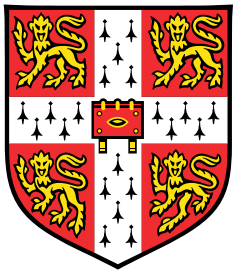
Speech-Based Emotion Modelling and Mental Disorder Detection



Wen Wu

Department of Engineering
University of Cambridge

This dissertation is submitted for the degree of
Doctor of Philosophy

Trinity College July 2024

Declaration

This thesis is the result of my own work and includes nothing which is the outcome of work done in collaboration except as declared in the preface and specified in the text. It is not substantially the same as any work that has already been submitted, or, is being concurrently submitted, for any degree, diploma or other qualification at the University of Cambridge or any other University or similar institution except as declared in the preface and specified in the text. It does not exceed the prescribed word limit for the relevant Degree Committee.

Wen Wu
July 2024

Acknowledgements

I would like to first devote profound gratitude to my PhD supervisor, Prof. Philip C. Woodland. I deeply appreciate the opportunity Phil offered me to undertake this challenging and rewarding PhD research. The countless hours dedicated to refining research directions, ideas and papers served as a constant source of motivation, support, and encouragement throughout my PhD studies. Moreover, Phil with his academic integrity and professionalism has been an exceptional role model, profoundly influencing me and for which I will be forever grateful.

I would like to thank Prof. Chao Zhang for his invaluable contributions to my research during my PhD studies. He is knowledgeable, patient, and always willing to help. His impressive level of engagement in the research discussions, along with his insightful suggestions is greatly appreciated. I would also like to express my gratitude to my collaborators at Google, Dr. Bo Li, Dr. Chung-Cheng Chiu, Dr. Qiujia Li, Dr. Junwen Bai, and Dr. Tara N. Sainath, who provided tremendous support and expertise during my internship. I would also like to thank my collaborator Wenlin Chen from Computational and Biological Learning Laboratory for the valuable insights and discussions.

I would like to express my gratitude to Cambridge Trust for awarding me the International Student Scholarship. It is a great honour to become a Trust Scholar. I also want to thank Department of Engineering and Trinity College for providing me financial support to present my work at international conferences. I appreciate the assistance and support provided by my college tutor at Trinity, Prof. Richard Serjeantson, with conference funding applications. I also sincerely appreciate the warm support from my labmates at the Machine Intelligence Laboratory, Dongcheng Jiang, Xianrui Zheng, Xiaodong Wu, Keqi Deng, Rao Ma, Dr. Guangzhi Sun, Dr. Mengjie Qian, and Dr. Xiang Li who have helped me in various ways.

Finally, I would like to express my deepest gratitude to my parents who always unconditionally support me and unwaveringly stand by my side. I would like to dedicate this thesis to them.

Abstract

Emotion modelling and understanding are crucial for artificial intelligence (AI) systems to achieve enhanced contextual understanding and adaptive, personalised human-AI interaction. Speech contains important clues for detecting emotion through a variety of vocal characteristics such as prosody, along with speech patterns such as hesitation and laughter. This thesis first explores automatic emotion recognition (AER) from speech input. Current AER systems face two primary challenges: (i) the mismatch between research experiments and practical applications such as the use of reference transcriptions and sentence segmentation; (ii) the inconsistency of emotion annotations due to ambiguous expressions and subjective perception.

To tackle the first challenge, an integrated system is developed which integrates AER with speaker diarisation and speech recognition in a jointly-trained system. Compared to separately optimised cascaded systems, the proposed system achieves not only improved efficiency but also reduced recognition errors for emotional speech. In addition, two novel metrics are introduced to evaluate AER performance with automatic segmentation based on time-weighted emotion classification errors.

In response to the second AER challenge, it is proposed to represent emotion as a distribution rather than a single class. Different emotion annotations provided by human annotators are treated as samples drawn from the emotion distribution. Evidential deep learning (EDL) is used to quantify the uncertainty in emotion distribution estimation by learning an utterance-specific prior distribution. Representing emotion as a distribution offers not only a more comprehensive representation of emotional content but also an inclusive representation of human opinions.

The challenge of inconsistent human opinions extends beyond emotion annotation and affects various subjective tasks such as speech quality assessment and toxic speech detection. A general framework for human annotator simulation is introduced, which accounts for the variability in human judgements. The framework meta-learns a conditional flow model, which demonstrates superior capability and efficiency in predicting the aggregated behaviour of human annotators, matching the distribution of human annotations, and simulating inter-

annotator disagreements. It is hoped that the proposed methods could contribute to the promotion of inclusivity and fairness in ethical AI practices.

Furthermore, emotion is closely linked with mental wellbeing. A speech-based automatic depression detection system is introduced which uses foundation models pretrained on large speech datasets to alleviate the data sparsity issue of medical datasets. It is shown that incorporating emotion information is useful for depression detection. Integrating representations from multiple foundation models achieves state-of-the-art results without requiring oracle transcriptions. To enhance the reliability of automatic diagnosis systems, confidence estimation methods are studied. The proposed method builds upon the EDL approach introduced previously for emotion distribution estimation, adapting it to learn the predictive distribution of mental illness detection. This method aims to foster reliable and trustworthy automatic diagnostic systems.

Table of contents

List of figures

List of tables

Nomenclature

List of Symbols

$\boldsymbol{b}$	Bias vector
$\delta(\cdot)$	Dirac delta function
$\mathcal{D}$	Data
$\odot$	Element-wise multiplication
ϵ, λ	Scale coefficient
$\mathbb{E}[\cdot]$	Expectation
$\forall$	For all
$\Gamma(\cdot)$	Gamma function
$\boldsymbol{h}$	Hidden states
$\mathbb{I}(\cdot)$	Indicator function
$\boldsymbol{\kappa}$	Fleiss' kappa
$\mathcal{L}(\cdot)$	loss
$\boldsymbol{\mu}$	Mean vector
$\mathcal{N}$	Normal (Gaussian) distribution
$\boldsymbol{\sigma}^2$	Variance vector
$\sigma(\cdot)$	Activation function
$\mathbf{W}$	Weight matrix

$\boldsymbol{x}$ Input vector

$\boldsymbol{y}$ Output vector

$\boldsymbol{z}$ Latent variable

List of Acronyms / Abbreviations

AD Alzheimer's Disease

AER Automatic Emotion Recognition

AI Artificial Intelligence

ANN Artificial Neural Network

ASR Automatic Speech Recognition

BBB Bayes-By-Backprop

CCC Concordance Correlation Coefficient

CNF Conditional Normalising Flow

CNN Convolutional Neural Network

CTC Connectionist Temporal Classification

CVAE Conditional Variational AutoEncoder

DER Diarisation Error Rate

DNN Deep Neural Network

DPN Dirichlet Prior Network

ECDF Empirical Cumulative Distribution Function

ECE Expected Calibration Error

EDL Evidential Deep Learning

ELBO Evidence Lower BOund

FAR False Alarm Rate

FC Fully-Connected

FFN Feed-Forward Network

FN False Negative

FP False Positive

FPR False Positive Rate

GP Gaussian Process

GPU Graphics Processing Unit

HAS Human Annotator Simulation

ICC Intra-class Correlation Coefficient

KL Kullback–Leibler

LLM Large Language Model

LSTM Long Short-Term Memory

MA Majority Agreed

MCDP Monte Carlo DroPout

MCE Maximum Calibration Error

MFB Mel frequency FilterBank energy

MFCC Mel Frequency Cepstral Coefficient

MHA Multi-Head Attention

MLE Maximum Likelihood Estimate

MOS Mean Opinion Score

MSR Missed Speech Rate

NIG Normal-Inverse-Gamma

NLL Negative Log Likelihood

NLP Natural Language Processing

NMA No Majority Agreed

OOD Out-Of-Domain

PWLM Piece-Wise Linear Mapping

RMSE Root Mean Square Error

RNN Recurrent Neural Networks

SDD Speech-based Depression Detection

SER Speech Emotion Recognition

SOTA State-Of-The-Art

SSL Self-Supervised Learning

TP True Positive

TPR True Positive Rate

TTS Text-To-Speech

UAR Unweighted Average Recall

VAD Voice Activity Detection

W2V2 Wav2Vec 2.0

WER Word Error Rate

WHO World Health Organisation

Chapter 1

Introduction

Artificial intelligence (AI) has garnered significant attention in recent years, captivating the interest of both the scientific community and the general public. Advancements in deep learning have significantly accelerated the development of AI. Deep learning, a subfield of machine learning inspired by the structure and function of the human brain, utilises artificial neural networks to learn complex patterns from vast amounts of data. It has proven remarkably effective in areas like speech recognition, natural language processing, and computer vision, leading to significant breakthroughs in various AI applications.

Emotion is a key part of human behaviour. The ability of AI to comprehend and respond to human emotions is crucial for achieving advanced intelligence and human-like interaction. The primary research focus of the thesis is on automatic emotion recognition (AER), with the scope subsequently extended to mental illness detection. A route map of the thesis is shown in Fig. 1.1.

1.1 Automatic Emotion Recognition

AER has attracted increasing attention due to its wide range of potential applications in conversational chatbots, voice assistants, and mental health analysis, *etc*. An AER system predicts the speaker's emotion state based on various input modalities such as speech, text, facial expressions, gestures, as well as psychological signals. This thesis mainly focuses on understanding human emotions from speech. A major challenge for AER systems is the mismatch between research experiments and practical applications, which includes the naturalness of emotion data, the use of reference text and sentence segmentation, *etc*. An integrated system is proposed which addresses some of the above issues in an efficient way by integrating AER with speaker diarisation and speech recognition.

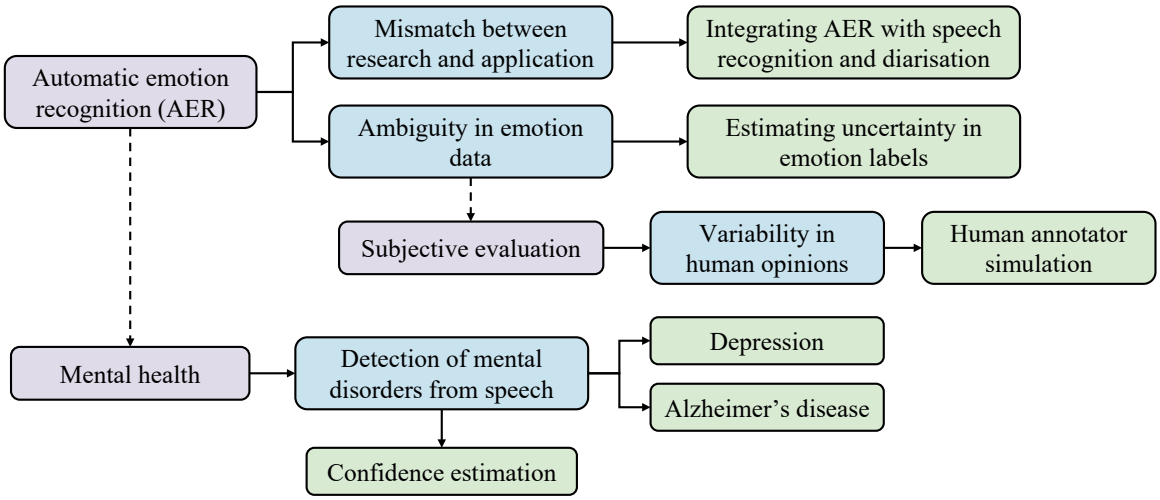


Fig. 1.1 Route map of the thesis.

Another challenge in AER lies in the inherent ambiguity and complexity of emotion. Human perception and interpretation of emotions are subjective. Different people could perceive the same utterance differently and there is no absolute right or wrong between different viewpoints. This leads to ambiguity in emotion expression and uncertainty in emotion annotation. This thesis studies various approaches to tackling this problem and designs a series of uncertainty estimation methods to explore better ways of representing emotions.

The scope is then extended from emotion to general subjective tasks where there is no single ground truth and human evaluations are usually involved for data annotation or model quality assessment (*e.g.*, perceptual quality evaluation of synthesised speech). Human evaluation is costly and time-consuming, which motivates the study of human annotator simulation (HAS). Most current HAS systems focus on predicting the majority/mean opinion. However, the majority/mean opinion can contain potential biases and over-representation. For example, in scoring quality of synthesised speech, the scores given by 20 individuals are unlikely to be identical, reflecting the variability of human opinions, which is crucial yet currently overlooked. Human annotators are employed for assessment because these subjective tasks inherently lack definitive answers. Therefore, it is imperative to account for the variability in human annotations rather than dedicating to obtaining a single correct answer, as no such absolute answer exists. This variability reflects the diverse perspectives and interpretations that human evaluators bring to the process, making it a critical aspect to consider when simulating human judgements. A variability-aware HAS system is proposed in this thesis, which considers this diversity rather than simply predicting the mainstream opinion to better simulate human perception and interpretation of the world.

1.2 Automatic Detection of Mental Disorders and Cognitive Diseases

Emotions are closely related to mental wellbeing, so another area of the thesis focuses on detection of mental illnesses and cognitive impairments, which includes depression and Alzheimer's disease (AD). Clinical depression is a psychiatric mood disorder, caused by an individual's difficulty in coping with stressful life events, and presents persistent feelings of sadness, negativity and difficulty dealing with everyday responsibilities. AD is the most common cause of dementia which is associated with an ongoing decline of brain functioning that can affect memory, thinking skills and other mental abilities. According to the world health organisation, approximately 280 million people in the world have depression (World Health Organisation, 2024). A 2019 report shows that 1 in every 14 of the population aged 65 years and over in the UK suffer from dementia and there will be over 1.5 million people with dementia in the UK, at the current rate of prevalence in 2040 (Wittenberg et al., 2019).

Early detection is important for timely intervention while a large proportion of patients remain undiagnosed due to, *e.g.,* costs and unawareness of the seriousness of the condition, which motivates the demand for automatic diagnosis. Unlike nodules and tumours, mental disorders (especially mild cases) usually lack definitive biomarkers and diagnosis is primarily based on clinical interview, which presents an opportunity for the automatic detection of mental disorders through speech analysis. Data scarcity is a major challenge in deep-learning-based automatic detection of mental illness. The thesis studies the use of foundation model pretrained on a large amount of unlabelled data to compensate the data sparsity issue in depression detection, which leverages generic speech knowledge. Apart from improving classification accuracy of automatic diagnosis systems, confidence estimation is also important to help reduce the risk of misdiagnosis. A confidence prediction method is then proposed that serves as a reference indicator for whether to trust the model's predictions.

1.3 Thesis Outline

An overview of each chapter is given in this section, including references to the corresponding publications[1].

Chapter 2: Deep Learning

This chapter covers the fundamental elements of deep learning, including their basic build-

[1]The list of published papers related to the thesis can be found in Appendix A.

ing blocks such as types of neural network layers, training algorithms and regularisation techniques.

Chapter 3: Automatic Emotion Recognition
This chapter introduces AER including a description of emotion states, a brief review of related work, benchmark emotion corpora, current challenges in AER, as well as an AER system based on foundation models.

Chapter 4: An Integrated System for AER and ASR with Automatic Segmentation
This chapter proposes an integrated system that combines AER with speaker diarisation and speech recognition. Two metrics are proposed to evaluate AER performance with automatic segmentation. This is the first work that considers emotion recognition with automatic segmentation and integrates emotion recognition, speech recognition and speaker diarisation into a jointly-trained model. This research work has been presented in the publication:

Wu et al. (2023b): **Wu, W.**, Zhang, C., and Woodland, P. C. (2023). Integrating emotion recognition with speech recognition and speaker diarisation for conversations. In *Proceedings of Interspeech 2023*.

Chapter 5: Handling Ambiguity in Emotion Class Labels
This chapter investigates three approaches to handling ambiguity in emotion. The emotion classification problem is transformed into an emotion distribution estimation problem to capture finer-grained emotion differences. Given an utterance with ambiguous emotion the proposed approach is able to provide a comprehensive representation of its emotion content as a distribution with a reliable uncertainty measure. This research work has been presented in the publication:

Wu et al. (2024b): **Wu, W.**, Li, B., Zhang, C., Chiu, C.-C., Li, Q., Bai, J., Sainath, T. N., and Woodland, P. C. (2024). Handling ambiguity in emotion: From out-of-domain detection to distribution estimation. In *Proceedings of the 62st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (ACL 2024).

Chapter 6: Estimating Uncertainty in Emotion Attributes
This chapter extends the approach proposed in Chapter 5 from categorical emotion labels to dimensional emotion attributes. Deep evidential emotion regression (DEER) is proposed to estimate both aleatoric and epistemic uncertainties in emotion attributes. This research work has been presented in the publication:

Wu et al. (2023a): **Wu, W.**, Zhang, C., and Woodland, P. C. (2023). Estimating the uncertainty in emotion attributes using deep evidential regression. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (ACL 2023).

Chapter 7: Subjective Human Evaluations

This chapter expands the scope beyond emotion to general subjective tasks and proposes a generic framework for variability-aware human annotator simulation. The labels it generates can not only capture the mainstream perspective but also align with the distribution of human viewpoints, thus promoting inclusivity and fairness. Part of this research work has been presented in the publication:

Wu et al. (2024a): **Wu, W.**, Chen, W., Zhang, C., and Woodland, P. (2024). Modelling variability in human annotator simulation. In *Findings of the Association for Computational Linguistics: ACL 2024*.

Chapter 8: Detection of Mental Disorders and Cognitive Diseases

This chapter introduces automatic detection of mental disorders and cognitive diseases from spontaneous speech. The chapter starts with relevant background knowledge of automatic diagnosis systems. Then foundation models are investigated for speech-based depression detection, which achieved state-of-the-art performance on the benchmark dataset. A novel Bayesian approach is proposed for confidence estimation for detection of depression and AD, which could improve the reliability of an automatic diagnosis system. The research work in this chapter has been presented in the publications:

Wu et al. (2023c): **Wu, W.**, Zhang, C., and Woodland, P. C. (2023). Self-supervised representations in speech-based depression detection. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing* (ICASSP 2023).

Wu et al. (2024c): **Wu, W.**, Zhang, C., and Woodland, P. C. (2024). Confidence estimation for automatic detection of depression and Alzheimer's disease based on clinical interviews. In *Proceedings of Interspeech 2024*.

Chapter 9: Conclusions and Future Work

The key contributions and outlook are summarised in this chapter.

Appendices A- E include lists of publications and datasets, detailed derivations, and further visualised examples.

Chapter 2

Deep Learning

This chapter introduces the theory and techniques of deep learning which are needed to support the discussion in the rest of the thesis. The chapter is structured into sections discussing the basic architectures of deep neural networks (Section 2.1), typical paradigms of training a deep neural network (Section 2.2), optimisation methods (Section 2.3) and regularisation strategies (Section 2.4).

2.1 Deep Neural Networks

An artificial neural network (ANN) is a model that aims to capture the underlying relationship in a set of data through a process that mimics the way the human brain works. Fig. 2.1 illustrates the structure of a simple ANN which consists of an input layer, an output layer, and two hidden layers. Layers are made up of neurons which is the basic computation unit with weighted combinations and non-linear transformations. If the network contains multiple hidden layers, it is called a deep neural network (DNN). DNNs are universal function approximators. Increasing the depth and width of DNNs allows more complex functions to be approximated (Bishop and Nasrabadi, 2006, Murphy, 2012, Goodfellow et al., 2016). An ANN maps input $\boldsymbol{x}$ to output $\boldsymbol{y}$ through a function parameterised by parameters $\boldsymbol{\theta}$:

$$\boldsymbol{y} = \boldsymbol{f}(\boldsymbol{x}|\boldsymbol{\theta}) \tag{2.1}$$

If the input is mapped to a discrete output space, the mapping is called classification. If the input is mapped to a continuous output space, the mapping is called regression.

The rest of this section summarises commonly used DNN architectures including feed-forward, convolution, recurrent neural networks along with encoder-decoder structure and the attention mechanism.

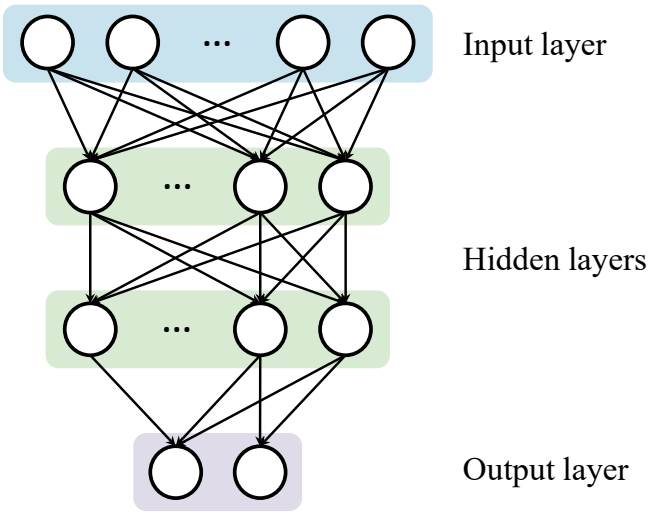


Fig. 2.1 A simple ANN with two hidden layers.

2.1.1 Feed-Forward Neural Networks

Fully-connected (FC) layers are the most fundamental building block of feed-forward neural networks. An FC layer consists of an affine transformation of the input followed by an element-wise non-linear activation function $\sigma(\cdot)$:

$$\begin{aligned} \boldsymbol{z}^{(l)} &= \mathbf{W}^{(l)}\boldsymbol{h}^{(l-1)} + \boldsymbol{b}^{(l)} \\ \boldsymbol{h}^{(l)} &= \sigma\left(\boldsymbol{z}^{(l)}\right) \\ \boldsymbol{\theta}^{(l)} &= \{\mathbf{W}^{(l)}, \boldsymbol{b}^{(l)}\} \end{aligned} \tag{2.2}$$

where l is the layer index, the weight matrix $\mathbf{W}^{(l)} \in \mathbb{R}^{d_{l-1} \times d_l}$ and the bias vector $\boldsymbol{b}^{(l)} \in \mathbb{R}^{d_l}$ are model parameters. Commonly used activation functions include the sigmoid function, the hyperbolic tangent (tanh) function, the rectified linear unit (ReLU) and the leaky ReLU (LReLU):

$$\begin{aligned} \text{sigmoid}(x) &= \frac{1}{1+\exp(-x)} \\ \tanh(x) &= \frac{\exp(x)-\exp(-x)}{\exp(x)+\exp(-x)} \\ \text{ReLU}(x) &= \max(0,\ x) \\ \text{LReLu}(x) &= \max(\alpha x,\ x), \quad 0 < \alpha < 1. \end{aligned}$$

The activation functions are illustrated in Fig. 2.2. The softmax function (defined in Eqn. (2.3) where K is the output dimension) can naturally represent the probability mass function over

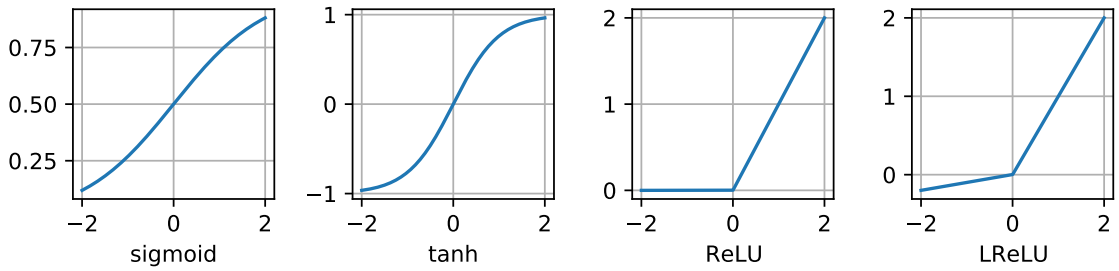


Fig. 2.2 Activation functions.

the outputs and are thus commonly used for classification.

$$\text{softmax}(x)_i = \frac{\exp(\boldsymbol{x}_i)}{\sum_{j=1}^{K} \exp(\boldsymbol{x}_j)} \tag{2.3}$$

For regression, linear output layers are typically used[1].

2.1.2 Convolutional Neural Networks

Feed-forward neural networks capture unstructured vector-to-vector mappings, *i.e.*, a single $\boldsymbol{x}$ maps to a single $\boldsymbol{y}$. In real-life applications, information is usually provided in structured data which resides in a fixed field within a record or file (*e.g.*, audio waveform, sentences, images). Structured data is usually high-dimensional and often in the form of sequences with (unknown) dependencies between data elements (*e.g.*, $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_T$).

