amount of data and improve the robustness of the embeddings extracted using the framework introduced in Chapter VII. Data augmentation has been successfully used in previous research to improve the robustness of speech models (e.g., [99, 185, 146]). In Chapter III of this dissertation, for instance, we demonstrated how data augmentation via speed perturbation can result in a significant performance improvements in SER. In addition, the x-vector system used for extracting the embeddings used in Chapter VI employed several data augmentation strategies (e.g., additive noise and reverberation) to increase the robustness of speaker embeddings [146]. We expect data augmentation to improve the robustness of the embeddings that we learn to extract with our framework.

Second, we propose to extend our study in Chapter VII to include datasets with a large number of speakers. We used the Blizzard dataset in our study because of its highly expressive and dynamic speech content. However, the single-speaker nature of the Blizzard dataset can impact the generalization prospects of the learned

embedding extractor. Thus, we propose to investigate using the LibriSpeech dataset in our framework [162]. Similar to the Blizzard dataset, the LibriSpeech dataset consists of audiobook data which contain both speech and text. However, unlike the Blizzard dataset, the LibriSpeech dataset contains a more diverse set of speakers (more than 9,000 speakers). Although it is intuitive to think that the addition of more speakers can yield better embeddings, we note that the speech content of the LibriSpeech dataset is less expressive than that of the Blizzard dataset. Thus, a thorough investigation needs to be carried before determining the utility of using LibriSpeech in our framework.

Finally, we propose to investigate how changes to the architecture of the style encoder used in Chapter VII impacts the quality of the extracted embeddings. One clear limitation of the style encoder proposed in Chapter VII is that it induces fixed size embeddings by taking mean of the output sequence from the last LSTM layer. As discussed in Chapter III, taking the mean over all acoustic frames assumes that all frames are considered equally important. However, some frames might not contain information that is relevant for capturing speech expressiveness. As a result, a mean operation can obfuscate the information from relevant frames with those from irrelevant frames. One method to alleviate this challenge is through the use of a max pooling operation as we demonstrated in Chapter III. Another method to alleviate this by the introduction of an an attention mechanism in the style encoder to select relevant frames for the embeddings [186].

8.3 Work Published

- Part of Chapters I, II, and III, in Zakaria Aldeneh and Emily Mower Provost. "Using regional saliency for speech emotion recognition." *IEEE international conference on acoustics, speech and signal processing (ICASSP)*. 2017.

- Part of Chapters I, II, and IV in Zakaria Aldeneh, Soheil Khorram, Dimitrios Dimitriadis, and Emily Mower Provost. "Pooling acoustic and lexical features for the prediction of valence." *Proceedings of the 19th ACM International Conference on Multimodal Interaction (ICMI)*. 2017.

- Part of Chapters I, II, and V in Soheil Khorram, Zakaria Aldeneh, Dimitrios Dimitriadis, Melvin McInnis, and Emily Mower Provost. "Capturing long-term temporal dependencies with convolutional networks for continuous emotion recognition." *INTERSPEECH*. 2017.

- Part of Chapters I, II, and VI in Zakaria Aldeneh and Emily Mower Provost. "You're Not You When You're Angry: Robust Emotion Features Emerge by Recognizing Speakers." *IEEE Transactions on Affective Computing*. 2020. (*in submission*)

- Part of Chapters I, II, and VII in Zakaria Aldeneh, Mathew Perez, and Emily Mower Provost. "Learning Paralinguistic Attributes from Audiobooks with Voice Conversion." *AAAI Conference on Artificial Intelligence*. 2021. (*in submission*)

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] Alessandro Vinciarelli et al. "Open challenges in modelling, analysis and synthesis of human behaviour in human–human and human–machine interactions". In: *Cognitive Computation* 7.4 (2015), pp. 397–413.

[2] Z. Karam et al. "Ecologically valid long-term mood monitoring of individuals with bipolar disorder using speech". In: *ICASSP*. 2014.

[3] Nicholas Cummins et al. "A review of depression and suicide risk assessment using speech analysis". In: *Speech Communication* 71 (2015), pp. 10–49.

[4] Rafael A Calvo and Sidney D'Mello. "Affect detection: An interdisciplinary review of models, methods, and their applications". In: *IEEE Transactions on affective computing* 1.1 (2010), pp. 18–37.

[5] Anna Esposito, Antonietta M Esposito, and Carl Vogel. "Needs and challenges in human computer interaction for processing social emotional information". In: *Pattern Recognition Letters* 66 (2015), pp. 41–51.

[6] Anastasia Pampouchidou et al. "Automatic assessment of depression based on visual cues: A systematic review". In: *IEEE Transactions on Affective Computing* (2017).

[7] John Gideon, Simon Stent, and Luke Fletcher. "A Multi-Camera Deep Neural Network for Detecting Elevated Alertness in Drivers". In: *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2018, pp. 2931–2935.

[8] Qiang Ji, Peilin Lan, and Carl Looney. "A probabilistic framework for modeling and real-time monitoring human fatigue". In: *IEEE Transactions on systems, man, and cybernetics-Part A: Systems and humans* 36.5 (2006), pp. 862–875.

[9] Joseph G Ellis et al. "Predicting evoked emotions in video". In: *2014 IEEE International Symposium on Multimedia*. IEEE. 2014, pp. 287–294.

[10] Zhiwei Deng et al. "Factorized variational autoencoders for modeling audience reactions to movies". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 2577–2586.

[11] Diane Litman and Kate Forbes-Riley. "Predicting student emotions in computer-human tutoring dialogues". In: *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)*. 2004, pp. 351–358.

[12] Carlos Busso, Sungbok Lee, and Shrikanth Narayanan. "Analysis of emotionally salient aspects of fundamental frequency for emotion detection". In: *IEEE transactions on audio, speech, and language processing* 17.4 (2009), pp. 582–596.

[13] Soroosh Mariooryad and Carlos Busso. "Facial expression recognition in the presence of speech using blind lexical compensation". In: *IEEE Transactions on Affective Computing* 7.4 (2015), pp. 346–359.

[14] Yelin Kim and Emily Mower Provost. "Say cheese vs. smile: Reducing speech-related variability for facial emotion recognition". In: *Proceedings of the 22nd ACM international conference on Multimedia*. 2014, pp. 27–36.

[15] Carlos Busso et al. "IEMOCAP: Interactive emotional dyadic motion capture database". In: *Language resources and evaluation* 42.4 (2008), p. 335.

[16] Foteini Agrafioti, Dimitris Hatzinakos, and Adam K Anderson. "ECG pattern analysis for emotion detection". In: *IEEE Transactions on affective computing* 3.1 (2011), pp. 102–115.

[17] S Jerritta et al. "Physiological signals based human emotion recognition: a review". In: *2011 IEEE 7th International Colloquium on Signal Processing and its Applications*. IEEE. 2011, pp. 410–415.

[18] Rosalind W. Picard, Elias Vyzas, and Jennifer Healey. "Toward machine emotional intelligence: Analysis of affective physiological state". In: *IEEE transactions on pattern analysis and machine intelligence* 23.10 (2001), pp. 1175–1191.

[19] Paul Ekman. "An argument for basic emotions". In: *Cognition & emotion* 6.3-4 (1992), pp. 169–200.

[20] Keith Oatley and Philip N Johnson-Laird. "Towards a cognitive theory of emotions". In: *Cognition and emotion* 1.1 (1987), pp. 29–50.

[21] Silvan S Tomkins. "Affect theory". In: *Approaches to emotion* 163.163-195 (1984).

[22] Jaak Panksepp. "Toward a general psychobiological theory of emotions". In: *Behavioral and Brain sciences* 5.3 (1982), pp. 407–422.

[23] Wilhelm Max Wundt and Charles Hubbard Judd. *Outlines of psychology*. Vol. 1. Scholarly Press, 1897.

[24] Harold Schlosberg. "Three dimensions of emotion." In: *Psychological review* 61.2 (1954), p. 81.

[25] James A Russell. "A circumplex model of affect." In: *Journal of personality and social psychology* 39.6 (1980), p. 1161.

[26] Albert Mehrabian. *Basic dimensions for a general psychological theory: Implications for personality, social, environmental, and developmental studies*. Oelgeschlager, Gunn & Hain Cambridge, MA, 1980.

[27] Paul Ekman. "Basic emotions". In: *Handbook of cognition and emotion* 98.45-60 (1999), p. 16.

[28] Carroll E Izard. *Human emotions*. Springer Science & Business Media, 2013.

[29] John M Zelenski and Randy J Larsen. "The distribution of basic emotions in everyday life: A state and trait perspective from experience sampling data". In: *Journal of Research in Personality* 34.2 (2000), pp. 178–197.

[30] Emily K Mower. "Emotions in Engineering: Methods for the Interpretation of Ambiguous Emotional Content". PhD thesis. University of Southern California, 2011.

[31] James A Russell. "Core affect and the psychological construction of emotion." In: *Psychological review* 110.1 (2003), p. 145.

[32] Jonathan Posner, James A Russell, and Bradley S Peterson. "The circumplex model of affect: An integrative approach to affective neuroscience, cognitive development, and psychopathology". In: *Development and psychopathology* 17.3 (2005), p. 715.