Convolutional neural networks (CNNs) (LeCun et al., 1995) are a special type of DNN where the matrix multiplication operation between the input for a layer and the layer weights is replaced by convolution – a specialised kind of linear operation. CNNs are good at capturing the temporal and/or spatial relationships in the data, and are thus widely used for image processing. CNNs are usually made up of convolution blocks which, in general, consecutively perform convolution, non-linear activation and pooling.

In the convolution layer, the weight matrix is replaced by a set of kernels $\mathbf{K}$. The number of kernels C is also called the number of channels. Kernels usually have small receptive field (*e.g.*, 3×3) and are therefore more efficient in terms of memory and computation. The output of each layer is the convolution of each kernel with each input channel, followed by

[1] Details about training models for classification and regression will be discussed in Section 2.2.1.

an element-wise non-linear activation function:

$$\boldsymbol{h}_j^{(l)} = \sigma\left(\sum_{i=1}^{C^{(l-1)}} \boldsymbol{h}_i^{(l-1)} * \mathbf{K}_j^{(l)} + \boldsymbol{b}_j^{(l)}\right), \quad \forall j \in \{1, 2, ..., C^{(l)}\} \tag{2.4}$$

where l is the layer index, $\boldsymbol{b}$ is the bias vector, j is the index of output channel, i is the index of input channel.

Pooling is a form of non-linear down-sampling that replaces the output at a certain location with a summary statistic of the nearby outputs. Max-pooling and mean-pooling are the two most widely used pooling functions (Zhou and Chellappa, 1988). Max-pooling only takes the maximum value within a rectangular area in the output while mean pooling takes the average or a weighted average of the elements within the defined area. The pooling operation effectively reduces the output dimension from each convolutional layer and improves the invariance of the representation against small temporal or spatial translation.

Currently, a wide range of general CNN architectures have shown success in image recognition tasks such as the VGG architecture (Simonyan and Zisserman, 2014), the residual network (ResNet) (He et al., 2016) and DenseNets (Iandola et al., 2014). Although CNNs were first developed for image processing and computer vision, recently they have also been used extensively in speech feature extraction (Baevski et al., 2020, Hsu et al., 2021, Chen et al., 2022).

2.1.3 Recurrent Neural Networks

Recurrent neural networks (RNNs) (Rumelhart et al., 1986, Pineda, 1987) are another type of DNN that operate on sequence data. In contrast to CNNs where the output of a convolution only depends on close neighbours, RNNs take all previous sequence history into consideration. RNNs can model arbitrary sequence-to-vector, vector-to-sequence and sequence-to-sequence mappings:

$$\begin{aligned} \{\boldsymbol{x}_1, ..., \boldsymbol{x}_T\} &\mapsto \boldsymbol{y} \\ \boldsymbol{x} &\mapsto \{\boldsymbol{y}_1, ..., \boldsymbol{y}_T\} \\ \{\boldsymbol{x}_1, ..., \boldsymbol{x}_T\} &\mapsto \{\boldsymbol{y}_1, ..., \boldsymbol{y}_T\}. \end{aligned} \tag{2.5}$$

RNNs have recurrent hidden states $\boldsymbol{h}_t$ that take the output $\boldsymbol{y}_{t-1}$ (Jordan style (Jordan, 1997)) or hidden state $\boldsymbol{h}_{t-1}$ (Elman style (Elman, 1990)) from the previous time step as an additional input. The structure of an Elman RNN is illustrated in Fig. 2.3. The hidden states encapsulate

the information in all past inputs:

$$\begin{aligned} \boldsymbol{h}_t &= \tanh\left(\mathbf{W}^H \boldsymbol{h}_{t-1} + \mathbf{W}^I \boldsymbol{x}_t + \boldsymbol{b}^H\right) \\ \boldsymbol{y}_t &= \tanh\left(\mathbf{W}^O \boldsymbol{h}_t + \boldsymbol{b}^O\right) \end{aligned} \tag{2.6}$$

where t denotes the current time step, $\mathbf{W}^I$ is the input weight matrix, $\mathbf{W}^H$ is the recurrent hidden state transition matrix, $\mathbf{W}^O$ is the output weight matrix, $\boldsymbol{b}^H$ and $\boldsymbol{b}^O$ are the bias vectors.

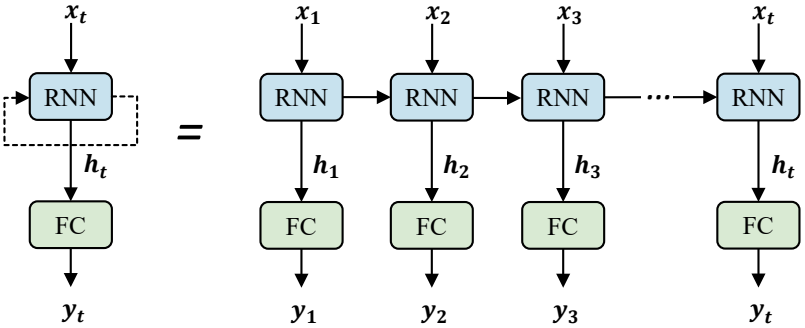


Fig. 2.3 Elman RNN and its unfolded representation.

One major limitation of the traditional RNN structure presented in Eqn. (2.6) is the cascaded forgetting issue for long sequences. Training an RNN is effectively equivalent to unfolding the time steps and training a very deep neural network. As the time step increases, the unfolded neural network gets deeper and the gradient vanishing problem (Bengio et al., 1994) that can happen when training very deep networks (which will be discussed in detail in Section 2.3) blocks the RNN from learning long-term dependencies. To overcome this limitation, the long short-term memory (LSTM) RNN was introduced (Hochreiter and Schmidhuber, 1997) which controls the past information retained by a memory cell.

Long Short-Term Memory

The LSTM introduces an extra cell state $\boldsymbol{c}_t$ and three gates – the input gate $\boldsymbol{g}^I$, forget gate $\boldsymbol{g}^F$ and output gate $\boldsymbol{g}^O$ – to control information flow to and from the memory cell. The gates are functions of the current input $\boldsymbol{x}_t$ and the past output hidden state $\boldsymbol{h}_{t-1}$:

$$\begin{aligned} \boldsymbol{g}_t^I &= \text{sigmoid}\left(\mathbf{U}^I \boldsymbol{h}_{t-1} + \mathbf{W}^I \boldsymbol{x}_t + \boldsymbol{b}^I\right) \\ \boldsymbol{g}_t^F &= \text{sigmoid}\left(\mathbf{U}^F \boldsymbol{h}_{t-1} + \mathbf{W}^F \boldsymbol{x}_t + \boldsymbol{b}^F\right) \\ \boldsymbol{g}_t^O &= \text{sigmoid}\left(\mathbf{U}^O \boldsymbol{h}_{t-1} + \mathbf{W}^O \boldsymbol{x}_t + \boldsymbol{b}^O\right). \end{aligned} \tag{2.7}$$

The candidate state $\tilde{\boldsymbol{c}}_t$ is computed in the same way as in a standard RNN:

$$\tilde{\boldsymbol{c}}_t = \tanh\left(\tilde{\mathbf{U}}\boldsymbol{h}_{t-1} + \tilde{\mathbf{W}}\boldsymbol{x}_t + \tilde{\boldsymbol{b}}\right) \tag{2.8}$$

The forget gate controls what should be carried from the previous cell states to the current cell states. The input gate controls what history should be taken into the current cell states. And the output gate controls what proportion of memory should be stored in the history:

$$\begin{aligned} \boldsymbol{c}_t &= \boldsymbol{g}_t^F \odot \boldsymbol{c}_{t-1} + \boldsymbol{g}_t^I \odot \tilde{\boldsymbol{c}}_t \\ \boldsymbol{h}_t &= \tanh(\boldsymbol{c}_t) \odot \boldsymbol{g}_t^O \end{aligned} \tag{2.9}$$

where $\odot$ denotes element-wise multiplication. The memory cell enables the LSTM to control the information stored in the memory. Compared to traditional RNNs, LSTMs are capable of carrying information over longer time spans, which allows them to capture longer-term dependencies in the data.

Bidirectional LSTM (Bi-LSTM) (Schuster and Paliwal, 1997) networks are an extension of LSTM networks designed to capture information from both past and future states of a sequence. A Bi-LSTM consists of two LSTMs: a forward LSTM that processes the sequence from the beginning to the end, and a backward LSTM that processes the sequence from the end to the beginning. The outputs from both the forward and backward LSTMs are combined, allowing the network to have access to information from both directions. This is beneficial in tasks where causality is not required[2] and the context from both past and future elements of the sequence is important.

2.1.4 Transformers

Models based on the Transformer (Vaswani et al., 2017) architecture are currently the most popular for sequence modelling tasks, and achieve state-of-the-art (SOTA) results in a wide range of applications in natural language processing (NLP) and speech processing. In contrast to RNNs which rely on the recurrent connection to model the inter-dependency across positions in the sequence, the Transformer employs a self-attention mechanism which allows the model to weight the importance of different positions in a sequence. Information at all time lags (near and far) is processed in the same way, allowing better learning of long-term dependencies. In addition, compared to recurrent architectures, the self-attention mechanism processes each position in the sequence in a feed-forward manner, thus enabling

[2] Since Bi-LSTMs use future information, they are not suitable for online applications where only the current and previous inputs are available for predicting outputs at the current step.

a higher level of parallelisation that can take further advantage of the graphics processing unit (GPU) capabilities. In summary, the self-attention mechanism enables the Transformer to capture long-range dependencies and contextual information more effectively and efficiently and are thus suitable for processing moderately long sequences.

The structure of a Transformer is illustrated in Fig. 2.4, which is an attention-based encoder-decoder model. It consists of an encoder which encodes input features into a sequence of high level vector representations and a decoder which generates predictions given the encoder output and previous predictions. The encoder and the decoder are connected by the attention mechanism. In the Transformer, both the encoder and the decoder contain repeated blocks that perform the same set of computations and are referred to as the Transformer encoder block and the Transformer decoder block respectively. The computation in a Transformer block contains scaled dot-product multi-head attention mechanisms, position-wise feed-forward networks (FFNs), layer normalisation and residual connections.

The scaled dot-product attention is the key to the Transformer structure. The input consists of queries and keys of dimension d_k, and values of dimension d_v. The output is computed as the weighted sum of the values, where the weight assigned to each value is computed by a compatibility function of the query with the corresponding key. Packing

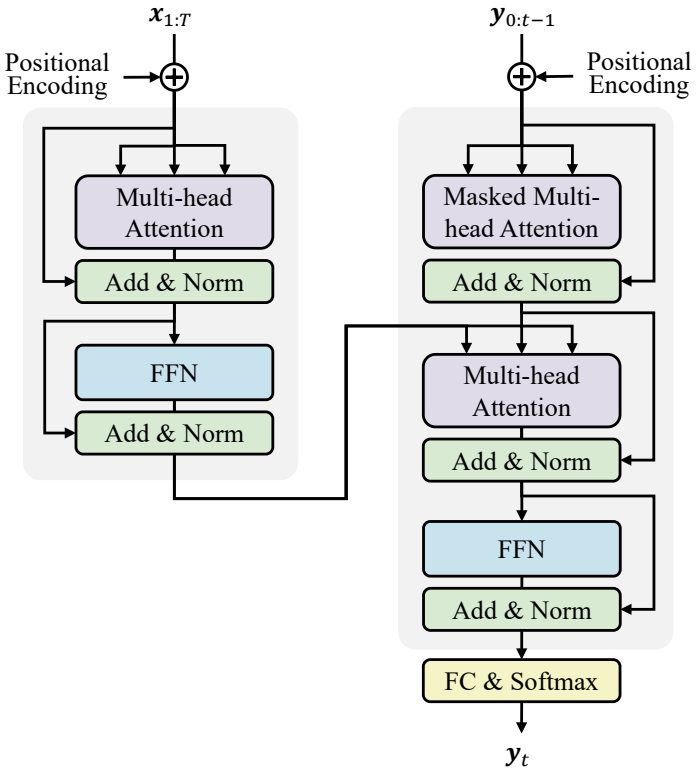


Fig. 2.4 Transformer model structure.

queries, keys, and values into matrices $\mathbf{Q} \in \mathbb{R}^{T \times d_k}, \mathbf{K} \in \mathbb{R}^{T \times d_k}, \mathbf{V} \in \mathbb{R}^{T \times d_v}$ where T is the sequence length, the scaled dot-product attention is computed as:

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}}\right)\mathbf{V} \tag{2.10}$$

The dot product is scaled by $1/\sqrt{d_k}$ to avoid pushing the softmax function into saturation regions where it has extremely small gradients.

Furthermore, the Transformer uses multi-head attention (MHA) which allows the model to jointly attend to information from different representation subspaces at different positions. Instead of performing a single attention function with d_{model}-dimensional keys, values and queries, MHA projects the queries, keys and values H times with different learned linear projections to d_k, d_k and d_v dimensions respectively:

$$\begin{aligned} \mathbf{Q}_i &= \mathbf{W}_i^Q \mathbf{Q}, \quad & \mathbf{W} &\in \mathbb{R}^{d_{\text{model}} \times d_k} \\ \mathbf{K}_i &= \mathbf{W}_i^K \mathbf{K}, \quad & \mathbf{K} &\in \mathbb{R}^{d_{\text{model}} \times d_k} \\ \mathbf{V}_i &= \mathbf{W}_i^V \mathbf{V}, \quad & \mathbf{V} &\in \mathbb{R}^{d_{\text{model}} \times d_v} \end{aligned} \tag{2.11}$$

where $i = 1, \ldots, H$ and H is referred to as the number of heads. The attention function is then performed in parallel for each of the projected version of queries, keys, and values, each producing a d_v-dimensional output values. These are then concatenated and projected once again resulting in the final values:

$$\begin{aligned} \text{MultiHead}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) &= \text{Concat}(\text{head}_1, ..., \text{head}_i)\mathbf{W}^O \\ \text{where head}_i &= \text{Attention}(\mathbf{Q}_i, \mathbf{K}_i, \mathbf{V}_i) \end{aligned} \tag{2.12}$$

where $\mathbf{W}^O \in \mathbb{R}^{Hd_v \times d_{\text{model}}}$ is a parameter matrix. In Transformer encoder blocks, all of the queries, keys, and values come from the output of the previous layer in the encoder, which is called self-attention. The Transformer decoder block contains two MHA modules. The first one is a masked self-attention MHA where an attention mask is applied to the dot product to prevent positions from attending to subsequent positions[3]. The second is an encoder-decoder cross attention where the queries come from the previous decoder layer, and the keys and values come from the output of the encoder.

In addition to the attention sub-layer, each Transformer block contains a fully-connected feed-forward network (FFN), which is applied to each position separately and identically. This consists of two linear transformations with a ReLU activation in between. A residual

[3]This mask ensures that the prediction for a given position only depends on the previous positions in the sequence and the generation process is thus causal (or auto-regressive).

connection (He et al., 2016) is applied around each sub-layer followed by layer normalisation (Ba et al., 2016): LayerNorm($\boldsymbol{x}$ + Sublayer($\boldsymbol{x}$)), which improves training stability and convergence.

Since Transformer blocks do not contain recurrence and process each position identically and in parallel, the order of the sequence is not explicitly maintained. Therefore, additional positional embeddings are added to the input before sending them to the Transformer blocks to inject some information about the relative or absolute position of the tokens in the sequence. The positional encoding has the same dimension d_{model} as the input embeddings. Positional encoding can either be fixed or learned (Gehring et al., 2017). A commonly used fixed form employs sinusoidal functions where each element in the embedding vector at position pos, dimension i is given by:

$$\begin{aligned} \text{PE}(pos, 2i) &= \sin\left(\frac{pos}{10000^{\frac{2i}{d_{\text{model}}}}}\right) \\ \text{PE}(pos, 2i+1) &= \cos\left(\frac{pos}{10000^{\frac{2i}{d_{\text{model}}}}}\right) \end{aligned} \tag{2.13}$$

The positional embedding is directly added to the input vector at position pos and each dimension of the positional encoding corresponds to a sinusoid.

Although initially proposed for text data, Transformers and their variants also show impressive performance for audio and image data and are now also widely used for speech processing (Baevski et al., 2020, Chen et al., 2022, Radford et al., 2023) and computer vision (Parmar et al., 2018, Chen et al., 2020, Kolesnikov et al., 2021).

2.1.5 Conformers

Transformer models are good at capturing content-based global interactions, while CNNs exploit local features effectively. The convolution-augmented transformer, or Conformer (Gulati et al., 2020), has been proposed to take advantage of both. By integrating convolutional layers with self-attention mechanisms, Conformers allow modelling both local and global dependencies of an audio sequence in a parameter-efficient way.

The overall structure of a Conformer block is illustrated in Fig. 2.5 (a). The two main modifications to the standard Transformer in the Conformer are the addition of the convolution module (Fig. 2.5 (b)) and the two half-step FFNs (Fig. 2.5 (c)) which replace the original single FFN in the Transformer block. The convolution module begins with a gating mechanism (Dauphin et al., 2017) which comprises of a pointwise convolution and a gated linear unit (GLU). Pointwise convolution applies convolution kernels of $C_{\text{out}} \times 1$ to the input

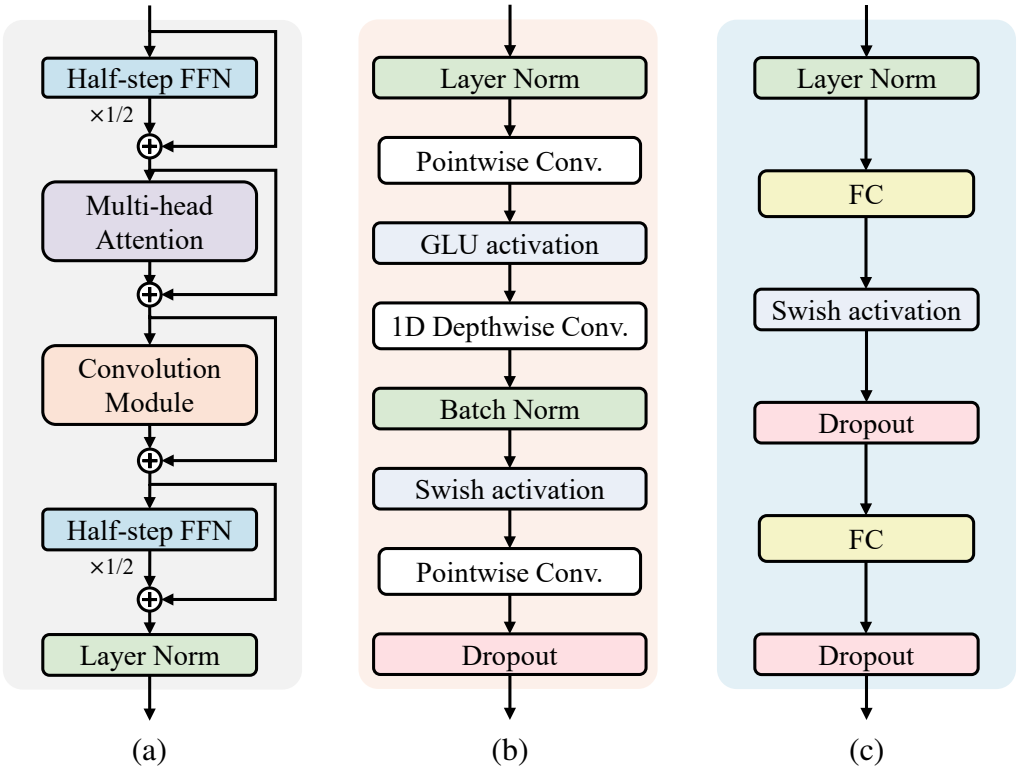


Fig. 2.5 The Conformer structure. (a) The overall structure of a conformer block (b) Structure of the convolution module. (c) Structure of the half-step feed-forward network (FFN).

of shape $C_{\text{in}} \times T$ where C_{in} is the number of input channels and C_{out} is the number of output channels. The first pointwise convolution has $C_{\text{out}} = 2C_{\text{in}}$ and the GLU after it uses the second half (*i.e.*, channels C_{in} to $2C_{\text{in}}$) for element-wise gating activation on the first half (*i.e.*, channels 1 to C_{in}). The depthwise convolution performs convolution along the time dimension and the second pointwise convolution has $C_{\text{out}} = C_{\text{in}}$. Both the convolution and the half-step FFN modules use the Swish activation function (Ramachandran et al., 2017):

$$\text{Swish}(\boldsymbol{x}) = \boldsymbol{x} \cdot \text{sigmoid}(\boldsymbol{x}) \tag{2.14}$$

The Swish activation function is often regarded as a smoothed version of the ReLU function, which is differentiable everywhere.

Another important difference from the standard Transformer block described in Section 2.1.4 is the use of pre-norm residual units (Wang et al., 2019, Nguyen and Salazar, 2019) that apply layer normalisation within the residual unit and on the input before the MHA or FFN layer. In addition, relative sinusoidal positional encoding (Dai et al., 2019) is used instead of fixed sinusoidal positional encoding which allows the self-attention module to generalise better for different input lengths resulting in enhanced robustness to the variations

in the utterance length. Detailed descriptions of batch/layer normalisation and dropout can be found in Sections 2.3.5 and 2.4.3 respectively.

2.2 Training Neural Networks

Having discussed the main types of networks and structures, this section presents the procedure to train a network for a specific task. The aim of training is to find the optimal value of the model parameters that maps the input to the output. A loss function (also known as a cost function or error function) $\mathcal{L}(\boldsymbol{\theta})$ is defined which measures how well the model captures the relationship in the data and is a function of the model parameters $\boldsymbol{\theta}$. The optimal model parameters are found by minimising the loss function:

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}). \tag{2.15}$$

2.2.1 Supervised Learning

Tasks where the training data contains inputs paired with their corresponding target outputs are called supervised learning problems. Classification and regression are two most common supervised learning problems. For classification, the desired output contains a finite number of discrete categories while for regression the desired output consists of one ore more continuous variables. If the training data consists of a set of inputs without any corresponding targets, the tasks are called unsupervised learning problems. The goal of such tasks includes grouping similar samples within the data, known as clustering, recognising the distribution of input data, known as density estimation, and projecting data from a high-dimensional space to a low-dimensional space, known as dimensionality reduction. This thesis mainly focuses on supervised learning problems.

Training Models for Classification

Consider a training dataset $\mathcal{D} = \{\boldsymbol{x}_i, \mathbf{t}_i\}_{i=1}^{N}$ where $\boldsymbol{x}_i$ is an input sample, $\mathbf{t}_i$ is its corresponding target and N is the number of samples in the training set. For a classification problem, $\mathbf{t}_i$ is a one-hot vector with the dimension corresponding to the correct class set to 1 and others being 0. The output $\boldsymbol{y}_i$ is a categorical probability distribution over the K classes which is usually the output of a softmax function. Cross-entropy is commonly used as the loss function for classification problems:

$$\mathcal{L}(\boldsymbol{y}_i, \mathbf{t}_i) = -\sum_{k=1}^{K} \mathbf{t}_{ik} \log \boldsymbol{y}_i. \tag{2.16}$$

Training Models for Regression

In a regression problem, the target $\mathbf{t}_i$ is a continuous-value vector (or a scalar if output dimension is one). The loss function is usually designed to minimise the error between network output $\boldsymbol{y}_i$ and the target $\mathbf{t}_i$. Commonly used error functions include the mean squared error (MSE) defined in Eqn. (2.17) and mean absolute error (MAE) defined in Eqn. (2.18). MSE, also known as the $L2$ loss, is the most widely used loss function for regression though it tends to be sensitive to outliers. MAE, also known as $L1$ loss, is sometimes employed as an alternative to the MSE when the dataset contains a large number of outliers.

$$\mathcal{L}_{\text{MSE}} = \sum_{i=1}^{N} (\mathbf{t}_i - \boldsymbol{y}_i)^2 \tag{2.17}$$

$$\mathcal{L}_{\text{MAE}} = \sum_{i=1}^{N} ||\mathbf{t}_i - \boldsymbol{y}_i|| \tag{2.18}$$

2.2.2 Self-Supervised Learning and Foundation Models

Self-supervised learning (SSL) is a special type of paradigm which utilises information extracted from the input data itself as the label to learn representations without explicit supervision. Unlike standard supervised learning problems, SSL does not rely on external labels provided by humans while techniques for supervised learning can often be used for SSL (*i.e.*, cross-entropy loss). A major benefit of SSL is the use of unlabelled data. Obtaining labels are costly and time-consuming, SSL leverages the abundance of unlabelled data that is often readily available. By training the model to predict some aspect of the input data based on other parts of the input data, the model can learn useful representations of the input data without explicit labels.