[33] Hao Hu, Ming-Xing Xu, and Wei Wu. "GMM supervector based SVM with spectral features for speech emotion recognition". In: *2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP'07*. Vol. 4. IEEE. 2007, pp. IV–413.

[34] Iker Luengo et al. "Automatic emotion recognition using prosodic parameters". In: *Ninth European Conference on Speech Communication and Technology*. 2005.

[35] Friedhelm Schwenker et al. "The GMM-SVM supervector approach for the recognition of the emotional status from speech". In: *International conference on artificial neural networks*. Springer. 2009, pp. 894–903.

[36] Carl E Williams and Kenneth N Stevens. "Emotions and speech: Some acoustical correlates". In: *The Journal of the Acoustical Society of America* 52.4B (1972), pp. 1238–1250.

[37] Jo-Anne Bachorowski and Michael J Owren. "Vocal expression of emotion: Acoustic properties of speech are associated with emotional intensity and context". In: *Psychological science* 6.4 (1995), pp. 219–224.

[38] Klaus R Scherer. "Vocal communication of emotion: A review of research paradigms". In: *Speech communication* 40.1-2 (2003), pp. 227–256.

[39] B. Schuller, S. Steidl, and A. Batliner. "The INTERSPEECH 2009 emotion challenge." In: *Proceedings of Interspeech*. 2009.

[40] Björn Schuller et al. "The INTERSPEECH 2013 computational paralinguistics challenge: Social signals, conflict, emotion, autism". In: *Proceedings INTERSPEECH 2013, 14th Annual Conference of the International Speech Communication Association, Lyon, France*. 2013.

[41] Florian Eyben et al. "The Geneva minimalistic acoustic parameter set (GeMAPS) for voice research and affective computing". In: *IEEE Transactions on Affective Computing* 7.2 (2016), pp. 190–202.

[42] Sayan Ghosh et al. "Learning representations of affect from speech". In: *arXiv preprint arXiv:1511.04747* (2015).

[43] Mousmita Sarma et al. "Emotion Identification from Raw Speech Signals Using DNNs." In: *Interspeech*. 2018, pp. 3097–3101.

[44] Biqiao Zhang et al. "f-Similarity Preservation Loss for Soft Labels: A Demonstration on Cross-Corpus Speech Emotion Recognition". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 33. 2019, pp. 5725–5732.

[45] S. Latif et al. "Multi-Task Semi-Supervised Adversarial Autoencoding for Speech Emotion Recognition". In: *IEEE Transactions on Affective Computing* (2020), pp. 1–1.

[46] Soheil Khorram, Melvin McInnis, and Emily Mower Provost. "Jointly aligning and predicting continuous emotion annotations". In: *IEEE Transactions on Affective Computing* (2019).

[47] Zixiaofan Yang and Julia Hirschberg. "Predicting Arousal and Valence from Waveforms and Spectrograms Using Deep Neural Networks." In: *INTERSPEECH*. 2018, pp. 3092–3096.

[48] Siddique Latif et al. "Direct modelling of speech emotion from raw speech". In: *arXiv preprint arXiv:1904.03833* (2019).

[49] G. Trigeorgis et al. "ADIEU features? End-to-end speech emotion recognition using a deep convolutional recurrent network". In: *ICASSP*. 2016.

[50] S. Parthasarathy and C. Busso. "Ladder Networks for Emotion Recognition: Using Unsupervised Auxiliary Tasks to Improve Predictions of Emotional At-

tributes". In: *Interspeech 2018*. Hyderabad, India, Sept. 2018, pp. 3698–3702. DOI: 10.21437/Interspeech.2018-1391.

[51] Biqiao Zhang, Georg Essl, and Emily Mower Provost. "Automatic recognition of self-reported and perceived emotion: Does joint modeling help?" In: *Proceedings of the 18th ACM International Conference on Multimodal Interaction*. 2016, pp. 217–224.

[52] Biqiao Zhang, Soheil Khorram, and Emily Mower Provost. "Exploiting acoustic and lexical properties of phonemes to recognize valence from speech". In: *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2019, pp. 5871–5875.

[53] Reza Lotfian and Carlos Busso. "Building naturalistic emotionally balanced speech corpus by retrieving emotional speech from existing podcast recordings". In: *IEEE Transactions on Affective Computing* (2017).

[54] Samuel Albanie et al. "Emotion recognition in speech using cross-modal transfer in the wild". In: *Proceedings of the 26th ACM international conference on Multimedia*. 2018, pp. 292–301.

[55] John Gideon et al. "Progressive neural networks for transfer learning in emotion recognition". In: *arXiv preprint arXiv:1706.03256* (2017).

[56] Bjorn Schuller et al. "Cross-corpus acoustic emotion recognition: Variances and strategies". In: *IEEE Transactions on Affective Computing* 1.2 (2010), pp. 119–131.

[57] Tauhidur Rahman and Carlos Busso. "A personalized emotion recognition system using an unsupervised feature adaptation scheme". In: *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2012, pp. 5117–5120.

[58] Carlos Busso et al. "Iterative feature normalization scheme for automatic emotion detection from speech". In: *IEEE transactions on Affective computing* 4.4 (2013), pp. 386–397.

[59] Mohammed Abdelwahab and Carlos Busso. "Domain adversarial for acoustic emotion recognition". In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 26.12 (2018), pp. 2423–2435.

[60] Zixing Zhang et al. "Unsupervised learning in cross-corpus acoustic emotion recognition". In: *2011 IEEE Workshop on Automatic Speech Recognition & Understanding*. IEEE. 2011, pp. 523–528.

[61] Saurabh Sahu, Rahul Gupta, and Carol Espy-Wilson. "On enhancing speech emotion recognition using generative adversarial networks". In: *arXiv preprint arXiv:1806.06626* (2018).

[62] John Gideon, Melvin McInnis, and Emily Mower Provost. "Improving Cross-Corpus Speech Emotion Recognition with Adversarial Discriminative Domain Generalization (ADDoG)". In: *IEEE Transactions on Affective Computing* (2019).

[63] Biqiao Zhang, Emily Mower Provost, and Georg Essi. "Cross-corpus acoustic emotion recognition from singing and speaking: A multi-task learning approach". In: *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2016, pp. 5805–5809.

[64] Haoqi Li et al. "Speaker-Invariant Affective Representation Learning via Adversarial Training". In: *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2020, pp. 7144–7148.

[65] D. Le and E. Mower Provost. "Emotion recognition from spontaneous speech using hidden markov models with deep belief networks". In: *Automatic Speech Recognition and Understanding (ASRU)*. 2013.

[66] Yu-An Chen et al. "Linear regression-based adaptation of music emotion recognition models for personalization". In: *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2014, pp. 2149–2153.

[67] Srinivas Parthasarathy and Carlos Busso. "Jointly Predicting Arousal, Valence and Dominance with Multi-Task Learning." In: *Interspeech*. 2017, pp. 1103–1107.

[68] Martin Wöllmer et al. "Abandoning emotion classes-towards continuous emotion recognition with modelling of long-range dependencies". In: *Proc. 9th Interspeech 2008 incorp. 12th Australasian Int. Conf. on Speech Science and Technology SST 2008, Brisbane, Australia*. 2008, pp. 597–600.

[69] Erik M Schmidt and Youngmoo E Kim. "Prediction of Time-varying Musical Mood Distributions from Audio." In: *ISMIR*. 2010, pp. 465–470.

[70] Erik M Schmidt and Youngmoo E Kim. "Modeling Musical Emotion Dynamics with Conditional Random Fields." In: *ISMIR*. Miami (Florida), USA. 2011, pp. 777–782.

[71] Ju-Chiang Wang et al. "Modeling the affective content of music with a Gaussian mixture model". In: *IEEE Transactions on Affective Computing* 6.1 (2015), pp. 56–68.

[72] Biqiao Zhang, Georg Essl, and Emily Mower Provost. "Predicting the distribution of emotion perception: capturing inter-rater variability". In: *Proceedings of the 19th ACM International Conference on Multimodal Interaction*. 2017, pp. 51–59.

[73] Arsha Nagrani, Joon Son Chung, and Andrew Zisserman. "Voxceleb: a large-scale speaker identification dataset". In: *arXiv preprint arXiv:1706.08612* (2017).

[74] Qin Jin et al. "Speech emotion recognition with acoustic and lexical features". In: *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE. 2015, pp. 4749–4753.

[75] Maarten Brilman and Stefan Scherer. "A multimodal predictive model of successful debaters or how i learned to sway votes". In: *Proceedings of the 23rd ACM international conference on Multimedia*. ACM. 2015, pp. 149–158.

[76] Soujanya Poria, Erik Cambria, and Alexander F Gelbukh. "Deep Convolutional Neural Network Textual Features and Multiple Kernel Learning for Utterance-level Multimodal Sentiment Analysis." In: *EMNLP*. 2015, pp. 2539–2544.

[77] Björn Schuller et al. "The INTERSPEECH 2011 speaker state challenge". In: *Twelfth Annual Conference of the International Speech Communication Association*. 2011.

[78] Ali Sharif Razavian et al. "CNN features off-the-shelf: an astounding baseline for recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*. 2014, pp. 806–813.

[79] Jason Yosinski et al. "How transferable are features in deep neural networks?" In: *Advances in neural information processing systems*. 2014, pp. 3320–3328.