Recently, a paradigm shift has been observed with the rise of foundation models (Bommasani et al., 2021). Foundation models are any models that are trained on broad range of data at scale and can be adapted (*e.g.*, finetuned) to a wide range of downstream tasks (Bommasani et al., 2021). SSL is one of the most common approaches to training a foundation model to learn representations useful for downstream tasks. The paradigm consists of two phases. In the first phase, a foundation model (also called an upstream model) is pretrained using SSL. In the second phase, a downstream model is trained in a supervised manner either using the learned representation from the frozen model or by finetuning the pretrained model.

Foundation models have achieved great success in natural language processing (*e.g.*, BERT (Devlin et al., 2019), GPT-3 (Brown et al., 2020)) and vision (*e.g.*, ViT (Kolesnikov et al., 2021), iGPT (Chen et al., 2020)) and has attracted increasing attention in speech pro-

cessing (*e.g.*, wav2vec 2.0 (Baevski et al., 2020), HuBERT (Hsu et al., 2021), WavLM (Chen et al., 2022)). Foundation models leverage a large amount of unlabelled data during pretraining, which can help alleviate the data sparsity issue in emotion and medical data.

In SSL, networks are trained to map the input to the desired representation by solving a pretext task. SSL models can be broadly grouped into three categories based on their pretext task type: generative, contrastive, and predictive SSL.

Generative SSL

In this category, the pretext task is to generate, or reconstruct, the input data based on some limited view. This includes predicting future inputs from past inputs, masked from unmasked, or the original from some other corrupted version. Examples of generative SSL includes autoencoders (Hinton and Zemel, 1993) which are trained to reconstruct the given input and BERT (Devlin et al., 2019) which can be trained by masked reconstruction and next sentence prediction.

Contrastive SSL

Contrastive models learn self-supervised representations by distinguishing a target sample (positive) from distractor samples (negatives) given an anchor representation. The pretext task minimises distances in a latent space between the anchor and positive samples while maximising the distance to negative samples. Typical contrastive models include contrastive predictive coding (Oord et al., 2018), Wav2vec (Schneider et al., 2019), Wav2vec 2.0 (Baevski et al., 2020).

Predictive SSL

Unlike generative representation learning approaches, predictive methods output a distribution over a discrete vocabulary given a masked input utterance. The predictive loss is only applied over the masked regions, forcing the model to learn good high-level representations of unmasked inputs to infer the targets of masked ones correctly. Typical predictive models include DiscreteBERT (Baevski and Mohamed, 2020), HuBERT (Hsu et al., 2021), WavLM (Chen et al., 2022), and Data2vec (Baevski et al., 2022).

2.2.3 Language Foundation Models

Foundation models used in this thesis are grouped in to language foundation models and speech foundation models depending on the input modality. This section describes language foundation models.

BERT

BERT (Bidirectional Encoder Representations from Transformers) (Devlin et al., 2019) is a popular technique for pretraining contextualised universal sentence embeddings based on a Transformer model. BERT enables bi-directional prediction and sentence-level understanding by obtaining both previous and subsequent context. BERT is trained with two tasks: masked language modelling which predicts the missing word given the context and next sentence prediction for understanding the relationships between sentences and telling whether one sentence is the following sentence of the other. The original English-language BERT model has two versions: BERT-Base which contains 12 Transformer encoders with 12 bidirectional self-attention heads, and BERT-Large which contains 24 encoders with 16 bidirectional self-attention heads. Both models are pretrained from unlabelled data extracted from the BooksCorpus (Zhu et al., 2015) (800M words) and English Wikipedia (2,500M words).

RoBERTa

RoBERTa (Robustly optimised BERT approach) (Liu et al., 2019) is a variant of the BERT model. Built upon the architecture and pretraining methodology of BERT, RoBERTa incorporates several modifications and enhancements: (i) dynamic masking where different masking patterns are applied to the input data in each training epoch, which helps the model generalise better to unseen data and prevent overfitting; (ii) larger training corpus with additional web data and books and a longer training period; (iii) the removal of the next sentence prediction task. Overall, RoBERTa is designed to produce more robust representations of text, leading to improved performance on various NLP tasks such as text classification, sentiment analysis, question answering, and natural language understanding.

2.2.4 Speech Foundation Models

This section introduces the speech foundation models used in this thesis.

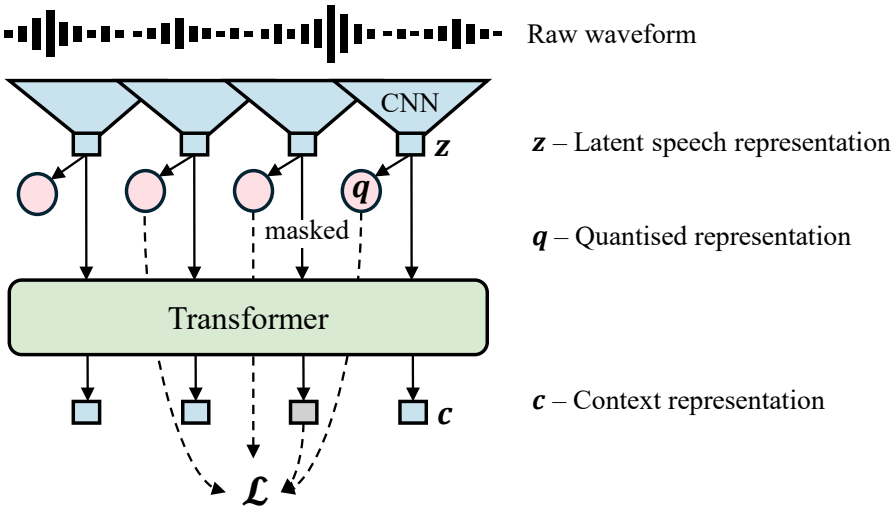


Fig. 2.6 Illustration of Wav2vec 2.0 framework.

Wav2vec 2.0

The Wav2vec 2.0 (W2V2) (Baevski et al., 2020) model combines masking and contrastive learning. The structure of W2V2 is illustrated in Fig. 2.6. It takes as input a waveform and encodes speech audio via a CNN and then masks spans of the resulting latent speech representations. The latent representations are then fed into a Transformer-based context network to generate contextualised representations. The model is trained via a contrastive pretext task where the true latent representation is to be distinguished from distractors. Given the context network output $\boldsymbol{c}_t$ centred over masked time step t, the model is trained to identify the true quantised latent speech representation $\boldsymbol{q}_t$ in a set of $K + 1$ quantised candidate representations $\mathbf{Q}_t$ which includes $\boldsymbol{q}_t$ and K distractors. Distractors are uniformly sampled from other masked time steps of the same utterance. The loss is defined as

$$\mathcal{L}_m = -\log \frac{\exp\left(\text{sim}\left(\boldsymbol{c}_t, \boldsymbol{q}_t\right)/K\right)}{\sum_{\tilde{\boldsymbol{q}} \in \mathbf{Q}_t} \exp\left(\text{sim}\left(\boldsymbol{c}_t, \tilde{\boldsymbol{q}}\right)/K\right)} \tag{2.19}$$

where $\text{sim}\left(\boldsymbol{c}_t, \boldsymbol{q}_t\right)$ computes the cosine similarity between context representations and quantised latent speech representations. The W2V2 models are primarily pretrained on audio data from the LibriSpeech corpus (Panayotov et al., 2015) (960 hours) or LibriVox (Kearns, 2014) (60k hours)[4]. Both datasets are derived from audio books.

[4]Different versions of W2V2 models may have been pretrained on different datasets.

HuBERT

Rather than relying on an advanced representation learning model for discretising continuous spoken inputs, HuBERT (Hidden unit BERT) (Hsu et al., 2021) examines the effectiveness of using the classic k-means units trained on MFCC features. HuBERT predicts the predetermined k-means cluster assignment given the masked continuous speech features. Similar to the Wav2vec 2.0, the HuBERT model consumes continuous waveform inputs using a convolutional encoder. Masking is then applied before the Transformer network as in Wav2vec 2.0. Since the targets are pre-computed cluster identities, the HuBERT model can directly evaluate the regular cross-entropy loss between the correct k-means cluster and the predicted one. This is opposed to constrastive methods which require negative samples to avoid degenerating to trivial solutions. The HuBERT models are pretrained on the LibriSpeech corpus (Panayotov et al., 2015) (960 hours) or Libri-Light (60k hours).

WavLM

Based on the HuBERT framework, WavLM emphasises spoken content modelling and speaker identity preservation (Chen et al., 2022). It extends the Transformer self-attention with a content-based gated relative position bias which improves the model's capability on recognition tasks compared to the convolutional feature extractor used in HuBERT and Wav2vec 2.0 models. The WavLM framework also proposed an utterance mixing strategy where partially overlapped signals from different speakers are constructed to augment the training data. The content information corresponding to the main speaker is used as the target during the masked prediction pretraining. This helps the model to learn paralinguistic information related to, *e.g.*, speaker attributes. The pretraining data used for WavLM extends the sources used for HuBERT and Wav2vec 2.0 to reach a total of 94k hours of public audio. Due to the use of multi-speaker data, WavLM shows superior performance on multi-speaker tasks such as separation and diarisation.

2.3 Optimisation

Given the loss $\mathcal{L}(\boldsymbol{\theta})$, the neural network is trained to find the set of parameters $\boldsymbol{\theta}$ that minimise the loss. Closed-form solutions are not available for deep neural networks. Instead, gradient-based methods (*e.g.* gradient descent (Gill et al., 1981)) are commonly used to optimise the network iteratively and error back-propagation (Rumelhart et al., 1986) is widely used to computed the gradients.

2.3.1 Error Back-Propagation

Back-propagation computes the gradient of the loss function with respect to each model parameter. Gradient-based optimisation algorithms then takes these gradients to update the value of the parameters. To compute the gradient, the loss is propagated from the output back through the network using the chain rule of partial differentiation. Taking the network in Eqn. (2.2) as an example, the gradient of loss $\mathcal{L}(\boldsymbol{\theta})$ with respect to $\mathbf{W}^{(l)}$ can be computed as:

$$\frac{\partial \mathcal{L}(\boldsymbol{\theta})}{\partial \mathbf{W}^{(l)}} = \frac{\partial \mathcal{L}(\boldsymbol{\theta})}{\partial \boldsymbol{z}^{(l)}} \frac{\partial \boldsymbol{z}^{(l)}}{\partial \mathbf{W}^{(l)}} = \frac{\partial \mathcal{L}(\boldsymbol{\theta})}{\partial \boldsymbol{z}^{(l)}} \boldsymbol{h}^{(l-1)} \tag{2.20}$$

where $\frac{\partial \mathcal{L}(\boldsymbol{\theta})}{\partial \boldsymbol{z}^{(l)}}$ is the derivative of the activation function. If the softmax function is used at the output layer and the loss function is cross entropy, then

$$\frac{\partial \mathcal{L}(\boldsymbol{\theta})}{\partial \boldsymbol{z}^{(o)}} = \mathbf{t} - \boldsymbol{z}^{(o)} \tag{2.21}$$

where $\mathbf{t}$ is the target and $\boldsymbol{z}^{(o)}$ is the logits at the output before passing through the softmax. If the sigmoid activation is used, the derivative $\frac{\partial \mathcal{L}(\boldsymbol{\theta})}{\partial \boldsymbol{z}^{(l)}}$ is less than $\frac{1}{4}$. As the network gets deeper, the chain expands and the gradient decreases, causing parameters closer to the input to not be updated efficiently. This problem is called gradient vanishing, which can be mitigated by replacing the sigmoid activation function with ReLU, adding residual connections, applying normalisation methods[5], *etc*.

2.3.2 Parameter Update

Stochastic Gradient Descent

Having computed the gradient, gradient descent is commonly used to update the model parameters iteratively. In each iteration, gradient descent methods update each parameter in the opposite direction to the gradient of the loss with respect to the parameter by a small step:

$$\Delta \boldsymbol{\theta}^{(t-1)} = -\epsilon \nabla_{\theta} \mathcal{L}\left(\boldsymbol{\theta}^{(t-1)}\right) \tag{2.22}$$

$$\boldsymbol{\theta}^{(t)} = \boldsymbol{\theta}^{(t-1)} + \Delta \boldsymbol{\theta}^{(t-1)} \tag{2.23}$$

where t is the index of the update step and ϵ controls the step size, which is called the learning rate.

[5]Normalisation methods will be discussed in detail in Section 2.3.5.

When the loss $\mathcal{L}(\boldsymbol{\theta})$ is computed over the entire training set in each iteration, the optimisation is call full batch gradient descent (BGD). A full pass over the entire training set is called an epoch. For BGD, each iteration is one epoch. BGD computes the exact gradient of the loss on the training data, but may take a long time to update the parameter once, especially for large datasets. Instead of updating the parameters after a full pass over the training set, an alternative method is to update the parameters after processing each data point, known as stochastic gradient descent (SGD) (Robbins and Monro, 1951). SGD can yield faster convergence than BGD, however, in SGD the gradient of loss at each data point is a single-sample approximation to the gradient of loss over the entire dataset. This leads to noise in gradient update and a far lower learning rate is usually necessary to avoid instability during training. A compromise between BGD and SGD is stochastic mini-batch gradient descent (mini-batch SGD)[6], which computes the loss over a mini-batch of samples. The mini-batches are usually shuffled between each epoch to avoid unnecessary bias into the model. Using mini-batches can also exploit the parallel processing ability of GPU where multiple mini-batches can be processed in parallel and thus achieves efficient training.

2.3.3 Momentum and Adaptive Learning Rates

The choice of the learning rate parameter ϵ is important when applying gradient descent in practice. It is typical for the local gradient vector on the error surface not to point directly towards the overall loss function minimum (*e.g.*, when the error surface forms a "valley"). A large step size ϵ can lead to oscillations back and forth across the valley, with these oscillations becoming divergent if ϵ is excessively large. Since ϵ must be kept sufficiently small to avoid divergent oscillations, the convergence becomes very slow and the optimisation is inefficient.

Momentum methods are typically used to avoid this behaviour and accelerate convergence by adding the previous parameter update, multiplied by the momentum rate β to the current parameter update:

$$\Delta\boldsymbol{\theta}^{(t-1)} = -\epsilon\nabla_{\boldsymbol{\theta}}\mathcal{L}\left(\boldsymbol{\theta}^{(t-1)}\right) + \beta\Delta\boldsymbol{\theta}^{(t-2)} \tag{2.24}$$

where $0 \leq \beta \leq 1$. The model parameters are then updated using Eqn. (2.23). The momentum effectively adds inertia to the motion through weight space and smooths out the oscillations, which increases the effective learning rate in the direction of consistent gradients and accelerates convergence.

[6] Mini-batch SGD is commonly used in practice and SGD sometimes refers to mini-batch SGD instead of "true" SGD in the literature.

Since the optimal learning rate depends on the local curvature of the error surface and can vary according to the direction in parameter space, it is desirable to use different learning rates for each parameter in the network. Adagrad (Duchi et al., 2011), AdaDelta (Zeiler, 2012), and RMSprop (Goodfellow et al., 2016) are classes of optimisers that automatically scale the learning rate for each parameter using accumulated gradients.

Adam (Kingma and Ba, 2014) is a widely adopted optimisation algorithm for neural networks, which combines momentum with automatic adjustment of learning rate for each parameter. Adam stores the momentum for each parameter separately using update equations that consist of exponentially weighted moving averages for both the gradients and the squared gradients:

$$s_i^{(t)} = \beta_1 s_i^{(t-1)} + (1-\beta_1)\left(\frac{\partial \mathcal{L}(\boldsymbol{\theta}^{(t-1)})}{\partial \theta_i}\right) \tag{2.25}$$

$$r_i^{(t)} = \beta_2 r_i^{(t-1)} + (1-\beta_2)\left(\frac{\partial \mathcal{L}(\boldsymbol{\theta}^{(t-1)})}{\partial \theta_i}\right)^2 \tag{2.26}$$

$$\hat{s_i}^{(t)} = \frac{s_i^{(t)}}{1-\beta_1^t} \tag{2.27}$$

$$\hat{r_i}^{(t)} = \frac{r_i^{(t)}}{1-\beta_2^t} \tag{2.28}$$

$$\theta_i^{(t)} = \theta_i^{(t-1)} - \epsilon \frac{\hat{s_i}^{(t)}}{\sqrt{\hat{r_i}^{(t)}} + \delta} \tag{2.29}$$

where i is the parameter index and δ is a small number (*e.g.*, 10^{-8}) for computational stability. Typical values for β_1 and β_2 is 0.9 and 0.99 respectively.

2.3.4 Learning Rate Schedules

Apart from adaptive learning rates which automatically apply different scaling coefficients to ϵ for different parameters, it is also common to apply a global learning rate scheduler which gradually changes the value of ϵ as training proceeds. The learning rate ϵ can be set to gradually ramp up at the beginning (also called warm-up period) and/or decay at a certain ratio associated with the number of training steps. Some schedulers decrease the learning rate based on the validation performance of the current model on a held-out set of data[7] during training.

[7] It is common to hold out a separate subset of the training set for validation purposes.

2.3.5 Initialisation and Normalisation

Iterative algorithms such as gradient descent require a choice for the initial values of the parameters being learned. The model performance can sometimes be sensitive to the initial parameter values, which determines the speed and location of convergence as well as the model's generalisation ability. Symmetry breaking is key to parameter initialisation which avoids redundancy where units computes the same function and update synchronously. Symmetry breaking is usually realised by initialising parameters randomly from some distribution, *e.g.*, uniform distribution or zero-mean Gaussian distribution. Another important type of technique for initialising the parameters of a neural network is by using the values from another network trained on a different task or exploit various forms of unsupervised training (*e.g.*, the upstream-downstream paradigm discussed in Section 2.2.2). These techniques fall into the broad class of transfer learning.

Normalisation is an important strategy that improves training stability and facilitates convergence by avoiding extremely large or extremely small values. Three types of normalisation are commonly used where normalisation is applied across input data, mini-batches, and layers, respectively.

Data Normalisation

It is common for datasets to contain different input variables that span very different ranges. Changes in a larger value tends to yield much larger changes in the output and thus the loss function. This would lead to an error surface with very different curvatures along different axes which makes optimisation more challenging. Min-max normalisation and z-score normalisation are two commonly used rescaling methods. Min-max normalisation is defined as follows:

$$\boldsymbol{x}_{ni}^{\text{norm}} = \frac{\boldsymbol{x}_{ni} - \boldsymbol{x}_i^{\min}}{\boldsymbol{x}_i^{\max} - \boldsymbol{x}_i^{\min}}, \quad \text{where } \boldsymbol{x}_i^{\min} = \min\{\boldsymbol{x}_{ni}\}_{n=1}^{N},\ \boldsymbol{x}_i^{\max} = \max\{\boldsymbol{x}_{ni}\}_{n=1}^{N} \tag{2.30}$$

where i is the feature index, N is the number of data samples. Min-max normalisation may not be suitable for data with outliers, as it can compress the majority of the data into a narrow range. In such cases, z-score normalisation (also called standardisation) may be more appropriate which subtracts feature mean from the original value and then divides the result by the standard deviation of the feature:

$$\boldsymbol{x}_{ni}^{\text{norm}} = \frac{\boldsymbol{x}_{ni} - \mu_i}{\sqrt{\boldsymbol{\sigma}_i^2 + \delta}}, \quad \text{where } \mu_i = \frac{1}{N}\sum_{n=1}^{N} \boldsymbol{x}_{ni},\ \boldsymbol{\sigma}_i^2 = \frac{1}{N}\sum_{n=1}^{N} (\boldsymbol{x}_{ni} - \mu_i)^2 \tag{2.31}$$

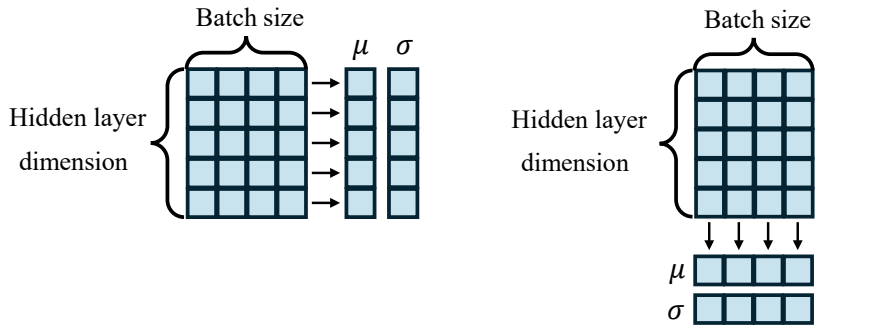
where δ is a small constant to avoid numerical issues in situations when $\boldsymbol{\sigma}_i^2$ is small. This ensures that the transformed data has a mean of 0 and a standard deviation of 1, making it easier to interpret and compare features.

Batch Normalisation

For variables in hidden layers of a deep network, normalising them to zero mean and unit variance helps prevent saturation in activation functions caused by extreme values and alleviates gradient vanishing or exploding. Batch normalisation is illustrated in Fig. 2.7(a) where the mean and variance are computed across the mini-batch separately for each hidden unit. Eqn. (2.31) is modified for batch normalisation as follows:

$$\boldsymbol{y}_{ni}^{\text{BatchNorm}} = \frac{\boldsymbol{y}_{ni} - \boldsymbol{\mu}_i}{\sqrt{\boldsymbol{\sigma}_i^2 + \delta}}, \quad \text{where } \boldsymbol{\mu}_i = \frac{1}{K}\sum_{n=1}^{K} \boldsymbol{y}_{ni},\ \boldsymbol{\sigma}_i^2 = \frac{1}{K}\sum_{n=1}^{K}(\boldsymbol{y}_{ni} - \boldsymbol{\mu}_i)^2 \tag{2.32}$$

where K is the batch size.



(a) Batch normalisation (b) Layer normalisation

Fig. 2.7 Illustration of (a) batch normalisation where the mean and variance are computed along the dimension of batch size and (b) layer normalisation where the mean and variance are computed along the dimension of hidden states.

Layer Normalisation

Batch normalisation is infeasible for RNNs where the distributions change after each time step. In addition, the estimates of mean and variance can be very noisy if the batch size is very small. It can also be inefficient if the mini-batches are split across different GPUs which can be common when training on very large datasets. An alternative is layer normalisation, which normalises across the hidden-unit values. As shown in Fig. 2.7(b), layer normalisation

computes the mean and variance across the hidden units separately for each data point. Layer normalisation is defined as follows:

$$\boldsymbol{y}_{ni}^{\text{LayerNorm}} = \frac{\boldsymbol{y}_{ni} - \boldsymbol{\mu}_n}{\sqrt{\boldsymbol{\sigma}_n^2 + \delta}}, \quad \text{where } \boldsymbol{\mu}_n = \frac{1}{D}\sum_{i=1}^{D} \boldsymbol{y}_{ni},\ \boldsymbol{\sigma}_n^2 = \frac{1}{D}\sum_{i=1}^{D}(\boldsymbol{y}_{ni} - \boldsymbol{\mu}_n)^2 \tag{2.33}$$

where D is the dimension of the hidden layer.

2.4 Regularisation

Overfitting is a common problem in deep learning where the model learns to fit the training data but fails to generalise to unseen data (Goodfellow et al., 2016). When the model is overfitting the training data, a common indication is that the training loss keeps decreasing while the validation loss plateaus before rising. Regularisation methods are developed to prevent the model from overfitting to the training target. This section introduces several techniques that are commonly used to help address overfitting.

2.4.1 Weight Decay

Weight decay (also known as parameter norm penalty) constrains the capability of the model by introducing an extra term to the optimisation objective that penalises larger weight values:

$$\tilde{\mathcal{L}}(\boldsymbol{\theta}) = \mathcal{L}(\boldsymbol{\theta}) + \lambda \parallel \boldsymbol{\theta} \parallel_p \tag{2.34}$$

where λ is a non-negative hyperparameter to scale the norm regularisation and $\parallel \boldsymbol{\theta} \parallel_p$ is the p-norm of all parameters. The $L2$ norm is commonly used on weights for DNNs. Another form of norm regularisation is $L1$, which encourages the weights to be sparse (Goodfellow et al., 2016) and can be useful in dimension reduction and feature selection.