[80] C. Busso et al. "MSP-IMPROV: An acted corpus of dyadic interactions to study emotion perception". In: *IEEE Transactions on Affective Computing* (2015).

[81] Fabien Ringeval et al. "Introducing the RECOLA multimodal corpus of remote collaborative and affective interactions". In: *International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*. IEEE. 2013, pp. 1–8.

[82] Michel Valstar et al. "AVEC 2016: Depression, Mood, and Emotion Recognition Workshop and Challenge". In: *Proceedings of the 6th International Workshop on Audio/Visual Emotion Challenge*. ACM. 2016, pp. 3–10.

[83] Jacob Sager et al. "VESUS: A Crowd-Annotated Database to Study Emotion Production and Perception in Spoken English." In: *INTERSPEECH*. 2019, pp. 316–320.

[84] E. Mower Provost, M. Mataric, and S. Narayanan. "A framework for automatic human emotion classification using emotion profiles". In: *IEEE Transactions on Audio, Speech, and Language Processing* (2011).

[85] R. Xia and Y. Liu. "A Multi-task Learning Framework for Emotion Recognition Using 2D Continuous Space". In: *IEEE Transactions on Affective Computing* (2016).

[86] D. Le and E. Mower Provost. "Data selection for acoustic emotion recognition: Analyzing and comparing utterance and sub-utterance selection strategies". In: *Affective Computing and Intelligent Interaction (ACII)*. 2015.

[87] Y. Kim and E. Mower Provost. "Emotion Spotting: Discovering Regions of Evidence in Audio-Visual Emotion Expressions." In: *ACM International Conference on Multimodal Interaction (ICMI)*. 2016.

[88] Y. Kim and E. Mower Provost. "Emotion classification via utterance-level dynamics: A pattern-based approach to characterizing affective expressions". In: *ICASSP*. 2013.

[89] T. Sainath et al. "Deep convolutional neural networks for LVCSR". In: *ICASSP*. 2013.

[90] A. Krizhevsky, I. Sutskever, and G. Hinton. "Imagenet classification with deep convolutional neural networks". In: *Advances in neural information processing systems (NIPS)*. 2012.

[91] Yoon Kim. "Convolutional neural networks for sentence classification". In: *arXiv preprint arXiv:1408.5882* (2014).

[92] Q. Mao et al. "Learning salient features for speech emotion recognition using convolutional neural networks". In: *IEEE Transactions on Multimedia* (2014).

[93] D. Yu and L. Deng. *Automatic Speech Recognition*. Springer, 2012.

[94] K. Han, D. Yu, and I. Tashev. "Speech emotion recognition using deep neural network and extreme learning machine". In: *Proceedings of Interspeech*. 2014.

[95] J. Lee and I. Tashev. "High-level feature representation using recurrent neural network for speech emotion recognition". In: *Proceedings of Interspeech*. 2015.

[96] R. Xia et al. "Modeling gender information for emotion recognition using denoising autoencoder". In: *ICASSP*. 2014.

[97] M. Zeiler et al. "On rectified linear units for speech processing". In: *ICASSP*. 2013.

[98] F. Eyben et al. "Recent developments in openSMILE, the munich open-source multimedia feature extractor". In: *ACM international conference on Multimedia*. 2013.

[99] T. Ko et al. "Audio augmentation for speech recognition". In: *Proceedings of Interspeech*. 2015.

[100] K. He et al. “Delving deep into rectifiers: Surpassing human-level performance on imagenet classification”. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2015.

[101] Tijmen Tieleman and Geoffrey Hinton. “Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude”. In: *COURSERA: Neural networks for machine learning* 4.2 (2012).

[102] Y. Zhang and B. Wallace. “A Sensitivity Analysis of (and Practitioners’ Guide to) Convolutional Neural Networks for Sentence Classification”. In: *arXiv preprint arXiv:1510.03820* (2015).

[103] Akira Fukui et al. “Multimodal compact bilinear pooling for visual question answering and visual grounding”. In: *arXiv preprint arXiv:1606.01847* (2016).

[104] Tsung-Yu Lin, Aruni RoyChowdhury, and Subhransu Maji. “Bilinear cnn models for fine-grained visual recognition”. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2015, pp. 1449–1457.

[105] Linchuan Li et al. “Combining CNN and BLSTM to Extract Textual and Acoustic Features for Rec-ognizing Stances in Mandarin Ideological Debate Competition”. In: *Interspeech 2016* (2016), pp. 1392–1396.

[106] Verónica Pérez-Rosas, Rada Mihalcea, and Louis-Philippe Morency. “Utterance-Level Multimodal Sentiment Analysis.” In: *ACL (1)*. 2013, pp. 973–982.