2.4.2 Early Stopping

Early stopping is a simple but commonly used training technique in deep learning which stops the training process before complete convergence based on some specific criterion (*e.g.*, performance on validation set). It has been shown that for a linear model with a quadratic loss function optimised by a simple gradient descent algorithm, early stopping is equivalent to $L2$ norm regularisation (Bishop, 1995).

2.4.3 Ensembles and Dropout

Generalisation can often also be improved by averaging the predictions from several different models trained to solve the same problem. Such combinations of models are usually called ensembles (Lakshminarayanan et al., 2017). Since different models tend to make different errors, ensemble methods can normally outperform individual models. However, it can be more computationally expensive as multiple models need to be trained separately.

Dropout (Srivastava et al., 2014) is an effective and computationally efficient alternative method which provides strong regularisation by randomly disabling neurons in a DNN during training. By applying different dropout masks to different layers of the network for each mini-batch, Dropout can be considered an approximation to simultaneously training an exponentially large number of neural networks with partially shared parameters. The proportion of neurons to be dropped for an iteration is controlled by a hyperparameter p_{dropout}, which is sometimes called the dropout rate. Dropout prevents some neurons from being over-specialised for certain data. During testing, predictions can in principle be made by averaging over the space of all dropout masks which is intractable. Instead, it can be approximated by sampling a small number of masks which is known as Monte Carlo dropout (Gal and Ghahramani, 2016), or re-scaling the weights in the network with no nodes masked out by $1 - p_{\text{dropout}}$, which is widely used. Both empirically perform well in practice.

2.4.4 Data Augmentation

Overfitting can happen when the model becomes too complex and fits the training data too closely. It learns to capture noise or random fluctuations in the training data rather than the underlying patterns or relationships so that it fails to generalise to unseen data. Data augmentation is a widely used approach to increase the amount of training data by generating new data from the existing data. By artificially increasing the diversity and size of the training dataset, the model can learn to generalise better and is less likely to overfit to the specific patterns present in the original training data. The addition of noise is an augmentation method applicable for various types of input data where some small random values are added to or subtracted from the original data. For image data, it is common to augment also by transformations such as cropping, scaling, translating, and rotating (Krizhevsky et al., 2012). For speech data, vocal tract length perturbation and speed perturbation are commonly used (Jaitly and Hinton, 2013, Ko et al., 2015). SpecAugment (Park et al., 2019) has been widely used for automatic speech recognition which augments the input spectrogram by applying multiple instances of time warping, frequency masking and time masking. By

introducing randomly corrupted input, this method prevents the model from overfitting specific features and enhances its ability to generalise to different acoustic conditions.

2.5 Chapter Summary

This chapter introduces the background of deep learning. In Section 2.1, several basic types of deep neural networks are discussed including feed-forward neural networks, convolutional neural networks, recurrent neural networks, Transformers and Conformers. Section 2.2 introduces common paradigms for training a neural network including standard supervised learning for classification and regression tasks, self-supervised learning, and various types of foundation models. Section 2.3 introduces the optimisation algorithm commonly used to learn the parameters of a deep neural network. Regularisation strategies to prevent overfitting are discussed in Section 2.4. Many terms will be often referred to throughout the thesis.

Chapter 3

Automatic Emotion Recognition

Emotion plays a pivotal role in human cognition and behaviour, influencing interpersonal relationships, decision-making, and various aspects of life. Understanding human emotion is crucial for empathetic and effective AI systems which can interact with humans in more natural and effective ways. Automatic emotion recognition (AER) aims at inferring the emotion state of a person. It has attracted increasing attention due to its wide potential application in human–computer interaction (Marín-Morales et al., 2020, Chowdary et al., 2023), healthcare (Lin et al., 2016, Chen et al., 2018, Zhang et al., 2021), education (Cen et al., 2016, Bahreini et al., 2016), agents and assistants (Picard, 2000, Qian et al., 2019, Spezialetti et al., 2020), entertainment (Yoon et al., 2007, Du et al., 2020), embodied AI (Duan et al., 2022), and beyond. AER offers opportunities to enhance user experiences, personalise interactions, and advance emotional wellbeing.

An AER system predicts the speaker's emotion state based on various input sources such as speech (Lugger and Yang, 2007, Shen et al., 2011, Wu et al., 2019, 2021), text (Taboada et al., 2011, Aydoğan and Akcayol, 2016, Mousa and Schuller, 2017, Liang et al., 2022), facial expressions (Pushpa et al., 2016, Li and Deng, 2020, Revina and Emmanuel, 2021, Li et al., 2021a), gestures (Glowinski et al., 2008, Saha et al., 2014, Sapiński et al., 2019, Huang et al., 2021), psychological signals (Ménard et al., 2015, Tang et al., 2021, Cai et al., 2021a, Dadebayev et al., 2022), and a combination of these inputs (Sebe et al., 2005, Bänziger et al., 2009, Tzirakis et al., 2017, Abdullah et al., 2021). AER that only uses speech as the input modality is also known as speech emotion recognition (SER), which will be the main focus of this thesis.

Current research directions for SER includes knowledge transfer between SER and other speech tasks (Bhosale et al., 2020, Lu et al., 2020, Pappagari et al., 2020, Nediyanchath et al., 2020, Zhang et al., 2022a), cross-corpus and cross-language SER (Zong et al., 2016, Deng et al., 2017, Abdelwahab and Busso, 2018, Parry et al., 2019, Gideon et al., 2019),

and development of new modelling structures (Wang et al., 2020, Xu et al., 2020, Wu et al., 2021, Shirian and Guha, 2021, Hu et al., 2022, Kim et al., 2022, Aftab et al., 2022, Chen et al., 2023a). Despite the progress, recognising emotion is still challenging because human emotion is inherently complex and ambiguous. It lacks clear temporal boundaries, can be easily affected by contextual information, and is extremely personal in expression and subjective in perception.

The rest of this chapter is organised as follows. Section 3.1 presents an introduction to theories of emotion. Section 3.2 provides a brief literature review of SER. Methods to collect emotional speech and commonly used benchmark emotion corpora are described in Section 3.3. Section 3.4 presents a SER system using foundation model. Current challenges in AER are discussed in Section 3.6, followed by the chapter summary.

3.1 Emotion Theories

Defining emotion is essential for establishing a criterion for emotion recognition. The basic theoretical framework for emotion was initially introduced by Ekman in the 1970s (Ekman, 1971). Psychologists across multidisciplinary fields such as neuroscience, philosophy, and computer science have proposed various theories to define emotion (Grandjean et al., 2008, Tracy and Randles, 2011, Gunes and Schuller, 2013, Marsella and Gratch, 2014). Although a universally recognised framework for emotion is still lacking, two theories are generally used to describe emotion: categorical emotion theory (also known as discrete emotion theory) (Ekman, 1971, Ekman et al., 1999) and dimensional emotion theory (also known as continuous emotion theory) (Mehrabian, 1980, Russell, 1980).

Categorical theory claims that there exists a small number of basic discrete emotions (*e.g.*, anger, disgust, happiness, sadness, fear, and surprise) that are inherent in our brain and universally recognised (Ekman, 1971, Picard, 2000). More than ninety definitions of the basic emotions have been proposed (Plutchik, 2001, 2003, Gunes et al., 2011). These basic emotions were derived with the following criteria (Ekman, 1971): (i) they come from human instinct; (ii) people can produce the same basic emotions under the same circumstances; (iii) people express basic emotions in similar ways. Although, the development of Ekman's basic emotion theory is based on the hypothesis that human emotions are universal across human ethnicity and cultures, different cultural backgrounds may have different interpretations of basic emotions, and different basic emotions can be mixed to produce complex or compound emotions (Ekman et al., 1999).

An alternative emotion description is dimensional emotion theory. In contrast to discrete emotions, dimensional emotion theory proposes several fundamental continuous-valued bipo-

lar dimensions and defines emotions as points in the dimensional emotion space. Commonly used dimensional emotion models includes valence-arousal (Russell and Mehrabian, 1977, Russell, 1979) and pleasure-arousal-dominance (Mehrabian, 1980, 1996). Valence (pleasure) dimension represents the magnitude of human joy from extreme ecstasy to distress (*i.e.*, positive or negative). The arousal (activation) dimension measures the intensity level (*i.e.*, excited or calm). The dominance (attention) dimension expresses the feeling of influencing or being influenced by the surrounding environment and others (*i.e.*, dominant or weak). It is claimed that the two dimensions of arousal and valence could represent the vast majority of different emotions (Russell and Mehrabian, 1977).

Categorical emotion theory is more intuitive to understand while dimensional theory allows the modelling of more subtle and complex emotions. It is noted that categorical and dimensional emotion representations can be transformed into each other to some extent. AER, in general, is either based on classification of categorical emotion categories, or based on regression of dimensional emotion attributes.

3.2 Speech Emotion Recognition

Speech emotion recognition (SER) detects the embedded emotions by processing and understanding speech signals (Lee and Narayanan, 2005). A SER system typically comprises two phases: emotion representation extraction from speech and classification/regression algorithms for emotion prediction (El Ayadi et al., 2011, Koolagudi and Rao, 2012)

3.2.1 Emotion Representation Extraction

Representations can be broadly grouped into hand-crafted features and learned embeddings. Hand-crafted features are manually engineered and require specialised prior knowledge while deep learning enables automatic representation learning.

Hand-crafted acoustic features, such as spectral features, prosodic features, and voice quality features, have been intensively explored and used for SER. Spectral features represent characteristics of the vocal tract. Commonly used spectral features include Mel frequency cepstral coefficients (MFCC) (Bitouk et al., 2010, Likitha et al., 2017, Gao et al., 2017, Daneshfar et al., 2020), Mel frequency filterbank energies (MFB) (Busso et al., 2007, Wu et al., 2021), and linear prediction cepstral coefficients (LPCC) (Shen et al., 2011, Seehapoch and Wongthanavasu, 2013, Sun et al., 2015). Prosodic features represent the long-time variations in perceived rhythm, stress, and intonation of speech, which can be perceived by humans. Popular examples include speaking rate, pitch, loudness, and energy dynamics.

In practice, the fundamental frequency (F0) and energy are the most widely used prosodic features as they relate to the perceptual characteristics of pitch and loudness (Lugger and Yang, 2007, Rajasekhar and Hota, 2018, Daneshfar et al., 2020, Wu et al., 2021). Voice quality measures the auditory perception of changes in vocal fold vibration and vocal tract shape, outside of pitch, loudness, and phonetic category. Commonly used voice quality features includes jitter, shimmer, and harmonic-to-noise ratio (Lugger and Yang, 2007, Guidi et al., 2019).

The time-consuming process of generating and selecting hand-crafted features has been gradually replaced by deep learning approaches which allow automatic representation learning and high-level data abstraction. Self-supervised learning (SSL) is currently the most common way to learn representations (see Section 2.2.2). SSL utilises information from the input data itself as labels, eliminating the need for costly manual labelling, thus benefiting from leveraging a large amount of unlabelled data for training. Representations extracted from SSL-pretrained speech foundation models (see Section 2.2.4) have achieved SOTA results in many speech processing tasks such as automatic speech recognition (ASR) and speaker verification (Baevski et al., 2020, Yang et al., 2021, Hsu et al., 2021, Chen et al., 2022). There is also a growing trend to adopt SSL representations for SER, which results in superior performance compared to hand-crafted features (Dissanayake et al., 2022, Morais et al., 2022, Zhang et al., 2022b, Zhao et al., 2022, Pasad et al., 2023, Li et al., 2023).

3.2.2 Models and Classifiers

Learning algorithms for SER can be broadly grouped into traditional machine-learning-based (ML-based) algorithms and deep-learning-based (DL-based) algorithms. Various ML-based classifiers have been implemented for SER, such as support vector machines (Seehapoch and Wongthanavasu, 2013, Gao et al., 2017, Rajasekhar and Hota, 2018, Kerkeni et al., 2019), random forests (Rong et al., 2007, Iliou and Anagnostopoulos, 2009, Dimitrova-Grekow and Konopko, 2019), k nearest neighbours (Zhang et al., 2010, Abdel-Hamid, 2020), logistic regression (Kerkeni et al., 2019, Zhu-Zhou et al., 2022), and Gaussian mixture models (Neiberg et al., 2006, Zhang et al., 2013).

The ability of deep learning to find intricate relationships and patterns has demonstrated superiority over traditional machine learning methods across diverse research domains, including SER. Commonly used DL models for SER includes CNNs (Mao et al., 2014, Badshah et al., 2017, Zhang et al., 2017), RNNs (Lee and Tashev, 2015, Ghosh et al., 2016, Mirsamadi et al., 2017, Atmaja and Akagi, 2019), attention mechanisms (Mirsamadi et al., 2017, Atmaja and Akagi, 2019, Wu et al., 2021, Wang et al., 2021, Goncalves and Busso, 2022), and the combination of them (Trigeorgis et al., 2016, Mirsamadi et al., 2017, Tzirakis

et al., 2018, Atmaja and Akagi, 2019, Neumann and Vu, 2019, Wu et al., 2019). The low-level and short-term discriminative capabilities of CNNs makes them effective for capturing local patterns and are thus commonly used to extract features from segments. RNNs are widely introduced to capture temporal dependencies between segments by maintaining internal memory states across time steps. Attention-based models (*i.e.,* Transformers) improve over RNNs in capturing long-range dependencies within sequences more effectively and efficiently and enabling parallel computing.

Recently, there has been a rising trend towards end-to-end training, where the entire system, from input to output, is jointly optimised and there's no clear boundaries between feature extraction and classification phases. CNNs can be directly applied to raw speech waveforms to extract relevant features, followed by Transformer-based encoder blocks that capture complex relationships between speech signals and emotion states (Yang et al., 2021). This approach leads to better performance compared to traditional pipelines with separate feature extraction and classification/regression stages.

3.3 Emotion Corpora

3.3.1 Approaches to Collecting Emotional Speech

Emotion datasets can be broadly grouped into three categories: acted, elicited, and natural datasets.

Acted Emotion

The acted emotion datasets are recorded by asking participants to express target emotions (Burkhardt et al., 2005, Busso et al., 2008, Cao et al., 2014). The acted emotion data is the simplest to collect among three dataset types. However, the acted renditions often resemble more prototypical and exaggerated behaviours, which might cause a mismatch with the complicated emotional expressions observed during daily interactions.

Elicited Emotion

Elicited speech is collected by putting a subject into such a situation that evokes a certain kind of emotion (Busso et al., 2008, Ringeval et al., 2013, Busso et al., 2017). Compared to the acted emotions, evoked emotions are closer to the emotion expressed in actual interactions. Typical approaches to eliciting emotions include scripted conversational settings that contains

emotion-dependent contextual information and simulated scenarios to recall memories of a past emotional experience.

Naturalistic Emotion

Naturalistic emotional speech databases are usually recorded from the general public conversations such as call centre conversations, artificial listener agents, and online websites (McKeown et al., 2012, Bagher Zadeh et al., 2018, Lotfian and Busso, 2019). In these spontaneous scenarios, the recordings do not follow a script and the participants are free to follow the flow of the conversation. While most naturalistic and authentic, it is difficult to control the emotion content and a large majority of common daily conversations are emotionally neutral. Therefore, most naturalistic emotion datasets suffer from severe class imbalance which poses challenges when training an emotion classifier. Furthermore, the recording protocol dictates that the emotional behaviours that are conveyed in the corpus, *e.g.*, call centre conversations might be biased toward negative behaviours. The biased emotional distribution caused by recording contextual scenarios can also lead to a mismatch with emotional behaviour observed in daily interactions.

3.3.2 Benchmark Datasets

Three benchmark emotion datasets used in this thesis are introduced below.

IEMOCAP

The interactive emotional dyadic motion capture (IEMOCAP) (Busso et al., 2008) corpus is one of the most widely used English datasets for verbal emotion classification. It consists of approximately 12 hours of audio-visual data, including speech, text transcriptions and facial recordings. It contains 5 dyadic conversational sessions performed by 10 professional actors with a session being a conversation between two speakers. In each session one male and female actor either performed selected emotional scripts or improvised based on hypothetical scenarios. There are in total 151 dialogues which includes 10,039 utterances. The recorded sessions were then manually segmented into utterances with an average duration of 4.5 s Each utterance was annotated by three human annotators for emotion class labels (neutral, happy, sad, and angry, *etc.*) and dimensional emotion attributes on a 5-point Likert scale (valence (1-negative vs 5-positive), activation (1-calm vs 5-excited), and dominance (1-weak vs 5-strong)). Each annotator was allowed to tag more than one emotion category for each sentence if they perceived a mixture of emotions. Ground-truth labels were determined by majority voting for categorical labels and averaging for dimensional labels.

MSP-Podcast

The MSP-Podcast database (Lotfian and Busso, 2019) contains naturalistic English speech from podcast recordings collected by the Multimodal Signal Processing (MSP) lab at the University of Texas at Dallas. Release 1.8 contains 73,042 utterances from 1,285 speakers amounting to more than 110 hours of speech. The average duration of an utterance is 5.6 s. The corpus was annotated using crowd-sourcing. Each utterance was labelled by at least 5 human annotators and has an average of 6.7 annotations per utterance. Each annotator selected a primary categorical emotion (only one option is allowed), secondary emotions (can select as many emotion classes as the annotator wish), as well as arousal, valence and dominance rankings on a 7-point Likert scale to each sentence. Ground-truth labels were determined by majority voting for categorical labels and averaging for dimensional labels. The data has been partitioned into the train set (44,879 segments), validation set (7,800 segments from 44 speakers (22 female, 22 male)), test set 1 (15,326 segments from 60 speakers (30 female, 30 male)) and test set 2 (randomly select 5,037 segments from 100 podcasts).

CREMA-D

The crowd-sourced emotional multi-modal actors dataset (CREMA-D) (Cao et al., 2014) contains 7,442 English utterances from 91 actors (48 male and 43 female between the ages of 20 and 74 coming from a variety of races and ethnicities). Actors spoke from a selection of 12 sentences using one of six different emotions (angry, disgust, fear, happy, neutral and sad) and four different emotion levels (low, medium, high and unspecified). The average duration of an utterance is 3.5 s. The dataset was annotated by crowd-sourcing. Annotators rated the emotion and emotion levels based on the combined audio-visual presentation, the video alone, and the audio alone. A total of 2,443 participants each rated 90 unique clips, 30 audio, 30 visual, and 30 audio-visual. 95% of the clips have more than 7 ratings and utterances have 9.21 ratings on average.

3.4 An AER System Based on Foundation Models

This section presents a SER system which follows the setup of the emotion recognition sub-task of the speech processing universal performance benchmark (SUPERB) (Yang et al., 2021). SUPERB defines a protocol to benchmark the performance of a shared model across a wide range of speech processing tasks with minimal architecture changes and labelled

data. Emotion recognition is one of the downstream tasks which assess model's capability of extracting paralinguistic information.

3.4.1 Experimental Setup

The model structure is illustrated in Fig. 3.1 which follows an upstream-downstream paradigm (Bommasani et al., 2021). The upstream model uses the USM model (Zhang et al., 2023) with 300M parameters which contains a CNN-based feature extractor and 12 Conformer (see Section 2.1.5) encoder blocks of dimension 1024 with 8 attention heads. The USM is pretrained by BEST-RQ (Chiu et al., 2022) which uses a BERT-style training task for the audio input to predict masked speech features. 128-dimensional MFBs were used as input to the upstream model. The downstream model performs utterance-level mean-pooling followed by a linear transformation with cross-entropy loss for emotion classification. The pretrained upstream USM model is frozen. The downstream model computes the weighted sum of the hidden states extracted from each layer of the upstream model. The weights are jointly trained with the downstream model.

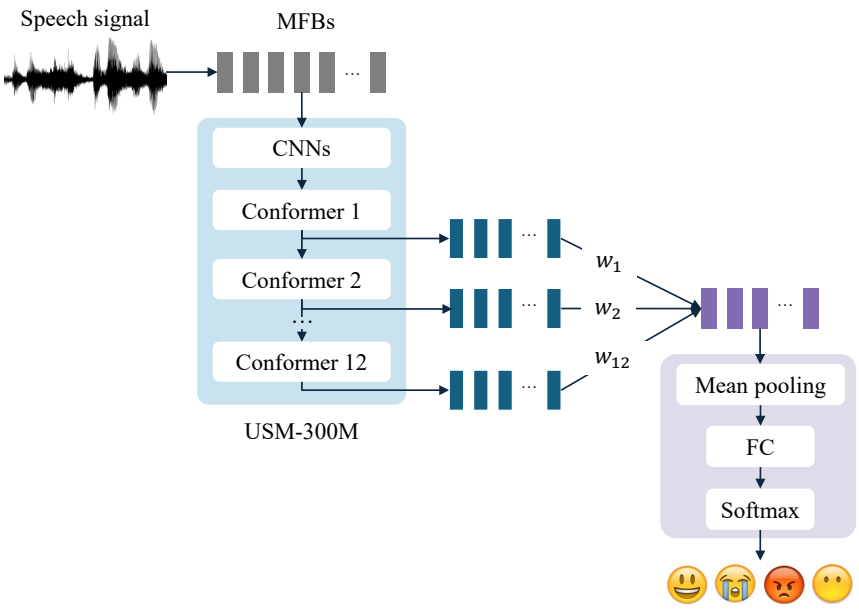


Fig. 3.1 Illustration of the model structure following SUPERB setup.

Following the SUPERB setup, the system is evaluated on four-way emotion classification on the IEMOCAP dataset with leave-one-session-out five-fold cross validation. The "excited" class was merged with "happy" and only utterances with ground-truth label belong to "happy", "sad", "angry", and "neutral" are considered, which results in a total of 5,531 utterances with 1,636, 1,103, 1,084, and 1,708 utterances for "happy", "angry", "sad" and "neutral"

respectively. This is also a commonly used setup for IEMOCAP dataset (Kim et al., 2013, Tripathi et al., 2018, Majumder et al., 2018, Poria et al., 2018, Liu et al., 2020, Chen and Zhao, 2020, Makiuchi et al., 2021). Despite being widely adopted, it is noted that this standard four-way setup discards nearly half of the data in IEMOCAP. More issues related to this setup with be discussed in Chapter 5.

3.4.2 Results

The system is compared to multiple models that use SOTA upstream models of similar size as the USM-300M model. The results are shown in Table 3.1. Except for the USM-300M model, all other results are from the cited papers. As shown in the table, the USM-based backbone structure outperforms other SOTA methods[1] and yields the highest emotion classification accuracy. This system will be used as baseline in Chapter 5.

Table 3.1 Four-way classification results on IEMOCAP following the SUPERB setup.

Upstream Model	# Parameters	%Accuracy
Wav2vec 2.0-large (Baevski et al., 2020)	317M	65.64
Data2vec-large (Baevski et al., 2022)	314M	66.31
HuBERT-large (Hsu et al., 2021)	317M	67.62
WavLM-large (Chen et al., 2022)	317M	70.62
USM-300M (Zhang et al., 2023)	290M	**71.06**

3.5 Ambiguity in Emotion Labelling

Emotion annotation is challenging due to the inherent ambiguity of mixed emotion, the personal variations in emotion expression, and the subjectivity in emotion perception (Cowen and Keltner, 2017, 2021). Mixed emotions and personal variations in emotion expression lead to inherent ambiguity of emotion. The subjectivity of emotional perception further complicates the problem that different people would interpret an emotion expression differently. In response to this problem, most datasets were created using the strategy of having multiple human annotators to provide multiple labels to each utterance. "Ground truth" is then commonly defined as the majority vote for discrete labels (Busso et al., 2008, 2017, Li et al., 2018, Poria et al., 2019) or the mean value for dimensional labels (Busso et al., 2008, Ringeval et al., 2013, Busso et al., 2017). However, such ground truth ignores the inherent uncertainty in emotion labelling.

[1] https://superbbenchmark.org/leaderboard, visited on December 15, 2023

3.5.1 Modelling Uncertainty in Categorical Emotion Labels

Instead of aggregating annotation by majority voting, some research suggests treating emotion classification as a multi-label task (Mower et al., 2010, Zadeh et al., 2018, Zhang et al., 2020, Ju et al., 2020, Chochlakis et al., 2023) where all emotion classes assigned by any annotator are considered as correct classes and the ground-truth label is presented as a multi-hot vector. The model is trained to predict the presence of each emotion class for each utterance. An issue with this approach is that it ignores the differences in strengths of different emotion classes. When more annotators are involved, it is likely that all possible emotions will eventually be marked correct for a given utterance.

An alternative approach uses "soft labels" as the proxy of ground truth, which is defined as the relative frequency of occurrence of each emotion class (Fayek et al., 2016, Han et al., 2017, Kim and Kim, 2018, Ando et al., 2018). The Kullback–Leibler (KL) divergence or distance metrics between the soft labels and model predictions are used to train the model. However, soft labels, being maximum likelihood estimates (MLE) of the underlying distribution based on observed samples, might not provide an accurate approximation to the unknown distribution when the number of observations (annotations) is limited. Also, although adopting soft labels, those methods still focus on obtaining a "correct" label (*i.e.,* pursuing improved classification accuracy). This results in a major inconsistency between training and evaluation (Mower et al., 2009).

3.5.2 Modelling Uncertainty in Dimensional Emotion Attributes

A straightforward way of aggregating emotional attribute annotations from several annotators is to take the average (Ringeval et al., 2013, Busso et al., 2008, Lotfian and Busso, 2019). Despite its simplicity, this method is problematic when annotators show a large degree of disagreement. Alternatively, an annotator-weighted mean has been proposed, which weights individual annotations based on inter-annotator agreement while filtering out unreliable annotators to improve the robustness of the results (Grimm and Kroschel, 2005, Kossaifi et al., 2019). However, these approaches force the discrepancies between annotators to be ignored.

Several approaches have been proposed to characterise the subjective property of emotion perception by modelling the inter-annotator disagreement level by the standard deviation of the dimensional labels, such as including a separate task to predict the standard deviation in a multi-task framework (Han et al., 2021, 2017), and predicting the standard deviation using Gaussian mixture regression models (Dang et al., 2017, 2018). Recently, alternative methods including Gaussian process (Atcheson et al., 2019), generative variational autoencoder (Srid-

har et al., 2021), and Monte Carlo dropout (Sridhar and Busso, 2020b) have been applied to the problem without explicitly using the standard deviation of dimensional emotion labels as additional training labels.

Another type of approach used for both categorical and dimensional annotations are multi-annotator models where multiple models are explicitly built, each emulating an individual annotator (Fayek et al., 2016, Chou and Lee, 2019, Davani et al., 2022). However, this approach is not scalable. It is computationally viable only when the number of annotators is relatively small, and it requires sufficient annotations from each annotator to be effective.

Beyond emotion recognition, label uncertainty is a common issue in many human perception and understanding tasks, since a ground-truth reference is usually not well-defined due to the subjective evaluation of annotators.

3.6 Challenges for Building an AER System

Apart from the ambiguity in emotion labelling described in the previous section that poses challenges in the problem formulation of emotion modelling, there exist other challenges for building an AER system towards practical applications.

3.6.1 Data Scarcity

Data scarcity is a major concern restricting the development of AER systems. Most publicly available emotion datasets suffer from limited size and limited number of speakers (Koolagudi and Rao, 2012, Pushpa et al., 2016), offer only a few hours of recordings and fewer than 20 speakers, which limits the generalisation of AER systems. Different people express emotions differently, so it is important to include a range of speakers so that the model can capture the intrinsic inter-speaker variability associated with the expression of emotion. Moreover, different acoustic condition and different annotation criteria makes cross-corpus adaptation challenging.

3.6.2 Lack of Naturalness and Imbalanced Emotional Content

Apart from limited size, the lack of naturalness is another issue associated with emotion corpora. As discussed in Section 3.3, acted emotion datasets recorded by asking participants to express target emotions is relatively more readily available while the emotions collected tend to be more exaggerated. However, naturalistic emotion databases usually suffer from imbalanced emotional content (*i.e.*, dominated by neutral emotion). It can be challenging

for an AER system to learn the minority emotion classes which can only contain a few samples (Lotfian and Busso, 2019).

3.6.3 Assumption of Reference Text and Segmentation

Apart from the emotion ambiguity and data issues, there are some common assumptions that also cause a mismatch between research experiments and practical applications such as the use of reference transcriptions and segmentation.

Although text information has been shown effective for AER (Poria et al., 2018, Wu et al., 2021), manually transcribed reference transcriptions are commonly used which are usually not available in practice. ASR systems trained on standard speech corpora can give poor recognition performance on emotional speech (Fernandez, 2004, Sahu et al., 2019) and replacing reference transcriptions by erroneous ASR output leads to marked decrease in AER performance (Wu et al., 2021). This motivates the study of multi-task training and transfer learning to improve the AER performance with ASR transcriptions (Feng et al., 2020, Cai et al., 2021b, Ghriss et al., 2022, Li et al., 2022).

Emotion can be labelled at either frame-level (*e.g.*, tens of milliseconds (Ringeval et al., 2013)) or utterance-level (or turn-level, *e.g.*, several seconds (Busso et al., 2008, Lotfian and Busso, 2019)). Frame-level labelling is typically used for emotion attributes, while utterance-level labelling is commonly used for both emotion classes and attributes. In the latter case, the segmentation of utterance can be important especially in conversational cases involving multiple speakers where diarisation is needed before applying AER. There is a current absence of suitable metrics to evaluate AER performance when segmentation errors are present. In such cases, classification accuracy is insufficient as it can not handle segment alignment.

3.7 Chapter Summary

This chapter introduces the background about automatic emotion recognition, including description of emotion states (Section 3.1), literature review on SER systems (Section 3.2), commonly used emotion corpora (Section 3.3), an example AER system (Section 3.4), as well as current challenges in AER including ambiguity in emotion labelling (Section 3.5) and limitations for building practical systems (Section 3.6).

Chapter 4

An Integrated System for AER and ASR with Automatic Segmentation

As discussed in Section 3.6.3, although AER has drawn significant research interest, there is still a mismatch between research experiments and practical applications. For example, most current AER studies use manually segmented utterances without considering potential sentence segmentation errors in practical dialogue systems. Moreover, automatic speech recognition (ASR) systems trained on standard speech corpora can give poor recognition performance on emotional speech (Fernandez, 2004, Sahu et al., 2019, Wu et al., 2021). This chapter proposes integrating AER with ASR and speaker diarisation in a jointly-trained system. Distinct output layers are built for four sub-tasks including AER, ASR, voice activity detection and speaker classification based on a shared encoder. Taking the audio of a conversation as input, the integrated system finds all speech segments and transcribes the corresponding emotion classes, word sequences, and speaker identities. Two metrics are proposed to evaluate AER performance with automatic segmentation: the time-weighted emotion error rate (TEER) and the speaker-attributed time-weighted emotion error rate (sTEER). This chapter is an extended version of the publication (Wu et al., 2023b)[1], which is the first work that considers emotion recognition with automatic segmentation and integrates emotion recognition, speech recognition and speaker diarisation into a jointly-trained model.

The rest of this chapter is organised as follows. Section 4.1 introduces the proposed integrated system. Section 4.2 introduces the TEER and sTEER metrics. The experimental setup and results are shown in Section 4.3 and Sections 4.4 respectively. Analysis and discussion are provided in Section 4.5 and are followed by the chapter summary.

[1] See Appendix A for a list of publications related to the thesis.

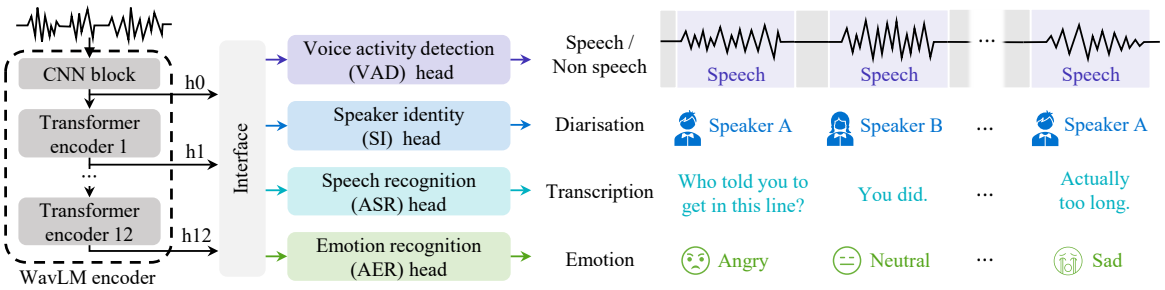


Fig. 4.1 Overview of the proposed integrated system. Taking a dialogue as input, the VAD and SI head perform automatic segmentation. The ASR and AER head recognise the text and emotion based on the predicted segments.

4.1 An Integrated System

This chapter proposes an integrated system for emotion recognition, speaker diarisation and speech recognition. Speaker diarisation is the process that detects speech regions of an audio recording and groups them into homogeneous segments according to the relative identity of the speaker. The outcome of speaker diarisation is referred to as segmentation in this chapter. As illustrated in Fig. 4.1, the system takes the audio recording of a dialogue as input. It automatically diarises the dialogue into segments associated with different speakers, transcribes the audio segments into text, and predicts the speaker's emotion state.

The structure of the proposed system is shown in Fig. 4.2, which contains four downstream heads for voice activity detection (VAD), speaker identity (SI) extraction, ASR, AER and an encoder shared by all downstream heads. Speaker diarisation is achieved by the VAD head and the SI head. The VAD head classifies each frame into speech or non-speech. The SI head learns speaker embeddings that capture the characteristics of each speaker. Based on the predicted segmentation, the ASR head converts each speech segment into text and the AER head predicts the emotion states of the corresponding speaker. A WavLM model (see Section 2.2.4) is used as the shared encoder which takes raw speech waveform as input. The downstream heads take the weighted sum of intermediate hidden states from the shared encoder as input, shown as interface in Fig. 4.2. Each head has an individual set of weights, which are trained jointly with the shared encoder and the downstream heads.

4.1.1 Shared Encoder and Interface

The WavLM model (see Section 2.2.4) is used as the shared encoder in this chapter, which takes the raw waveform as input. The base version is used which contains a CNN block as the feature extractor and 12 Transformer encoder blocks with 768-dimensional hidden states

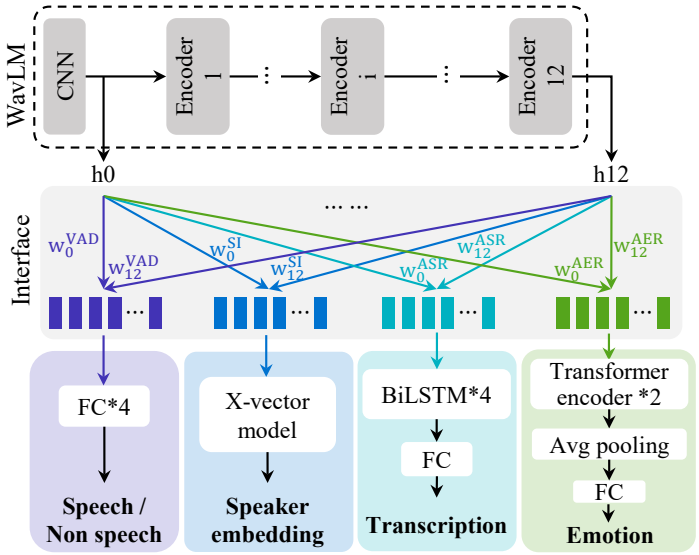


Fig. 4.2 Structure of the proposed integrated system. The intermediate representations of the WavLM model ($\boldsymbol{h}$ vectors in the figure) are summed with trainable weights.

and 8 attention heads. The output of the encoder is a frame sequence with a frame shift of 20 ms.

All the semantic and non-semantic information co-exist in the same speech signal. Research (Pasad et al., 2021, Yang et al., 2021, Chen et al., 2022) has shown that intermediate representations of such foundation models contain different levels of information. The weighted sum of embeddings from the CNN block and each Transformer encoder block is used as the input to the downstream tasks to exploit this property. Each downstream head has its own set of weights which are trainable.

4.1.2 Downstream Heads

The VAD head consists of 3 FC layers with a hidden dimension of 256 and leaky ReLU activation plus an output FC layer with softmax activation which performs frame-level speech/non-speech detection. The SI head consists of an X-vector speaker embedding model (Snyder et al., 2018), which generates one speaker embedding for each input sequence. The speaker embedding is fed into a FC layer with softmax activation for speaker classification during training. During testing, spectral clustering is conducted based on speaker embeddings to produce speaker diarisation. The ASR head consists of 4 Bi-LSTM layers (see Section 2.1.3) with dimension of 256, followed by a FC layer for token prediction. A vocabulary of 29 graphemes is used[2]. The AER head consists of 2 Transformer encoder layers

[2]The vocabulary includes 26 uppercase English letters, along with space, period, and <blank>.

of dimension 256. The representations are mean-pooled along the time axis before feeding into a FC layer with softmax activation for emotion classification. Six emotion classes are used: “happy”, “sad”, “angry”, “neutral”, “other”, “no majority agreement (NMA)”. “NMA” denotes that the human annotators don’t have a majority agreed emotion class label for this utterance.

4.1.3 Multi-Task Training Loss

Apart from the ASR head, the other three heads are trained using the cross-entropy loss (see Section 2.2.1). The ASR head is trained using the connectionist temporal classification (CTC) loss (Graves et al., 2006), which is commonly used in sequence-to-sequence tasks such as ASR. The CTC loss considers all possible alignments of the input speech sequence to the target text sequence and computes the negative log probability of the correct output sequence. The shared encoder and four downstream heads are jointly trained using a multi-task loss:

$$\mathcal{L}_{\text{Total}} = \epsilon_{\text{VAD}}\mathcal{L}_{\text{VAD}} + \epsilon_{\text{SI}}\mathcal{L}_{\text{SI}} + \epsilon_{\text{ASR}}\mathcal{L}_{\text{ASR}} + \epsilon_{\text{AER}}\mathcal{L}_{\text{AER}} \tag{4.1}$$

where $\epsilon_{\text{VAD}}, \epsilon_{\text{SV}}, \epsilon_{\text{ASR}}, \epsilon_{\text{AER}}$ are coefficients that are set manually to keep the weighted loss of the four heads in the same scale.

4.1.4 Training and Testing Procedures

Reference segmentations are used during training. The system takes segmented utterances as input. The encoder and downstream heads are trained jointly using the multi-task loss defined in Eqn. (4.1). The segmented utterances can contain silence at the beginning, between words and at the end. The VAD head is trained based on the intra-utterance silence.

During testing, dialogues are input into the system. The system can benefit from knowing the context. A sliding window of 3 s length and 1 s overlap is applied to the dialogue. VAD is performed on each window. To avoid overlapped regions being counted twice at the output, the results of the middle second is kept for each window. This is equivalent to taking the previous 1 s and future 1 s as context when making a prediction on the current 1 s of audio data. Post-processing is applied to the VAD predictions. Speech/non-speech regions shorter than 0.25 s are removed. A smaller sliding window of 1 s length and 0.5 s overlap is then applied to the detected speech regions. The SI head extracts a single speaker embedding from each window. Spectral clustering is used based on the speaker embeddings which groups segments from the same speaker together, thus producing an automatic segmentation. Based

on the automatic segmentation, the ASR and AER heads takes each segment as input and predict the text and emotion respectively.

4.2 Evaluating Emotion Classification with Automatic Segmentation

The segments predicted by the system can have different start and end times to the reference segments. Therefore, the classification accuracy is no longer sufficient to evaluate the performance of emotion classification in this case since it cannot handle the alignment between segments. This chapter, therefore, proposes the time-weighted emotion error rate (TEER) in order to evaluate the AER performance given non-oracle segmentations. The TEER is computed as follows:

$$\text{TEER} = \frac{\text{MS} + \text{FA} + \text{CONF}_{\text{emo}}}{\text{TOTAL}} \tag{4.2}$$

where missed speech (MS) is the duration of speech incorrectly classified as non-speech, false alarm speech (FA) is the duration of non-speech incorrectly classified as speech, confusion (CONF_{emo}) is the audio duration where emotion is wrongly classified, and TOTAL is the sum of the reference speech duration for all utterances.

Furthermore, the speaker-attributed TEER (sTEER) is proposed which expects the system to accurately predict both the speaker and the corresponding emotion. The sTEER is computed as follows:

$$\text{sTEER} = \frac{\text{MS} + \text{FA} + \text{CONF}_{\text{emo+spk}}}{\text{TOTAL}} \tag{4.3}$$

where CONF_{emo} in Eqn. (4.2) is replaced by $\text{CONF}_{\text{emo+spk}}$, which is the duration where either speaker or emotion is wrong. sTEER reflects the overall performance of both speaker diarisation and emotion classification.

4.3 Experimental Setup

4.3.1 Dataset

The IEMOCAP dataset (see Section 3.3.2) is used in this chapter which provides the time-stamp of each utterance in a dialogue as well as word-level alignments of each utterance. The alignments show that 40% frames in the segmented utterances are silence. Each utterance received at least three annotations. The ground-truth class is defined via majority voting. A

Table 4.1 Statistics of six-way classification setup of IEMOCAP.

Emotion class	Happy	Sad	Angry	Neutral	Others	NMA
# of utterances	1,636	1,103	1,084	1,708	2,001	2,507

six-way classification setup is used in this chapter. The emotion class "excited" is merged with "happy". All sentences with ground-truth emotion label other than " happy", "sad", "angry", "neutral" are grouped into the class "others". Sentences that don't have a majority agreed emotion label (*i.e.*, tied votes) from the annotators are grouped into the sixth class "NMA"[3], which stands for no majority agreed. The number of utterances in each class are given in Table 4.1. Speaker exclusive leave-one-session-out five-fold cross validation was performed and the average results are reported. Speakers in the test set are unseen in the training and validation set and utterances from the same dialogue are either all in the training set or all in the validation set.

4.3.2 Evaluation Metrics

The false alarm rate (FAR) and missed speech rate (MSR) were used to evaluate the VAD performance. FAR computes the ratio of the number of non-speech frames mispredicted as speech to the total number of speech frames. MSR computes the ratio of the number of speech frames mispredicted as non-speech to the total number of speech frames.

The diarisation error rate (DER) is used to evaluate the performance of diarisation which maps the predicted relative speaker identity to the true speaker identity and measures the fraction of time not attributed correctly to a speaker or to non-speech. DER is defined as the duration of false alarm (FA), missed detection (MS), and speaker confusion errors (CONF_{spk}) divided by the ground-truth duration:

$$\text{DER} = \frac{\text{MS} + \text{FA} + \text{CONF}_{\text{spk}}}{\text{TOTAL}}. \tag{4.4}$$

Overlapped speech is considered when computing DER. Since manual annotations cannot be precise at the audio sample level, it is common to remove from evaluation a forgiveness collar around each segment boundary. Unless otherwise mentioned, a collar of 0.25 s is applied when evaluating with automatic segmentation.

The word error rate (WER) and classification accuracy (ACC_{emo}) are used to evaluate the performance of speech recognition and emotion classification respectively with oracle segmentation. WER is the ratio of errors in a transcript to the total words spoken, which is

[3]The ground truth of NMA utterances are marked as "xxx" in the database documentation.

defined as follows:

$$\text{WER} = \frac{\text{SUB} + \text{DEL} + \text{INS}}{\text{TOTAL}} \tag{4.5}$$

where SUB is the number of substitutions, DEL is the number of deletions, INS is the number of insertions, and TOTAL is the number of words in the reference.

With automatic segmentation, the concatenated minimum-permutation word error rate (cpWER) (Watanabe et al., 2020) is used to evaluate the ASR system performance which concatenates utterances of the same speaker and computes the WER. The sTEER and TEER are used to evaluate the AER system which have been introduced in Section 4.2.

4.3.3 Baselines

Since this is the first work to examine AER with automatic segmentation, there are no readily available systems or published numbers for direct comparison. Therefore, two additional baseline systems were built for comparison.

- "Baseline-reference": A cascaded system of separately optimised models (reference models) which have been trained on external larger datasets. The reference ASR model was pretrained using 100 hours of LibriSpeech (Panayotov et al., 2015)[4] training data, which has a WER of 5.64% on "test-clean" set and 12.15% on "test-other" set. The reference speaker embedding model[5] was pretrained on Voxceleb 1.0 (Nagrani et al., 2017)[6]. Both the reference ASR model and the reference speaker embedding model consist of a WavLM encoder and a downstream model with the same structure as the corresponding head of the proposed system. A VAD module (Sun et al., 2021) pretrained on the augmented multi-party interaction (AMI) meeting corpus (Carletta et al., 2005)[7] was used as the reference baseline for VAD. It consists of seven FC layers with ReLU activation functions and has 2.1% FAR and 4.7% MSR on the AMI eval set. No reference model was used for AER since we are evaluating on the emotion dataset. The reference system is the cascade of the reference models in the order of VAD, SI, and ASR/AER.

- "Baseline-frozen": A system which shares the same structure as the proposed system except that the shared encoder was frozen during training. In this case, the four downstream heads are independent of each other and the multi-task loss becomes equivalent to training each head separately.

[4]More details about the LibriSpeech dataset can be found in Appendix B.1.
[5]Available at: https://huggingface.co/microsoft/wavlm-base-plus-sv
[6]More details about the Voxceleb 1.0 dataset can be found in Appendix B.2.
[7]More details about the AMI meeting dataset can be found in Appendix B.3.

4.3.4 Training Specifications

The shared encoder was initialised with the publicly available WavLM Base+ model[8]. It was finetuned jointly with the downstream head while the CNN feature extractor of the WavLM model was frozen during finetuning. Speed perturbation was applied to the ASR and SI heads. For each epoch, the speed of each waveform was randomly adjusted to 0.95 or 1.05 of the original speech or remain unchanged. Speed perturbation was not applied to AER since speed is an important clue for emotion detection. Scaling coefficients ϵ_{VAD} and ϵ_{SI} in Eqn. (4.1) were set to 1.2 while ϵ_{ASR} and ϵ_{AER} were set to 1.

4.4 Experimental Results

The performance of the SI, ASR and AER heads were first evaluated with oracle segmentations in Section 4.4.1. The complete system was then evaluated with automatic segmentation in Section 4.4.2.

4.4.1 Performance with Oracle Segmentation

In this section, utterances based on the reference segmentation were used as input to the SI, ASR and AER heads. The results are shown in Table 4.2. Comparing "Baseline-reference", pretrained on external larger datasets, to "Baseline-frozen", where the encoder was frozen and the downstream models were finetuned on the IEMOCAP dataset, the ASR head with a frozen encoder reduced the WER from 33.3% to 31.4%, and the SI head with a frozen encoder reduced the DER from 1.1% to 0.4%. The proposed system with the shared encoder jointly finetuned with downstream heads further reduced the WER and DER to 24.6% and 0.3% respectively. The 6-way emotion classification accuracy increased from 44.4% to 49.5%. The proposed integrated system outperforms the baselines for all three heads, which

Table 4.2 Results with reference segmentation on IEMOCAP. The collar was set to 0 when computing DER since oracle segmentation was assumed. "↑" denotes the higher the better, "↓" denotes the lower the better. The best results of each column are shown in bold.

	%ACC$_{\text{emo}}$↑	**%WER↓**	**%DER↓**
Baseline-reference	/	33.3	1.10
Baseline-frozen	44.4	31.4	0.40
Proposed	**49.5**	**24.6**	**0.30**

[8]Available at: https://huggingface.co/microsoft/wavlm-base-plus

Table 4.3 VAD and speaker diarisation results on IEMOCAP. "↓" denotes the lower the better. FAR stands for false alarm rate. MSR stands for missed speech rate. The best results of each column are shown in bold.

	%FAR↓	**%MSR↓**	**%DER↓**
Baseline-reference	5.15	1.30	8.20
Baseline-frozen	3.14	1.16	7.04
Proposed	**2.91**	**1.06**	**6.87**

Table 4.4 Speaker-attributed ASR and AER performance under automatic segmentation on IEMOCAP. "↓" denotes the lower the better. The best results of each column are shown in bold.

	%cpWER↓	**%sTEER↓**	**%TEER↓**
Baseline-reference	43.8	/	/
Baseline-frozen	41.2	69.5	68.7
Proposed	**36.2**	**66.0**	**65.2**

indicates that finetuning the pretrained encoder on emotion data helps to adapt it to the specific domain, while sharing the encoder between the four downstream heads helps to capture general information relevant to the domain and avoids overfitting to trivial patterns, especially given the scarcity of data.