[107] Florian Eyben, Martin Wöllmer, and Björn Schuller. “Opensmile: the munich versatile and fast open-source audio feature extractor”. In: *Proceedings of the 18th ACM international conference on Multimedia*. ACM. 2010, pp. 1459–1462.

[108] Jonathan Chang and Stefan Scherer. “Learning representations of emotional speech with deep convolutional generative adversarial networks”. In: *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*. IEEE. 2017.

[109] Tara N Sainath et al. “Learning filter banks within a deep neural network framework”. In: *Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop on*. IEEE. 2013, pp. 297–302.

[110] Carlos Busso, Sungbok Lee, and Shrikanth S Narayanan. “Using neutral speech models for emotional speech analysis”. In: *Eighth Annual Conference of the International Speech Communication Association*. 2007.

[111] Junyoung Chung et al. “Empirical evaluation of gated recurrent neural networks on sequence modeling”. In: *arXiv preprint arXiv:1412.3555* (2014).

[112] Yang Gao et al. “Compact bilinear pooling”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016, pp. 317–326.

[113] Hamed Pirsiavash, Deva Ramanan, and Charless C Fowlkes. "Bilinear classifiers for visual recognition". In: *Advances in neural information processing systems*. 2009, pp. 1482–1490.

[114] Ninh Pham and Rasmus Pagh. "Fast and scalable polynomial kernels via explicit feature maps". In: *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM. 2013, pp. 239–247.

[115] Andrew Rosenberg. "Classifying Skewed Data: Importance Weighting to Optimize Average Recall." In: *Interspeech*. 2012, pp. 2242–2245.

[116] François Chollet et al. *Keras*. `https://github.com/fchollet/keras`. 2015.

[117] Martıén Abadi et al. "Tensorflow: Large-scale machine learning on heterogeneous distributed systems". In: *arXiv preprint arXiv:1603.04467* (2016).

[118] Michael Lewis, Jeannette M Haviland-Jones, and Lisa Feldman Barrett. *Handbook of emotions*. Guilford Press, 2010.

[119] Patrick Cardinal et al. "ETS System for AVEC 2015 Challenge". In: *Proceedings of the 5th International Workshop on Audio/Visual Emotion Challenge*. ACM. 2015, pp. 17–23.

[120] Lang He et al. "Multimodal affective dimension prediction using deep bidirectional long short-term memory recurrent neural networks". In: *Proceedings of the 5th International Workshop on Audio/Visual Emotion Challenge*. ACM. 2015, pp. 73–80.

[121] Kevin Brady et al. "Multi-Modal Audio, Video and Physiological Sensor Learning for Continuous Emotion Prediction". In: *Proceedings of the 6th International Workshop on Audio/Visual Emotion Challenge*. ACM. 2016, pp. 97–104.

[122] Filip Povolny et al. "Multimodal Emotion Recognition for AVEC 2016 Challenge". In: *Proceedings of the 6th International Workshop on Audio/Visual Emotion Challenge*. ACM. 2016, pp. 75–82.

[123] Tom Sercu and Vaibhava Goel. "Dense Prediction on Sequences with Time-Dilated Convolutions for Speech Recognition". In: *arXiv preprint arXiv:1611.09288* (2016).

[124] Daniel Povey et al. "The Kaldi speech recognition toolkit". In: *workshop on automatic speech recognition and understanding (ASRU)*. IEEE. 2011.

[125] Fisher Yu and Vladlen Koltun. "Multi-scale context aggregation by dilated convolutions". In: *arXiv preprint arXiv:1511.07122* (2015).

[126] Aäron van den Oord et al. "Wavenet: A generative model for raw audio". In: *CoRR abs/1609.03499* (2016).

[127] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. "Learning deconvolution network for semantic segmentation". In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE. 2015, pp. 1520–1528.

[128] Matthew D Zeiler et al. "Deconvolutional networks". In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE. 2010, pp. 2528–2535.

[129] Vincent Dumoulin and Francesco Visin. "A guide to convolution arithmetic for deep learning". In: *arXiv preprint arXiv:1603.07285* (2016).

[130] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. "Segnet: A deep convolutional encoder-decoder architecture for image segmentation". In: *arXiv preprint arXiv:1511.00561* (2015).

[131] Weifeng Chen et al. "Single-image depth perception in the wild". In: *Advances in Neural Information Processing Systems (NIPS)*. 2016, pp. 730–738.

[132] Se Rim Park and Jinwon Lee. "A Fully Convolutional Neural Network for Speech Enhancement". In: *arXiv preprint arXiv:1609.07132* (2016).

[133] James Bergstra et al. "Theano: a CPU and GPU Math Expression Compiler". In: *Proceedings of the Python for Scientific Computing Conference (SciPy)*. Austin, TX, 2010.