4.4.2 Performance with Automatic Segmentation

The VAD performance and diarisation results based on VAD predictions are summarised in Table 4.3. The proposed system produced the best results on both VAD and diarisation. It improves over "Baseline-frozen" with a relative decrease of 7.3% for FAR, 8.6% for MSR, and 2.4% for DER.

ASR and AER were conducted based on the diarisation outputs. As shown in Table 4.4, the proposed integrated system reduced cpWER by 12% and both sTEER and TEER by 5% relative to the baselines. sTEER is slightly higher than TEER as it takes speaker prediction error into account. The proposed system outperforms the single AER head in both emotion metrics, showing its superior performance for emotion recognition with automatic segmentation.

4.5 Discussion and Analysis

4.5.1 Trainable Weights of the Interface

The trainable weights of the four downstream heads are plotted in Fig. 4.3. As can be seen, layer 0 and layer 4 are particularly useful for extracting speaker information. Layers 8-10 are more effective for AER and layer 11 contains most text information. This shows a similar pattern to previous findings (Pasad et al., 2021, Chen et al., 2022) that block-wise evolution of intermediate representations of a foundation model follows an acoustic-linguistic hierarchy, where the lower layers encode speaker-related information and higher layers encode phonetic/semantic information.

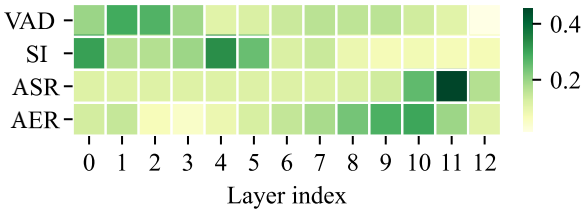


Fig. 4.3 Weights of the interface for different downstream heads.

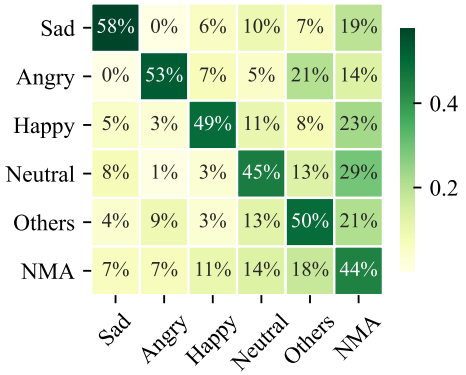


Fig. 4.4 Confusion matrix of six-way emotion recognition.

4.5.2 Confusion Matrix of Six-Way Emotion Classification

Based on the 6-way predictions, 4-way classification accuracy considering “happy”, “sad”, “angry”, “neutral” is 74.0%, which is better than the results of Wav2vec 2.0 Base (63.4%) and WavLM Base+ (68.7%) from the SUPERB leaderboards (Yang et al., 2021). The class “NMA” is relatively easily confused, as shown by the 6-way confusion matrix in Fig. 4.4. For utterances classified as “NMA”, the human annotators gave different emotion class labels and didn’t reach majority agreement. These utterances may contain ambiguous emotions, mixed emotions, or emotions that tend to confuse the annotators. Among the other five classes, “angry” is the least likely to be confused with “NMA” probably because “angry” is relatively less ambiguous. By contrast, “neutral” is more likely to be wrongly predicted as “NMA”, possibly because neutral emotions are relatively weak and human annotators are likely to disagree due to subjective perception. Alternative ways to deal with “NMA” utterances will be further discussed in Chapter 5.

4.6 Chapter Summary

This chapter introduces a system that integrates emotion recognition with speech recognition and speaker diarisation in a jointly-trained model, which is the first to investigate emotion recognition with automatic segmentation as needed in practical applications. The time-weighted emotion error rate (TEER) and speaker-attributed time-weighted emotion error rate (sTEER) have been proposed to evaluate emotion classification performance when segmentation is non-oracle. Results on the IEMOCAP dataset show that the proposed jointly-trained system consistently outperforms two strong baselines with separately optimised single-task systems on all four tasks evaluated: voice activity detection, speaker diarisation, speech recognition, and emotion recognition. Apart from enabling emotion recognition with automatic segmentation, the system also improves speech recognition performance of emotional speech given a 12% relative reduction in word error rate.

Chapter 5

Handling Ambiguity in Emotion Class Labels

As discussed in Section 3.5, the inherent subjectivity of human emotion perception introduces complexity in annotating emotion datasets. Multiple annotators are often involved in labelling each utterance. Several typical annotation situations are given in Table 5.1.

Table 5.1 Typical situations for emotion class annotations. "MA" stands for majority agreed. "NMA" stands for no majority agreed.

Annotations	Majority Vote	MA/NMA
Happy, Happy, Happy	Happy	MA
Happy, Happy, Neutral	Happy	MA
Happy, Neutral, Angry	–	NMA
Happy, Happy, Neutral, Neutral	–	NMA

The majority-agreed (MA) class is usually used as the ground truth (Busso et al., 2008, Cao et al., 2014, Busso et al., 2017). Utterances that have no majority agreed (NMA) labels (*i.e.*, with tied votes) are typically excluded when training an emotion classifier (Kim et al., 2013, Poria et al., 2017, Wu et al., 2021, Yang et al., 2021). Excluding NMA utterances from training can pose problems. First, it discards a considerable amount of data (*e.g.*, 25% of IEMOCAP), which is not ideal given the already small size of emotion datasets. More importantly, ambiguous emotions are prevalent in daily life. Excluding them may lead to issues when the AER system encounters such expressions in practical applications.

This chapter investigates three methods to handle ambiguous emotion[1]:

[1]Work performed while W. W. was an intern at Google. Part of this chapter has been published as a conference paper (Wu et al., 2024b). See Appendix A for more detail.

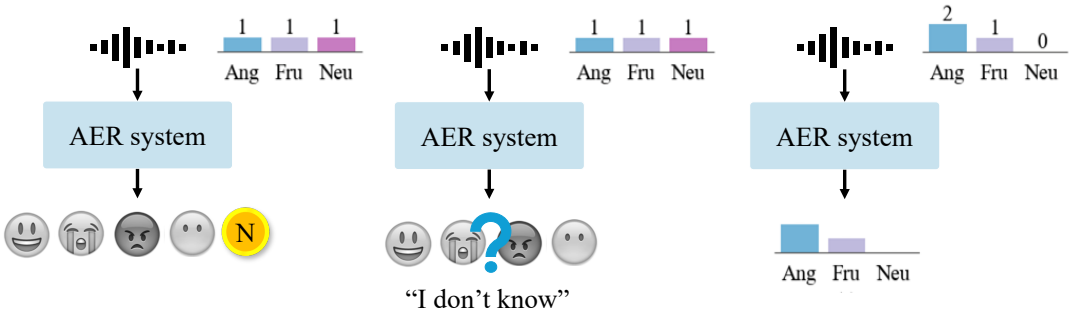


Fig. 5.1 Illustration of the three approaches investigated in this chapter for handling ambiguity in emotion. Three emotion classes are considered in this example: **Ang**ry, **Fru**strated, **Neu**tral.

1. Similar to the approach adopted in Chapter 4, NMA utterances are incorporated as an additional class in the classifier, as illustrated in Fig. 5.1 (a). This approach proves problematic as it reduces the classification performance of the other emotion classes.

2. NMA utterances are detected as out-of-domain (OOD) samples by quantifying the uncertainty in emotion classification using evidential deep learning (Sensoy et al., 2018). When a classifier trained on MA data encounters an NMA utterance during the test, the model should identify it as an OOD sample by providing a high uncertainty score, indicating its uncertainty regarding the specific emotion class to which the NMA utterance belongs, as illustrated in Fig. 5.1 (b). This approach retains the classification accuracy while effectively detects ambiguous emotion expressions.

3. To further obtain fine-grained distinctions among ambiguous emotions, emotion is represented as a distribution instead of a single class label, as illustrated in Fig. 5.1 (c). The task is thus re-framed from classification to distribution estimation where every individual annotation is taken into account, not just the majority opinion. The evidential uncertainty measure is extended to quantify the uncertainty in emotion distribution estimation.

The rest of the chapter is organised as follows. Section 5.1 presents the general experimental setup used in this chapter. Section 5.2 introduces the first approach which incorporates NMA as an additional class. Section 5.3 introduces the second approach which quantifies uncertainty in emotion classification via evidential deep learning. Section 5.4 introduces the third approach which represents emotion as a distribution instead of a single class. Section 5.5 provides an analysis of cases when the second approach fails and how the third approach comes to rescue in those cases, followed by chapter summary.

5.1 Experimental Setup

5.1.1 Datasets

The IEMOCAP and CREMA-D datasets are used in this chapter (see Section 3.3.2). For the IEMOCAP dataset, the annotations are grouped into five emotion classes: happy (merged with excited), sad, neutral, angry, and others. The "others" category includes all emotions not covered in the previous four categories which is dominated by frustration (92%). Based on the grouped emotion categories, 14.2% of the utterances don't have a majority agreed emotion class label[2] and only 16.1% of the utterances have an all-annotators-agreed emotion label. CREMA-D contains six emotion categories: anger, disgust, fear, happy, neutral and sad. 5.1% of utterances have an all-annotators-agreed emotion label and 8.7% don't have a majority agreed emotion class label.

Both datasets are divided into a MA (majority agreed) subset and a NMA (no majority agreed) subset. All methods were trained only on MA data except for the first approach where 25% of NMA utterances were reserved for testing and the rest were included in training. For IEMOCAP, Session 5 was reserved for testing, and Sessions 1-4 were split into training and validation with a ratio of 4:1. For the CREMA-D dataset, the MA subset was split into train, validation, test in the ratio 70 : 15 : 15 following prior work (Ristea and Ionescu, 2021).

5.1.2 Model Structure and Implementation Details

The model structure in Section 3.4 was used as backbone in this chapter, which follows the SUPERB (Yang et al., 2021) setup and uses USM-300M (Zhang et al., 2023) as the upstream foundation model. Since the CREMA-D dataset is extremely imbalanced (*i.e.*, neutral accounts for over 50%), a balanced sampler was applied during training.

5.2 Ambiguous Emotion as an Additional Class

First, a naive method was tested which aggregates NMA utterances into an additional class when training an emotion classifier. Some of the NMA utterances need to be involved during training.

[2]This number differs from that in Chapter 4 because Chapter 4 directly groups the ground-truth labels (majority votes) provided by the dataset following conventional approaches, while in this chapter, the raw annotations are re-processed by first performing grouping and then conducting majority voting.

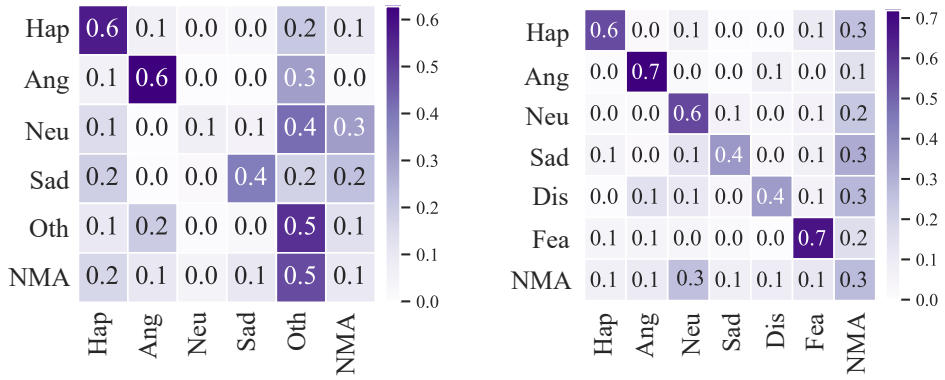
5.2.1 Experiments: Including NMA as an Extra Class

The classification performance on MA data was compared between the original system trained using only MA utterances and the first approach that added NMA as an additional class during training. The classification performance was evaluated by classification accuracy (ACC) and unweighted average recall (UAR) which is the sum of class-wise accuracy divided by the number of classes. The results are shown in Table 5.2, which reveals that the addition of the NMA class has a detrimental impact on the classification performance of the original MA emotion classes. Comparing to the original classifier, the first approach yields a ~23% relative decrease in both ACC and UAR on IEMOCAP and a ~20% relative decrease in ACC and UAR on CREMA-D.

Table 5.2 Classification performance on MA utterances when including NMA as an additional class for IEMOCAP and CREMA-D.

	IEMOCAP		**CREMA-D**	
	ACC↑	**UAR↑**	**ACC↑**	**UAR↑**
Original MA classifier	0.582	0.577	0.714	0.672
+ an extra NMA class	0.447	0.438	0.568	0.540

The confusion matrices are shown in Fig. 5.2. It can be seen from the bottom right entry that NMA itself is challenging to predict, possibly because it essentially contains a mix of different emotion content. The last column demonstrates that grouping these utterances into one class can confuse the model, particularly for the classes neutral, sad, frustrated, and disgust.



(a) IEMOCAP (b) CREMA-D

Fig. 5.2 Confusion matrix of the first approach on IEMOCAP and CREMA-D where NMA is included as an additional class.

5.3 OOD Detection by Quantifying Emotion Classification Uncertainty

Despite its simplicity, the first approach discussed in Section 5.2, which includes NMA as an additional class, diminishes the classification performance of the MA classes. This section studies an alternative method: whether an emotion classifier can appropriately respond with "I don't know" for ambiguous emotion data that does not fit into any predefined emotion classes. For an emotion classifier trained on MA utterances, NMA utterances that haven't been seen during training can be treated as out-of-domain (OOD) samples. The model is expected to output a high uncertainty score when encountering ambiguous emotions, indicating that the utterance doesn't belong to any of the predefined MA classes. This is realised by quantifying the uncertainty in emotion classification using evidential deep learning (EDL) (Sensoy et al., 2018).

5.3.1 Limitations of Softmax Activation Function

A neural network model classifier transforms the continuous logits at the output layer into class probabilities by a softmax function (Eqn. (2.3)). The model prediction can thus be interpreted as a categorical distribution with the discrete class probabilities associated with the model outputs. The model is then optimised by maximising the categorical likelihood of the correct class, known as the cross-entropy loss.

However, the softmax activation function is known to have a tendency to inflate the probability of the predicted class due to the exponentiation applied to transform the logits, resulting in unreliable uncertainty estimations (Gal and Ghahramani, 2016, Guo et al., 2017). Furthermore, cross-entropy is essentially a maximum likelihood estimate (MLE), a frequentist technique lacking the capability of inferring the variance of the predictive distribution.

In the following section, the model uncertainty is estimated using evidential deep learning (EDL) (Sensoy et al., 2018) which places a "second-order probability"[3] over the categorical distribution.

5.3.2 Evidential Deep Learning

Consider an emotion class label as a one-hot vector $\boldsymbol{y}$ where $\boldsymbol{y}_k$ is one if the emotion belongs to class k else zero. $\boldsymbol{y}$ is sampled from a categorical distribution $\boldsymbol{\pi}$ where each component

[3] It models the distribution of a distribution.

π_k corresponds to the probability of sampling a label from class k:

$$\boldsymbol{y} \sim \mathrm{P}(\boldsymbol{y}|\boldsymbol{\pi}) = \mathrm{Cat}(\boldsymbol{\pi}) = \boldsymbol{\pi}_k^{\boldsymbol{y}_k}. \tag{5.1}$$

To model the probability of the predictive distribution, the categorical distribution is assumed to be sampled from a Dirichlet distribution:

$$\boldsymbol{\pi} \sim \mathrm{p}(\boldsymbol{\pi}|\boldsymbol{\alpha}) = \mathrm{Dir}(\boldsymbol{\pi}|\boldsymbol{\alpha}) = \frac{1}{\mathrm{B}(\boldsymbol{\alpha})} \prod_{k=1}^{K} \boldsymbol{\pi}_k^{\boldsymbol{\alpha}_k - 1} \tag{5.2}$$

where $\mathrm{B}(\cdot)$ is the Beta function:

$$\mathrm{B}(\boldsymbol{\alpha}) = \frac{\prod_{k=1}^{K} \Gamma\left(\boldsymbol{\alpha}_k\right)}{\Gamma\left(\sum_{k=1}^{K} \boldsymbol{\alpha}_k\right)} \tag{5.3}$$

where $\Gamma(\cdot)$ denotes the Gamma function $\Gamma(z) = \int_0^\infty \boldsymbol{x}^{z-1} e^{-x} \mathrm{d}\boldsymbol{x}$. $\boldsymbol{\alpha}_k$ is the hyperparameter of the Dirichlet distribution and $\boldsymbol{\alpha}_0 = \sum_{k=1}^{K} \boldsymbol{\alpha}_k$ is the Dirichlet strength. The output of a standard neural network classifier is a probability assignment over the possible classes and the Dirichlet distribution represents the probability of each such probability assignment, hence modelling second-order probabilities and uncertainty. The modelling process is illustrated in Fig. 5.3.

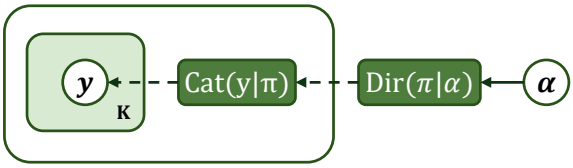


Fig. 5.3 Illustration of the Dirichlet process.

Subjective logic (Jsang, 2018) establishes a connection between the Dirichlet distribution and the belief representation in Dempster–Shafer belief theory (Dempster, 1968), also known as evidence theory. Consider K classes each associated with a belief mass b_k and an overall uncertainty mass u, which satisfies:

$$u + \sum_{k=1}^{K} b_k = 1 \tag{5.4}$$

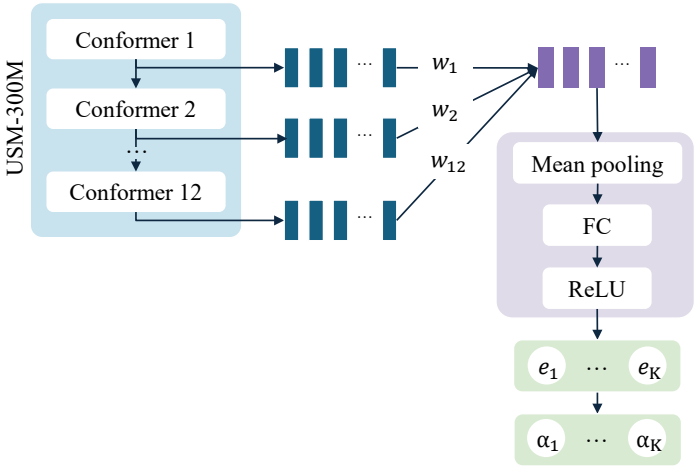


Fig. 5.4 Illustration of the model structure for quantifying uncertainty in emotion classification by evidential deep learning.

The belief mass assignment corresponds to the Dirichlet hyperparameter $\boldsymbol{\alpha}_k$:

$$b_k = \frac{\boldsymbol{\alpha}_k - 1}{\boldsymbol{\alpha}_0}, \tag{5.5}$$

where $e_k = \boldsymbol{\alpha}_k - 1$ is usually termed evidence. The overall uncertainty can then be computed as:

$$u = \frac{K}{\boldsymbol{\alpha}_0}. \tag{5.6}$$

A neural network $\boldsymbol{f}_{\boldsymbol{\Lambda}}$ is trained to predict $\mathrm{Dir}(\boldsymbol{\pi}_i|\boldsymbol{\alpha}_i)$ for a given sample $\boldsymbol{x}_i$ where $\boldsymbol{\Lambda}$ is the model parameters. The network is similar to standard neural networks for classification except that the softmax output layer is replaced with a ReLU activation layer to assure non-negative outputs, which is taken as the evidence vector for the predicted Dirichlet distribution: $\boldsymbol{f}_{\boldsymbol{\Lambda}}(\boldsymbol{x}_i) = \boldsymbol{e}_i$. The concentration parameter of the Dirichlet distribution can be calculated as $\boldsymbol{\alpha}_i = \boldsymbol{f}_{\boldsymbol{\Lambda}}(\boldsymbol{x}_i) + 1$. Given $\mathrm{Dir}(\boldsymbol{\pi}_i|\boldsymbol{\alpha}_i)$, the estimated probability of class k can be calculated by:

$$\mathbb{E}[\boldsymbol{\pi}_{ik}] = \frac{\boldsymbol{\alpha}_{ik}}{\boldsymbol{\alpha}_{0i}}. \tag{5.7}$$

Training

For brevity, superscript i is omitted in this section. Given one-hot label $\boldsymbol{y}$ and predicted Dirichlet $\mathrm{Dir}(\boldsymbol{\pi}|\boldsymbol{\alpha})$, the network can be trained by maximising the marginal likelihood of sampling $\boldsymbol{y}$ given the Dirichlet prior. Since the Dirichlet distribution is the conjugate prior of

the categorical distribution, the marginal likelihood is tractable:

$$\begin{aligned} \mathrm{P}(\boldsymbol{y}|\boldsymbol{\alpha}) &= \int \mathrm{P}(\boldsymbol{y}|\boldsymbol{\pi})\mathrm{p}(\boldsymbol{\pi}|\boldsymbol{\alpha})\mathrm{d}\boldsymbol{\pi} = \int \prod_k \boldsymbol{\pi}_k^{\boldsymbol{y}_k} \frac{1}{\mathrm{B}(\boldsymbol{\alpha})} \prod_k \boldsymbol{\pi}_k^{\boldsymbol{\alpha}_k - 1} = \frac{\mathrm{B}(\boldsymbol{\alpha} + \boldsymbol{y})}{\mathrm{B}(\boldsymbol{\alpha})} \\ &= \frac{\Gamma(\sum_k \boldsymbol{\alpha}_k)}{\Gamma(\sum_k \boldsymbol{\alpha}_k + \sum_k \boldsymbol{y}_k)} \frac{\prod_k \Gamma(\boldsymbol{\alpha}_k + \boldsymbol{y}_k)}{\prod_k \Gamma(\boldsymbol{\alpha}_k)} = \frac{\prod_{k=1}^K \boldsymbol{\alpha}_k^{\boldsymbol{y}_k}}{\boldsymbol{\alpha}_0^{\sum_{k=1}^K \boldsymbol{y}_k}}. \end{aligned} \tag{5.8}$$

It is equivalent to training the model by minimising the negative log marginal likelihood:

$$\mathcal{L}^{\mathrm{NLL}} = \sum_{k=1}^{K} \boldsymbol{y}_k (\log(\boldsymbol{\alpha}_0) - \log(\boldsymbol{\alpha}_k)). \tag{5.9}$$

Following Sensoy et al. (2018), a regularisation term is added to penalise the misleading evidence:

$$\mathcal{L}^{\mathrm{R}} = \mathcal{KL}(\mathrm{Dir}(\boldsymbol{\pi}|\tilde{\boldsymbol{\alpha}})||\,\mathrm{Dir}(\boldsymbol{\pi}|\mathbf{1})), \tag{5.10}$$

where $\mathrm{Dir}(\boldsymbol{\pi}|\mathbf{1})$ denotes a Dirichlet distribution with zero total evidence and $\tilde{\boldsymbol{\alpha}} = \boldsymbol{y} + (1 - \boldsymbol{y}) \odot \boldsymbol{\alpha}$[4] is the Dirichlet parameters after removal of the non-misleading evidence from predicted $\boldsymbol{\alpha}$. This penalty explicitly enforces the total evidence to shrink to zero for a sample if it cannot be correctly classified. The overall loss is $\mathcal{L} = \mathcal{L}^{\mathrm{NLL}} + \lambda\mathcal{L}^{\mathrm{R}}$ where λ is the regularisation coefficient, which was set to 0.8 for IEMOCAP and 0.2 for CREMA-D.

5.3.3 Evaluation Metrics

The proposed method is evaluated in terms of majority prediction, uncertainty estimation, and OOD detection.

Majority Prediction

Majority prediction for MA utterances is evaluated by classification accuracy (ACC) and unweighted average recall (UAR) which is the sum of class-wise accuracy divided by the number of classes.

Uncertainty Estimation

Model calibration is evaluated by expected calibration error (ECE) (Naeini et al., 2015) and maximum calibration error (MCE) (Naeini et al., 2015). ECE measures model calibration by computing the difference in expectation between confidence and accuracy. The expectation

[4] $\odot$ denotes element-wise multiplication.

is approximated by partitioning predictions into Q bins equally spaced in the [0,1] range based on predicted confidence. ECE can then be computed by taking a weighted average of the bins' accuracy/confidence difference:

$$\mathrm{ECE} = \sum_{q=1}^{Q} \frac{|B_q|}{N} \left|\mathrm{Acc}(B_q) - \mathrm{Conf}(B_q)\right|. \tag{5.11}$$

where B_q is the samples in the q^{th} bin. $|B_q|$ denotes the number of sample in the q^{th} bin and N denotes the total number of samples. A smaller ECE value indicates better model calibration. MCE is a variation of ECE which measures the largest calibration gap:

$$\mathrm{MCE} = \max_{q \in \{1,\ldots,Q\}} \left|\mathrm{Acc}(B_q) - \mathrm{Conf}(B_q)\right|. \tag{5.12}$$

OOD Detection

The area under the receiver operating characteristic (AUROC) and the area under the precision-recall curve (AUPRC) are used to evaluate the performance of OOD detection.

AUROC is calculated as the area under the ROC curve. Samples are classified as positive based on different threshold values (*e.g.*, predicted probability) and true positive rate (TPR) and false positive rate (FPR) are computed at each threshold. TPR, also known as recall, is the ratio of correctly predicted positive observations to all actual positives. FPR is the ratio of incorrectly predicted positive observations to all actual negatives. An ROC curve plots FPR against TPR at different thresholds and the AUROC is a single scalar value that summarises the performance across all thresholds. Similarly, AUPRC is calculated as the area under the precision-recall curve where precision computes the ratio of true positive predictions to the total number of positive predictions. The estimated uncertainty is used as the decision threshold for both AUROC and AUPRC. The baseline is 50% for AUROC and is the fraction of positives for AUPRC. NMA utterances are set as the positive class to detect.

5.3.4 Experiments: Detecting NMA as OOD

Baselines

The proposed methods were compared to the following baselines:

- MLE: a deterministic classification network with softmax activation trained by the cross-entropy loss between the majority vote label and model predictions. It is denoted as "MLE" since cross-entropy is essentially maximum likelihood estimation.

- MLE+: a MLE model with NMA as an extra class, which is the first approach described in Section 5.2.
- MCDP: a Monte Carlo dropout (Gal and Ghahramani, 2016) model with a dropout rate of 0.5 which is forwarded 100 times to obtain 100 samples during testing.
- Ensemble: an ensemble (Lakshminarayanan et al., 2017) of 10 MLE models with the same structure trained by bagging.

Uncertainty estimation of the EDL model is computed by Eqn. (5.6) while max probability is used as uncertainty measure for other methods.

Performance

The proposed EDL-based method is compared to baselines in Table 5.3 and Table 5.4 on the IEMOCAP and CREMA-D datasets respectively. For the first approach (denoted as "MLE+"), which trains an emotion classifier with NMA as an extra class, some of the NMA utterances are included in MLE+ training while the remainder are used for testing. Therefore, OOD detection is evaluated only on NMA (test) data for MLE+.

First, as shown by the values of ACC and UAR, the proposed method demonstrates comparable classification performance to the baselines, suggesting that the extension to uncertainty estimation does not undermine the model's capabilities. Although the Ensemble achieves the highest accuracy on CREMA-D, it involves training 10 individual systems. The proposed method achieves overall the best classification performance with only a tenth of the computational cost of Ensemble during both training and testing. In addition, the proposed method offers superior model calibration, as shown by the lowest values of ECE and MCE.

Table 5.3 Results of quantifying uncertainty in emotion classification on the IEMOCAP dataset. The baseline for AUPRC is 0.433 for the entire NMA set and 0.160 for the NMA test subset. The best value in each column is indicated in bold, and the second-best value is underlined.

	Classify MA				**Detect NMA (all)**		**Detect NMA (test)**	
	ACC↑	**UAR↑**	**ECE↓**	**MCE↓**	**AUROC↑**	**AUPRC↑**	**AUROC↑**	**AUPRC↑**
MLE+	0.447	0.438	0.303	0.383	/	/	0.461	0.139
MLE	0.582	0.577	0.206	0.239	0.550	0.471	0.549	0.177
MCDP	0.584	0.572	<u>0.128</u>	<u>0.184</u>	0.566	<u>0.491</u>	<u>0.568</u>	<u>0.203</u>
Ensemble	<u>0.593</u>	<u>0.595</u>	0.439	0.594	<u>0.567</u>	<u>0.491</u>	0.563	0.192
EDL	**0.611**	**0.596**	**0.103**	**0.145**	**0.610**	**0.530**	**0.620**	**0.227**

Table 5.4 Results of quantifying uncertainty in emotion classification on the CREMA-D dataset. The baseline for AUPRC is 0.387 for the entire NMA set and 0.097 for the NMA test subset.

	Classify MA				Detect NMA (all)		Detect NMA (test)	
	ACC↑	**UAR↑**	**ECE↓**	**MCE↓**	**AUROC↑**	**AUPRC↑**	**AUROC↑**	**AUPRC↑**
MLE+	0.568	0.540	0.216	0.476	/	/	0.552	0.156
MLE	0.714	0.672	0.150	0.156	0.578	0.467	0.571	0.179
MCDP	0.717	0.687	0.102	0.109	0.619	0.481	0.614	0.201
Ensemble	**0.731**	0.674	0.362	0.496	0.598	0.481	0.605	0.198
EDL	0.711	**0.714**	**0.057**	**0.080**	**0.645**	**0.506**	**0.657**	**0.234**

It also outperforms the baselines in effectively identifying NMA as OOD samples, as shown by the highest AUROC and AUPRC values.

Reject Option for Accuracy

This section performs a reject option where the system has a option to reject a test sample based on predicted uncertainty. Fig. 5.5 shows the change of accuracy when samples with uncertainty larger than a threshold are excluded. The model tends to provide less accurate predictions when it is less confident about its prediction, shown by the decrease of classification accuracy when the uncertainty threshold increases, which demonstrates the effectiveness of uncertainty prediction.

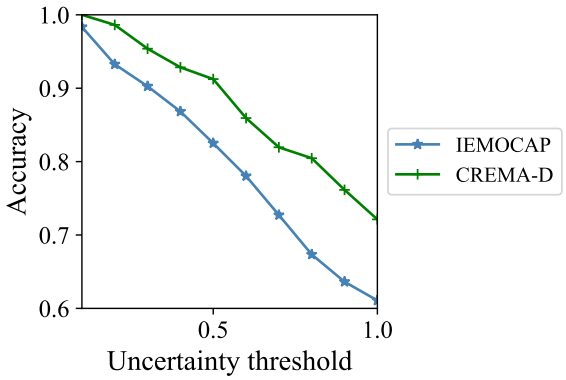


Fig. 5.5 The change of accuracy with respect to the uncertainty threshold for EDL-based methods on IEMOCAP and CREMA-D.

Table 5.5 Comparison of EDL methods with different activation functions on IEMOCAP and CREMA-D.

IEMOCAP	Classify MA				Detect NMA (all)		Detect NMA (test)	
	ACC	UAR	ECE	MCE	AUROC	AUPRC	AUROC	AUPRC
EDL (ReLU)	**0.611**	0.596	0.103	**0.145**	0.610	0.530	0.620	0.227
EDL (Softplus)	0.608	0.574	**0.035**	0.173	**0.617**	**0.534**	**0.639**	**0.251**
EDL (Exponential)	0.588	**0.601**	0.167	0.230	0.593	0.502	0.619	0.225
CREMA-D	**Classify MA**				**Detect NMA (all)**		**Detect NMA (test)**	
	ACC	UAR	ECE	MCE	AUROC	AUPRC	AUROC	AUPRC
EDL (ReLU)	0.701	**0.714**	**0.057**	**0.080**	**0.645**	**0.506**	**0.657**	**0.234**
EDL (Softplus)	0.692	0.696	0.113	0.309	0.640	**0.506**	0.633	0.230
EDL (Exponential)	**0.723**	0.602	0.277	0.277	0.623	0.495	0.626	0.197

5.3.5 Analysis

Analysis of Alternative Activation Functions

As described in Section 5.3.2, ReLU is used as the output activation function in EDL to ensure the evidence is non-negative. This section compares the use of different activation functions including ReLU, softplus and exponential functions. The three activation functions are plotted in Fig. 5.6.

As shown in Table 5.5, using exponential function tends to result in less effective model calibration, shown by the largest ECE and MCE values. It also produces poorer performance for NMA detection, shown by the smallest AUROC and AUPRC. Recall that AU-

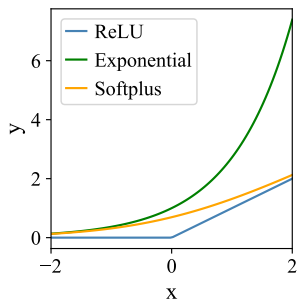
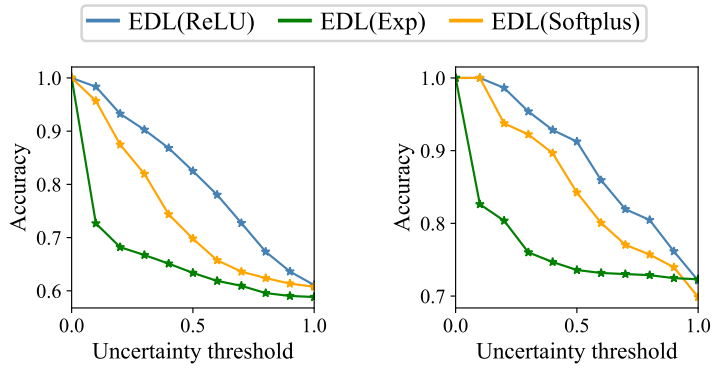
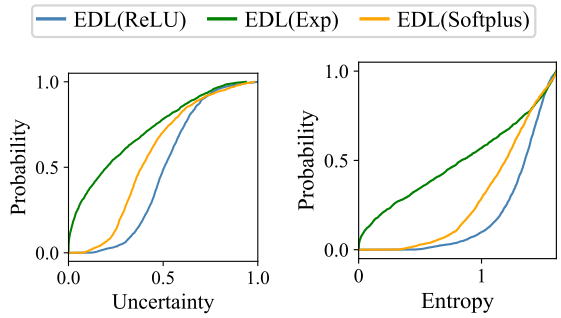


Fig. 5.6 Illustration of the activation functions.



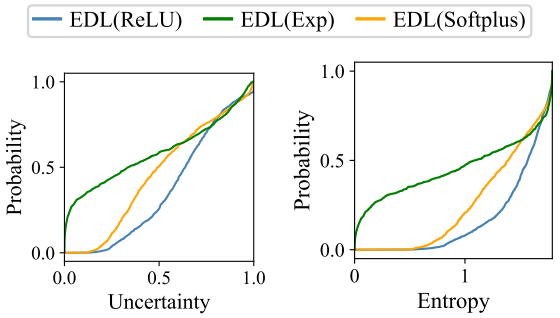
(a) IEMOCAP (b) CREMA-D

Fig. 5.7 Reject option for accuracy for EDL methods with different activation functions.



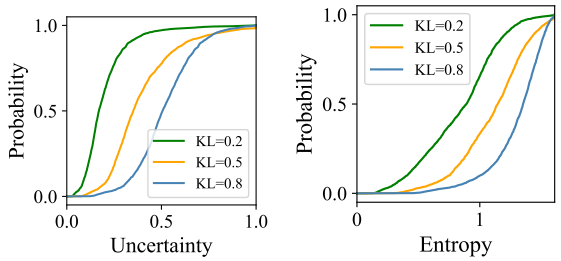
(a) ECDF of uncertainty (b) ECDF of entropy

Fig. 5.8 ECDF of uncertainty (left) and entropy (right) on IEMOCAP for EDL method with different activation functions.



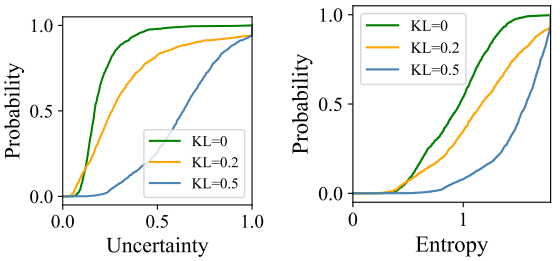
(a) ECDF of uncertainty (b) ECDF of entropy

Fig. 5.9 ECDF of uncertainty (left) and entropy (right) on CREMA-D for EDL method with different activation functions.



(a) ECDF of uncertainty (b) ECDF of entropy

Fig. 5.10 ECDF of uncertainty (left) and entropy (right) on IEMOCAP for EDL method with different regularisation coefficient λ.



(a) ECDF of uncertainty (b) ECDF of entropy

Fig. 5.11 ECDF of uncertainty (left) and entropy (right) on CREMA-D for EDL method with different regularisation coefficient λ.

ROC/AUPRC measures the performance of classifying samples as NMA based on the predicted uncertainty at different uncertainty thresholds.

Fig. 5.7 shows the reject option for accuracy of EDL with different activation functions. A drop in accuracy when the uncertainty threshold increases from 0 to 0.1 is observed for model using the exponential activation. This indicates that exponential activation tends to lead to smaller uncertainty values.

The empirical cumulative distribution function (ECDF) of uncertainty and entropy on IEMOCAP and CREMA-D are plotted in Fig. 5.8 and Fig. 5.9 respectively. It can be seen that exponential activation leads to smaller uncertainty and entropy, which echos the statement in Section 5.3.1 that exponential activation tends to inflate the probability of the correct class.

Analysis of Regularisation Coefficient

This section analyses the effect of regularisation coefficient λ in Eqn. (5.10) for EDL. The empirical CDF of uncertainty and entropy when different regularisation coefficients were used is plotted in in Fig. 5.10 and Fig. 5.11 for IEMOCAP and CREMA-D respectively. It is observed that larger λ values lead to a larger entropy and uncertainty. This aligns with the definition of the regularisation term in Eqn. (5.10) which tends to enforce a flat prior with small evidence.

5.4 Emotion Distribution Estimation

Consider the example shown in Fig. 5.12 with the annotations assigned to three utterances. Since the majority emotion classes are "angry" for both utterances (a) and (b), they will be assigned the same ground-truth label "angry" in the aforementioned classification system, which implies that they convey the same emotion content and is clearly unsuitable. On the contrary, utterance (c), though being an NMA utterance, is more likely to share similar emotional content with utterance (b). Therefore, in order to obtain more comprehensive representations of emotion content, we further propose representing emotion as a distribution rather than a single class label and re-framing emotion recognition as a distribution estimation problem rather than a classification problem. A novel algorithm is proposed which extends EDL to estimate the underlying emotion distribution given observed human annotations and quantify the uncertainty in emotion distribution estimation. In this approach, the system is trained to maximise the marginal likelihood of observing all human annotations from a multinomial distribution under the Dirichlet prior. All human annotations are considered rather than relying solely on the majority vote class. Instead of simply saying "I don't know", the proposed system demonstrates the ability to estimate the emotion distributions of the NMA utterances and also offer a reliable uncertainty measure for the distribution estimation.

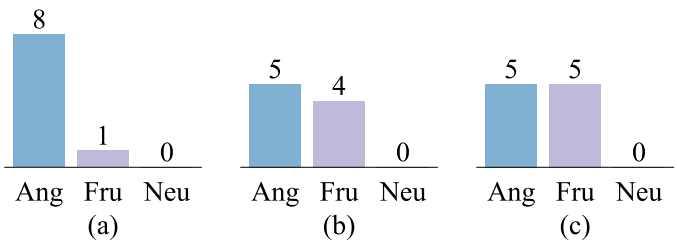


Fig. 5.12 The bar chart shows the number of labels assigned by annotators to the emotion class "**Ang**ry", "**Fru**strated", and "**Neu**tral" in an example. In utterance (a), eight annotators interpret the emotion as angry while one interprets it as frustrated.

5.4.1 Representing Emotion as a Distribution

Consider an input utterance $\boldsymbol{x}_i$ associated with M_i labels from human annotators $\mathcal{D}_i = \{\boldsymbol{y}_i^{(m)}\}_{m=1}^{M_i}$ where $\boldsymbol{y}_i^{(m)} = [\boldsymbol{y}_{i1}^{(m)}, \ldots, \boldsymbol{y}_{iK}^{(m)}]$ is a one-hot vector. Instead of representing the emotion content by the majority vote class, the underlying emotion distribution $\boldsymbol{\pi}$ is estimated based on the observations $\{\boldsymbol{y}_i^{(m)}\}_{m=1}^{M_i}$. The emotion classification problem is thus re-framed as a distribution estimation problem. In contrast to the "soft label" method mentioned in Section 3.5.1 which approximates the emotion distribution of each $\boldsymbol{x}_i$ solely based on $\mathcal{D}_i$ by MLE and trains the model to learn this proxy in a supervised manner, the proposed approach trains a distribution estimator $\boldsymbol{f}_{\boldsymbol{\Lambda}}$ across all data points $\{\boldsymbol{x}_i, \mathcal{D}_i\}_{i=1}^{N}$ where N is the number of utterances in training. This approach leverages the knowledge about the emotion expression and annotation variability across different utterances,

5.4.2 Extending EDL for Distribution Estimation

For brevity, the superscript i is omitted in this section. Assume $\{\boldsymbol{y}^{(m)}\}_{m=1}^{M}$ are samples drawn from a multinomial distribution. Let $\hat{\boldsymbol{y}} = \sum_{m=1}^{M} \boldsymbol{y}_m$ represents the counts of each emotion class:

$$\{\boldsymbol{y}^{(m)}\}_{m=1}^{M} \sim \mathrm{P}(\boldsymbol{y}|\boldsymbol{\pi}) = \mathrm{Mult}(\boldsymbol{\pi}, M) \tag{5.13}$$

$$\mathrm{Mult}(\boldsymbol{\pi}, M) = \frac{\Gamma(M+1)}{\prod_{k=1}^{K}\Gamma(\hat{\boldsymbol{y}}_k+1)}\boldsymbol{\pi}_k^{\hat{\boldsymbol{y}}_k}. \tag{5.14}$$

The categorical distribution in Eqn. (5.1) is the special case when $M = 1$.

The network is trained by maximising the marginal likelihood of sampling $\{\boldsymbol{y}^{(m)}\}_{m=1}^{M}$ given the predicted Dirichlet prior $\mathrm{Dir}(\boldsymbol{\pi}|\boldsymbol{\alpha})$:

$$\begin{aligned}
\mathrm{P}(\{\boldsymbol{y}^{(m)}\}_{m=1}^{M}|\boldsymbol{\alpha}) &= \int \mathrm{P}(\{\boldsymbol{y}^{(m)}\}_{m=1}^{M}|\boldsymbol{\pi})\mathrm{p}(\boldsymbol{\pi}|\boldsymbol{\alpha})d\boldsymbol{\pi} \\
&= \int \frac{\Gamma(M+1)}{\prod_{k=1}^{K}\Gamma(\hat{\boldsymbol{y}}_k+1)}\prod_k \boldsymbol{\pi}_k^{\hat{\boldsymbol{y}}_k}\frac{1}{B(\boldsymbol{\alpha})}\prod_k \boldsymbol{\pi}_k^{\boldsymbol{\alpha}_k-1}d\boldsymbol{\pi} \\
&= \frac{\Gamma(M+1)}{\prod_{k=1}^{K}\Gamma(\hat{\boldsymbol{y}}_k+1)}\frac{B(\boldsymbol{\alpha}+\boldsymbol{y})}{B(\boldsymbol{\alpha})} \\
&= \underbrace{\frac{\Gamma(M+1)}{\prod_{k=1}^{K}\Gamma(\hat{\boldsymbol{y}}_k+1)}}_{\text{Part A}}\underbrace{\frac{\prod_{k=1}^{K}\boldsymbol{\alpha}_k^{\hat{\boldsymbol{y}}_k}}{\boldsymbol{\alpha}_0^{\sum_{k=1}^{K}\hat{\boldsymbol{y}}_k}}}_{\text{Part B}}.
\end{aligned} \tag{5.15}$$

The multinomial coefficient (Part A in Eqn. (5.15)) is independent of $\boldsymbol{\alpha}$ and Part B in Eqn. (5.15) is the same as Eqn. (5.8) except for replacing one-hot majority label $\boldsymbol{y}$ with $\hat{\boldsymbol{y}}$. It is thus verified that $\mathcal{L}^{\text{NLL}}$ in Eqn. (5.9) can be generalised to the distribution estimation framework by replacing one-hot majority label $\boldsymbol{y}$ with $\hat{\boldsymbol{y}}$:

$$\mathcal{L}^{\text{NLL}^*} = \sum_{k=1}^{K} \hat{\boldsymbol{y}}_k (\log(\boldsymbol{\alpha}_0) - \log(\boldsymbol{\alpha}_k)). \tag{5.16}$$

The regulariser in Eqn. (5.10) is replaced with:

$$\mathcal{L}^{\text{R1}} = \mathcal{KL}(\text{Dir}(\boldsymbol{\pi}|\hat{\boldsymbol{\alpha}})||\,\text{Dir}(\boldsymbol{\pi}|\mathbf{1})) \tag{5.17}$$

where $\hat{\boldsymbol{\alpha}} = \bar{\boldsymbol{y}} + (1 - \bar{\boldsymbol{y}}) \odot \boldsymbol{\alpha}$[5] and $\bar{\boldsymbol{y}} = \frac{1}{M} \sum_{m=1}^{M} \boldsymbol{y}_m$ is the soft label. An alternative regulariser is proposed in order to explicitly regularise the predicted multinomial distribution:

$$\mathcal{L}^{\text{R2}} = \mathcal{KL}(\bar{\boldsymbol{y}}||\mathbb{E}[\boldsymbol{\pi}]). \tag{5.18}$$

Hence, the EDL method described in Section 5.3.2 for classification has been extended to quantify the uncertainty in distribution estimation, with the original method (Sensoy et al., 2018) being a special case when $M = 1$ and $\hat{\boldsymbol{y}}$ becomes the one-hot majority label $\boldsymbol{y}$. In addition, it is worth noting that the proposed approach does not require a fixed number of annotators for every utterance and can easily generalise to a large number of annotators (*i.e.*, for crowd-sourced datasets).

5.4.3 Further Evaluation Metrics and Baselines

In addition to the metrics introduced in Section 5.3.3 for majority prediction (ACC, UAR) and uncertainty estimation (ECE, MCE), an additional metric is adopted to evaluate the performance of emotion distribution estimation: negative log likelihood (NLL) of sampling human annotations from the predicted emotion distribution.

An additional baseline is also adopted for distribution estimation, which is the "soft label" approach mentioned in Section 3.5.1. It is trained by minimising KL divergence between the soft label $\bar{\boldsymbol{y}}$ and predictions, which is denoted as **"MLE*"** since it is an extension of MLE from one-hot majority vote labels to soft labels.

The systems proposed in Section 5.4 using regularisation terms defined in Eqn. (5.17) and Eqn. (5.18) are denoted as **"EDL*(R1)"** and **"EDL*(R2)"** respectively.

[5] $\odot$ denotes element-wise multiplication.

Table 5.6 Classification and calibration performance of distribution-based methods on MA data. The best value in each column is indicated in bold.

	IEMOCAP				CREMA-D			
	ACC↑	**UAR↑**	**ECE↓**	**MCE↓**	**ACC↑**	**UAR↑**	**ECE↓**	**MCE↓**
MLE*	0.564	0.562	0.151	0.279	0.693	0.621	0.109	0.115
EDL*(R1)	0.623	0.612	0.081	0.208	**0.740**	0.694	**0.029**	**0.095**
EDL*(R2)	**0.624**	**0.616**	**0.025**	**0.201**	0.718	**0.722**	0.084	0.107

Table 5.7 Emotion distribution estimation results on IEMOCAP and CREMA-D. NMA stands for NMA (all).