[134] Duc Le, Zakaria Aldeneh, and Emily Mower Provost. "Discretized Continuous Speech Emotion Recognition with Multi-Task Deep Recurrent Neural Network". In: *Interspeech, 2017 (to apear)*. 2017.

[135] Siddique Latif et al. "Multi-task semi-supervised adversarial autoencoding for speech emotion recognition". In: *IEEE Transactions on Affective Computing* (2020).

[136] Mitchell McLaren et al. "The Speakers in the Wild (SITW) speaker recognition database." In: *Interspeech*. 2016, pp. 818–822.

[137] A. Nagrani, J. S. Chung, and A. Zisserman. "VoxCeleb: a large-scale speaker identification dataset". In: *INTERSPEECH*. 2017.

[138] J. S. Chung, A. Nagrani, and A. Zisserman. "VoxCeleb2: Deep Speaker Recognition". In: *INTERSPEECH*. 2018.

[139] Wei Wu et al. "Study on speaker verification on emotional speech". In: *Ninth International Conference on Spoken Language Processing*. 2006.

[140] Srinivas Parthasarathy and Carlos Busso. "Predicting speaker recognition reliability by considering emotional content". In: *2017 Seventh International Conference on Affective Computing and Intelligent Interaction (ACII)*. IEEE. 2017, pp. 434–439.

[141] Srinivas Parthasarathy et al. "A study of speaker verification performance with expressive speech". In: *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2017, pp. 5540–5544.

[142] Michelle Bancroft et al. "Exploring the Intersection Between Speaker Verification and Emotion Recognition". In: *2019 8th International Conference on Affective Computing and Intelligent Interaction Workshops and Demos (ACIIW)*. IEEE. 2019, pp. 337–342.

[143] John HL Hansen and Taufiq Hasan. "Speaker recognition by machines and humans: A tutorial review". In: *IEEE Signal processing magazine* 32.6 (2015), pp. 74–99.

[144] Ehsan Variani et al. "Deep neural networks for small footprint text-dependent speaker verification". In: *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2014, pp. 4052–4056.

[145] Quan Wang et al. "Speaker diarization with lstm". In: *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2018, pp. 5239–5243.

[146] David Snyder et al. "X-vectors: Robust dnn embeddings for speaker recognition". In: *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2018, pp. 5329–5333.

[147] Li Wan et al. "Generalized end-to-end loss for speaker verification". In: *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2018, pp. 4879–4883.

[148] Raghavendra Pappagari et al. "x-vectors meet emotions: A study on dependencies between emotion and speaker recognition". In: *arXiv preprint arXiv:2002.05039* (2020).

[149] Björn Schuller et al. "The interspeech 2012 speaker trait challenge". In: *Thirteenth Annual Conference of the International Speech Communication Association*. 2012.

[150] Soroosh Mariooryad and Carlos Busso. "Compensating for speaker or lexical variabilities in speech for emotion recognition". In: *Speech Communication* 57 (2014), pp. 1–12.

[151] David Snyder et al. "Deep Neural Network Embeddings for Text-Independent Speaker Verification." In: *Interspeech*. 2017, pp. 999–1003.

[152] Zhong Meng et al. "Speaker-invariant training via adversarial learning". In: *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2018, pp. 5969–5973.

[153] George Saon et al. “Speaker adaptation of neural network acoustic models using i-vectors”. In: *2013 IEEE Workshop on Automatic Speech Recognition and Understanding*. IEEE. 2013, pp. 55–59.

[154] Desh Raj et al. “Probing the Information Encoded in x-vectors”. In: *arXiv preprint arXiv:1909.06351* (2019).

[155] David Snyder et al. “Speaker recognition for multi-speaker conversations using x-vectors”. In: *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2019, pp. 5796–5800.

[156] Alan McCree, Gregory Sell, and Daniel Garcia-Romero. “Speaker Diarization Using Leave-One-Out Gaussian PLDA Clustering of DNN Embeddings.” In: *Interspeech*. 2019, pp. 381–385.

[157] Jennifer Williams and Simon King. “Disentangling Style Factors from Speaker Representations”. In: *Proc. Interspeech*. Vol. 2019. 2019, pp. 3945–3949.

[158] Erik Marchi et al. “A novel approach for automatic acoustic novelty detection using a denoising autoencoder with bidirectional LSTM neural networks”. In: *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE. 2015, pp. 1996–2000.

[159] Bo Zong et al. “Deep autoencoding gaussian mixture model for unsupervised anomaly detection”. In: (2018).

[160] Mohammad Sabokrou et al. “Adversarially learned one-class classifier for novelty detection”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 3379–3388.

[161] Laura Beggel, Michael Pfeiffer, and Bernd Bischl. “Robust anomaly detection in images using adversarial autoencoders”. In: *arXiv preprint arXiv:1901.06355* (2019).

[162] Vassil Panayotov et al. “Librispeech: an asr corpus based on public domain audio books”. In: *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2015, pp. 5206–5210.

[163] Douglas Bates et al. “Package ‘lme4’”. In: *Convergence* 12.1 (2015), pp. 470–474.

[164] R Core Team et al. “R: A language and environment for statistical computing”. In: (2013).

[165] Dale J Barr et al. “Random effects structure for confirmatory hypothesis testing: Keep it maximal”. In: *Journal of memory and language* 68.3 (2013), pp. 255–278.

[166] Mohammed Abdelwahab and Carlos Busso. "Study of dense network approaches for speech emotion recognition". In: *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2018, pp. 5084–5088.

[167] Srinivas Parthasarathy and Carlos Busso. "Semi-supervised speech emotion recognition with ladder networks". In: *arXiv preprint arXiv:1905.02921* (2019).

[168] Björn Schuller et al. "The INTERSPEECH 2010 paralinguistic challenge". In: *Eleventh Annual Conference of the International Speech Communication Association*. 2010.

[169] Florian Eyben et al. "The Geneva minimalistic acoustic parameter set (GeMAPS) for voice research and affective computing". In: *IEEE transactions on affective computing* 7.2 (2015), pp. 190–202.

[170] Reza Lotfian and Carlos Busso. "Emotion recognition using synthetic speech as neutral reference". In: *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2015, pp. 4759–4763.

[171] Yu-An Chung et al. "An unsupervised autoregressive model for speech representation learning". In: *arXiv preprint arXiv:1904.03240* (2019).

[172] Wei-Ning Hsu, Yu Zhang, and James Glass. "Unsupervised learning of disentangled and interpretable representations from sequential data". In: *Advances in neural information processing systems*. 2017, pp. 1878–1889.

[173] Seyed Hamidreza Mohammadi and Alexander Kain. "An overview of voice conversion systems". In: *Speech Communication* 88 (2017), pp. 65–82.

[174] Jian Gao et al. "Nonparallel emotional speech conversion". In: *arXiv preprint arXiv:1811.01174* (2018).

[175] Ravi Shankar et al. "Automated Emotion Morphing in Speech Based on Diffeomorphic Curve Registration and Highway Networks." In: *INTERSPEECH*. 2019, pp. 4499–4503.

[176] Ravi Shankar, Jacob Sager, and Archana Venkataraman. "A Multi-Speaker Emotion Morphing Model Using Highway Networks and Maximum Likelihood Objective." In: *INTERSPEECH*. 2019, pp. 2848–2852.

[177] Yuxuan Wang et al. "Style tokens: Unsupervised style modeling, control and transfer in end-to-end speech synthesis". In: *arXiv preprint arXiv:1803.09017* (2018).

[178] Ya-Jie Zhang et al. "Learning latent representations for style control and transfer in end-to-end speech synthesis". In: *ICASSP 2019-2019 IEEE International*

Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE. 2019, pp. 6945–6949.

[179] Brian McFee et al. "librosa: Audio and music signal analysis in python". In: *Proceedings of the 14th python in science conference*. Vol. 8. 2015, pp. 18–25.

[180] Jing-Xuan Zhang, Zhen-Hua Ling, and Li-Rong Dai. "Non-parallel sequence-to-sequence voice conversion with disentangled linguistic and speaker representations". In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 28 (2019), pp. 540–552.

[181] Jacob Benesty, M Mohan Sondhi, and Yiteng Huang. *Springer handbook of speech processing*. Springer, 2007.

[182] Masanori Morise, Fumiya Yokomori, and Kenji Ozawa. "WORLD: a vocoder-based high-quality speech synthesis system for real-time applications". In: *IEICE TRANSACTIONS on Information and Systems* 99.7 (2016), pp. 1877–1884.

[183] Jing-Xuan Zhang et al. "Sequence-to-sequence acoustic modeling for voice conversion". In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 27.3 (2019), pp. 631–644.

[184] Sefik Emre Eskimez, Zhiyao Duan, and Wendi Heinzelman. "Unsupervised learning approach to feature analysis for automatic speech emotion recognition". In: *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2018, pp. 5099–5103.

[185] Tom Ko et al. "A study on data augmentation of reverberant speech for robust speech recognition". In: *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2017, pp. 5220–5224.

[186] Qiongqiong Wang et al. "Attention mechanism in speaker recognition: What does it learn in deep speaker embedding?" In: *2018 IEEE Spoken Language Technology Workshop (SLT)*. IEEE. 2018, pp. 1052–1059.