	IEMOCAP		CREMA-D	
	NLL^{MA} ↓	**NLL^{NMA} ↓**	**NLL^{MA} ↓**	**NLL^{NMA} ↓**
MLE	1.310	1.924	1.532	2.054
MCDP	0.972	1.266	0.965	1.292
Ensemble	2.572	2.055	2.285	2.089
EDL	0.958	1.019	0.757	1.021
MLE*	0.941	1.137	0.648	0.774
EDL*(R1)	0.861	**0.951**	0.614	0.722
EDL*(R2)	**0.833**	0.953	**0.606**	**0.698**

5.4.4 Experiments: Estimating Emotion Distribution

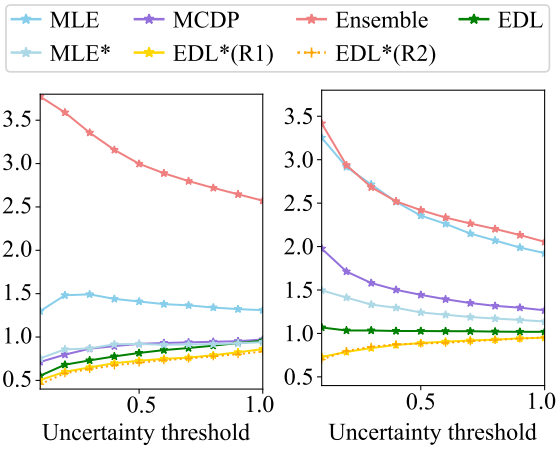
The proposed EDL* methods were first evaluated in terms of majority class prediction. The results of distribution-based methods on classification of MA data are shown in Table 5.6. Compared to the classification-based methods in Table 5.3 and Table 5.4, it can be seen that EDL* does not reduce the performance of emotion classification (in terms of ACC and UAR) and model calibration (in terms of ECE and NCE) on MA data. This indicates that the information about the majority class is retained when representing emotion as a distribution. Note that when representing emotion as a distribution, it is no longer appropriate to consider NMA utterances as OOD samples, as illustrated by case (b) and (c) in Fig. 5.12. Although still trained only on MA data, the proposed distribution-based system shows good generalisation ability in predicting the emotion distribution of NMA data, which we will see shortly. When encountering NMA data during testing, instead of simply returning "I don't know", the proposed system can provide reliable estimation of its emotional content, which is a key benefit.

The proposed EDL* methods were then evaluated regarding distribution estimation. Table 5.7 compares EDL* to the baselines in terms of the negative log likelihood of sampling

target labels from the predicted emotion distribution. As can be seen from the table, EDL* methods produce improved distribution estimation, achieving the smallest NLL values on both MA and NMA data. Among the two EDL* methods employing different regularisation terms, EDL* with R2 (defined in Eqn. (5.18)), which directly applies regularisation to the predicted distribution, exhibits better distribution estimation without sacrificing model calibration.

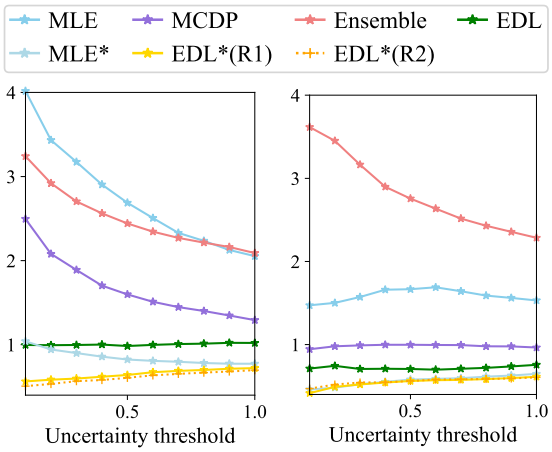
Reject Option for NLL

A reject option was then evaluated for NLL (instead of accuracy) to examine model calibration. For a well-calibrated model, an increase in the NLL value, which is associated with poorer distribution estimation, is expected when the model becomes less confident. Fig. 5.13 and Fig. 5.14 visualise the change of NLL for MA data and NMA data when uncertainty increases for IEMOCAP and CREMA-D respectively. For MA data, *i.e.,* the type of data that has been seen by the models during training, most methods can successfully reject uncertain samples except for MLE and Ensemble, as shown by an increase in NLL values when the uncertainty threshold increases. However, for NMA data which the model hasn't seen in training, only the EDL* methods exhibit the ability to demonstrate an increasing trend in NLL values.



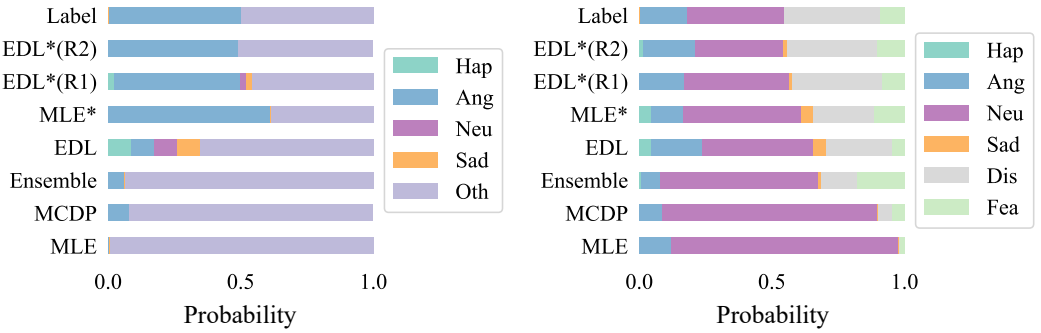
(a) MA data (b) NMA data

Fig. 5.13 Reject option for NLL on IEMOCAP.



(a) MA data (b) NMA data

Fig. 5.14 Reject option for NLL on CREMA-D.



(a) "Ses04M_impro02_F024" (b) "1084_TSI_ANG_XX"

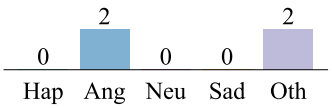
Fig. 5.15 Visualisation of emotion distribution for case study. Utterance (a) from IEMOCAP. Utterance (b) from CREMA-D.

5.4.5 Case Study

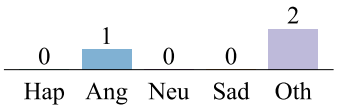
Emotion distributions estimated by different methods are visualised against the label distributions for two representative examples in Fig. 5.15. In general, distribution-based methods show superior performance for distribution estimation than classification-based methods. In the case of utterance (a) which received two "angry" labels and two "frustrated" labels, the proposed EDL* methods stand out by effectively capturing the tie between the emotions, whereas the predictions of classification-based methods tend to be predominantly skewed towards "frustrated". As for utterance (b), where both "disgust" and "neutral" receive four votes, along with two votes for "angry" and one for "fear", the emotion distributions predicted by the EDL* methods also show a similar pattern. These examples show that the proposed method can not only provide a more comprehensive emotion representation but also better reflect the variability of human opinions. Additional examples can be found in Appendix C.

The proficiency of the proposed EDL* methods for estimating the emotion distribution and providing reliable confidence predictions, demonstrates the method's capacity to estimate both aleatoric uncertainty (Matthies, 2007, Der Kiureghian and Ditlevsen, 2009), arising from data complexity (*i.e.*, the ambiguity of emotion expression), and epistemic uncertainty (Der Kiureghian and Ditlevsen, 2009)[6], corresponding to the amount of uncommitted belief in subjective logic.

[6]These two type of uncertainty will be discussed in detail in Chapter 6.



(a) "Ses04M_impro02_F024"



(b) "Ses05M_impro01_M014"

Fig. 5.16 Human annotations for (a) NMA utterance "Ses04M_impro02_F024" and (b) MA utterance "Ses05M_impro01_M014".

5.5 Analysis: When OOD Detection Fails

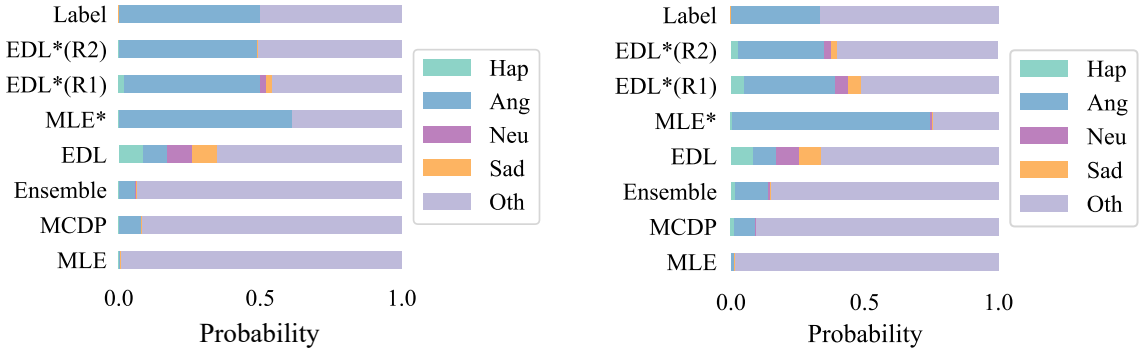
This section includes examples and analysis of particular utterances where OOD detection is problematic and shows how distribution-based methods improve over the classification-based systems in handling complex ambiguous emotions.

5.5.1 A False Negative Case

Consider the following false negative case where the OOD detection model fails to detect an NMA sample. Utterance "Ses04M_impro02_F024" from the IEMOCAP dataset has two "angry" labels and two "frustrated" labels as shown in Fig. 5.16 (a). The EDL system predicts this utterance as "frustrated" with a belief mass of 0.567 and an overall uncertainty score of 0.433, which reveals that the system fails to detect the utterance as NMA.

A possible cause of this failure is that the model gets confused by MA utterances seen in the training that convey similar emotional content, such as "Ses05M_impro01_M014" whose annotations are shown in Fig. 5.16 (b) with a MA emotion class "frustrated". Although one annotator considered it as "angry", the MA ground-truth target was "frustrated" in a classification-based system. Both utterances occur within a dyadic situation where two people disagree, with the speaker being the one who compromises, feeling unhappy and frustrated. Such similar emotional content may confuse a classification-based system to also predict the NMA utterance as frustrated. It is worth noting that data with the same distribution as "Ses04M_impro02_F024", which has tied votes, is not included during the training of a classification-based model because there is no majority vote available to serve as ground truth.

This complex emotional expression can be better described by the distribution-based EDL* systems. The predicted distribution of the NMA and MA utterances are shown in Fig. 5.17. It can be seen that the classification-based methods produce a similar distribution for the two utterances, with "frustrated" being dominant. However, the proposed EDL* methods can better match the label distribution and distinguish between these two cases. Although not been trained on NMA data, the EDL* methods are still capable of providing



(a) "Ses04M_impro02_F024" (b) "Ses05M_impro01_M014"

Fig. 5.17 Predicted emotion distribution of (a) NMA utterance "Ses04M_impro02_F024" and (b) MA utterance "Ses05M_impro01_M014".

accurate predictions of its emotional content. This is a key benefit of the distribution-based approaches.

5.5.2 A False Positive Case

Next, a typical false positive instance is provided where a MA utterance is mis-detected as OOD. The MA utterance "1087_IEO_FEA_LO" from the CREMA-D dataset has four "neutral", two "sad", and three "fear" human labels as in Fig. 5.18 (a). The NMA utterance "1052_ITH_FEA_XX" has four "neutral", two "sad", and four "fear" human labels as in Fig. 5.18 (b). The OOD system successfully predicts the NMA utterance as OOD with an overall uncertainty of 0.691 while also predicting the MA utterance as an OOD sample with an overall uncertainty of 0.623.[7] This failure is possible because the MA utterance "1087_IEO_FEA_LO" contains a complex mixture of emotions shown by the rather flat label distribution similar to "1052_ITH_FEA_XX", which confuses the OOD detection system. Note that the MA class "neutral" in Fig. 5.18 (a) comprises only $\frac{4}{4+2+3} \times 100\% = 44.4\%$

(a) "1087_IEO_FEA_LO" (b) "1052_ITH_FEA_XX"

Fig. 5.18 Human annotations for (a) MA utterance "1087_IEO_FEA_LO" and (b) NMA utterance "1052_ITH_FEA_XX".

[7] Assume the OOD detection threshold is taken as 0.5.

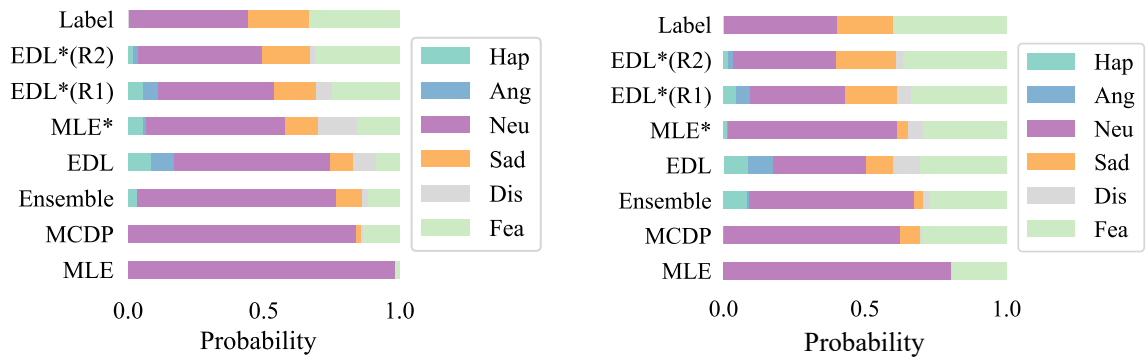
of the annotations and hence is not an absolute majority, which reduces the severity of this detection error.

Again, the distribution-based EDL* methods show superior capability in handling such complex case. The predicted distribution of the utterances are shown in Fig. 5.19. For the MA utterance, although human opinions diverge, the classification-based methods only capture the majority prediction, with the predicted distribution being dominated by "neutral". However, the emotion distribution predicted by the proposed EDL* methods retains the probability for "sad" and "fear" which accounts for the minority human opinions. Therefore, we show that the proposed EDL* method improves over the OOD system by providing a more comprehensive representation of emotional content as well as a more inclusive representation of human opinions.

5.6 Chapter Summary

In subjective tasks like emotion recognition, there is usually no single "correct" answer. The conventional approach of imposing a single ground truth through majority voting may overlook valuable nuances within each annotator's evaluation and the disagreements between them, potentially resulting in the under-representation of minority views. It also disregards a considerable amount of data. In this chapter, the emotion classification problem is re-examined, starting with an exploration of ways to handle data with ambiguous emotions.

It is first shown that incorporating ambiguous emotions as an extra class reduces the classification performance of the original emotion classes. Then, evidence theory is adopted to quantify uncertainty in emotion classification which allows the classifier to output "I don't



(a) "1087_IEO_FEA_LO" (b) "1052_ITH_FEA_XX"

Fig. 5.19 Predicted emotion distribution of (a) MA utterance "1087_IEO_FEA_LO" and (b) NMA utterance "1052_ITH_FEA_XX".

know" when it encounters utterances with ambiguous emotion. The model is trained to predict the hyperparameters of a Dirichlet distribution, which models the second-order probability of the probability assignment over emotion classes. Furthermore, to capture finer-grained emotion differences, the emotion classification problem is transformed into an emotion distribution estimation problem. All annotations are taken into account rather than only the majority opinion. A novel approach is proposed which extends standard EDL to quantify uncertainty in emotion distribution estimation. Experimental results show that given an utterance with ambiguous emotion the proposed approach is able to provide a comprehensive representation of its emotion content as a distribution with a reliable uncertainty measure.

Chapter 6

Estimating Uncertainty in Emotion Attributes

In AER, labels assigned by different human annotators to the same utterance are often inconsistent due to the inherent complexity of emotion and the subjectivity of perception. Although deterministic labels generated by averaging or voting are often used as the ground truth, it ignores the intrinsic uncertainty revealed by the inconsistent labels. Chapter 5 has discussed three approaches to handling inconsistency in categorical emotion annotations. This chapter extends the evidential deep learning approach to dimensional emotion attributes.

As discussed in Section 3.1, an emotional state can be defined based on either categorical or dimensional theory. The categorical theory claims the existence of a small number of basic discrete emotions that are inherent in our brain and universally recognised. Dimensional emotion theory characterises emotional states by a small number of roughly orthogonal fundamental continuous-valued bipolar dimensions, also known as emotion attributes, which allow us to model more subtle and complex emotions and are thus more common in psychological studies.

Two types of uncertainties commonly exist in AER: aleatoric uncertainty and epistemic uncertainty (Matthies, 2007, Der Kiureghian and Ditlevsen, 2009). The inconsistency in emotion annotations is a typical manifestation of aleatoric uncertainty, also referred to as data uncertainty, which arises from the intrinsic complexity of emotion data. Aggregating such inconsistent labels with deterministic labels obtained by majority voting (Busso et al., 2008, 2017) or (weighted) averages (Grimm and Kroschel, 2005, Ringeval et al., 2013, Lotfian and Busso, 2019, Kossaifi et al., 2019) ignores the discrepancies between annotators and the aleatoric uncertainty in emotion data. Epistemic uncertainty, also known as model uncertainty, is associated with uncertainty in model parameters that best explain the observed

data. Aleatoric and epistemic uncertainty are combined to induce the total uncertainty, also called predictive uncertainty, that measures the confidence of model predictions.

In this chapter, deep evidential emotion regression (DEER) is proposed to estimate these uncertainties in emotion attributes[1]. In contrast to Bayesian neural networks that place priors on model parameters (Blundell et al., 2015, Kendall and Gal, 2017), evidential deep learning (Sensoy et al., 2018, Malinin and Gales, 2018, Amini et al., 2020) places priors over the likelihood function. Every training sample adds support to a learned higher-order prior distribution called the evidential distribution. Sampling from this distribution gives instances of lower-order likelihood functions from which the data was drawn.

In DEER, the inconsistent human labels of each utterance are considered as observations drawn independently from an unknown Gaussian distribution. To probabilistically estimate the mean and variance of the Gaussian distribution, a normal-inverse-gamma (NIG) prior is introduced, which places a Gaussian prior over the mean and an inverse-gamma prior over the variance. The AER system is trained to predict the hyperparameters of the NIG prior for each utterance by maximising the per-observation-based marginal likelihood of each observed label under this prior. As a result, DEER enables a joint estimation of emotion attributes along with the aleatoric and epistemic uncertainties. As a further improvement, a novel regulariser is proposed based on the mean and variance of the observed labels to better calibrate the uncertainty estimation. The proposed methods were evaluated on the MSP-Podcast and IEMOCAP datasets. Experiments show that DEER produced SOTA results for both the mean values and the distribution of emotion attributes.

The rest of this chapter is organised as follows. Section 6.1 introduces the proposed DEER approach. Experimental setup and results are presented in Section 6.2 and Section 6.3 respectively. Section 6.4 provides analysis and discussion, followed by the chapter summary.

6.1 Deep Evidential Emotion Regression

6.1.1 Problem Setting

Consider an input utterance $\boldsymbol{x}$ with M emotion attribute labels $\boldsymbol{y}^{(1)}, \ldots, \boldsymbol{y}^{(M)}$ provided by multiple annotators. Assuming $\boldsymbol{y}^{(1)}, \ldots, \boldsymbol{y}^{(M)}$ are observations drawn i.i.d. from a Gaussian distribution with unknown mean $\boldsymbol{\mu}$ and unknown variance $\boldsymbol{\sigma}^2$, where $\boldsymbol{\mu}$ is drawn from a Gaussian prior and $\boldsymbol{\sigma}^2$ is drawn from an inverse-gamma prior:

$$\boldsymbol{y}^{(1)}, \ldots, \boldsymbol{y}^{(M)} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\sigma}^2)$$

[1] Part of this chapter has been published as a conference paper (Wu et al., 2023a). See Appendix A for more detail.

$$\boldsymbol{\mu} \sim \mathcal{N}(\gamma, \boldsymbol{\sigma}^2 \upsilon^{-1}), \quad \boldsymbol{\sigma}^2 \sim \Gamma^{-1}(\boldsymbol{\alpha}, \boldsymbol{\beta})$$

where $\gamma \in \mathbb{R}$, $\upsilon > 0$, and $\Gamma(\cdot)$ is the Gamma function with $\boldsymbol{\alpha} > 1$ and $\boldsymbol{\beta} > 0$.

Denote $\{\boldsymbol{\mu}, \boldsymbol{\sigma}^2\}$ and $\{\gamma, \upsilon, \boldsymbol{\alpha}, \boldsymbol{\beta}\}$ as Ψ and Ω. The posterior $p(\Psi|\Omega)$ is a NIG distribution, which is the Gaussian conjugate prior:

$$\begin{aligned} p(\Psi|\Omega) &= p(\boldsymbol{\mu}|\boldsymbol{\sigma}^2, \Omega)\, p(\boldsymbol{\sigma}^2|\Omega) = \mathcal{N}(\gamma, \boldsymbol{\sigma}^2 \upsilon^{-1})\, \Gamma^{-1}(\boldsymbol{\alpha}, \boldsymbol{\beta}) \\ &= \frac{\boldsymbol{\beta}^{\alpha}\sqrt{\upsilon}}{\Gamma(\boldsymbol{\alpha})\sqrt{2\pi\boldsymbol{\sigma}^2}} \left(\frac{1}{\boldsymbol{\sigma}^2}\right)^{\alpha+1} \exp\left\{-\frac{2\boldsymbol{\beta} + \upsilon(\gamma - \boldsymbol{\mu})^2}{2\boldsymbol{\sigma}^2}\right\} \end{aligned} \tag{6.1}$$

Drawing a sample Ψ_i from the NIG distribution yields a single instance of the likelihood function $\mathcal{N}(\boldsymbol{\mu}_i, \boldsymbol{\sigma}_i^2)$. The NIG distribution therefore serves as the higher-order, evidential distribution on top of the unknown lower-order likelihood distribution from which the observations are drawn. The NIG hyperparameters Ω determine not only the location but also the uncertainty associated with the inferred likelihood function.

By training a deep neural network model to output the hyperparameters of the evidential distribution, evidential deep learning allows the uncertainties to be found by analytic computation of the maximum likelihood Gaussian without the need for repeated inference for sampling (Amini et al., 2020). Furthermore, it also allows an effective estimate of the aleatoric uncertainty computed as the expectation of the variance of the Gaussian distribution, as well as the epistemic uncertainty defined as the variance of the predicted Gaussian mean. Given an NIG distribution, the prediction, aleatoric, and epistemic uncertainty can be computed as:

$$\text{Prediction: } \mathbb{E}[\boldsymbol{\mu}] = \gamma \tag{6.2}$$

$$\text{Aleatoric: } \mathbb{E}[\boldsymbol{\sigma}^2] = \frac{\boldsymbol{\beta}}{\boldsymbol{\alpha} - 1}, \quad \forall\, \boldsymbol{\alpha} > 1 \tag{6.3}$$

$$\text{Epistemic: } \mathrm{Var}[\boldsymbol{\mu}] = \frac{\boldsymbol{\beta}}{\upsilon(\boldsymbol{\alpha} - 1)}, \quad \forall\, \boldsymbol{\alpha} > 1 \tag{6.4}$$

6.1.2 Training

The training of DEER is structured as fitting the model to the data while enforcing the prior to calibrate the uncertainty when the prediction is wrong.

Maximising the Data Fit

The likelihood of an observation $\boldsymbol{y}$ given the evidential distribution hyperparameters Ω is computed by marginalising over the likelihood parameters Ψ:

$$\mathrm{p}(\boldsymbol{y}|\Omega) = \int \mathrm{p}(\boldsymbol{y}|\Psi)\mathrm{p}(\Psi|\Omega)\ \mathrm{d}\Psi = \mathbb{E}_{\mathrm{p}(\Psi|\Omega)}[\mathrm{p}(\boldsymbol{y}|\Psi)] \tag{6.5}$$

An analytical solution exists in the case of placing an NIG prior on the Gaussian likelihood function:

$$\begin{aligned}\mathrm{p}(\boldsymbol{y}|\Omega) &= \frac{\Gamma(1/2+\boldsymbol{\alpha})}{\Gamma(\boldsymbol{\alpha})}\sqrt{\frac{\boldsymbol{\upsilon}}{\pi}}\,(2\boldsymbol{\beta}(1+\boldsymbol{\upsilon}))^{\boldsymbol{\alpha}}\left(\boldsymbol{\upsilon}(\boldsymbol{y}-\boldsymbol{\gamma})^2+2\boldsymbol{\beta}(1+\boldsymbol{\upsilon})\right)^{-(\frac{1}{2}+\boldsymbol{\alpha})} \\ &= \mathrm{St}_{2\boldsymbol{\alpha}}\left(\boldsymbol{y}|\boldsymbol{\gamma}, \frac{\boldsymbol{\beta}(1+\boldsymbol{\upsilon})}{\boldsymbol{\upsilon}\,\boldsymbol{\alpha}}\right)\end{aligned} \tag{6.6}$$

where $\mathrm{St}_\nu\,(t|r,s)$ is Student's t-distribution evaluated at t with location parameter r, scale parameter s, and ν degrees of freedom. The predicted mean and variance can be computed analytically as

$$\mathbb{E}[\boldsymbol{y}] = \boldsymbol{\gamma}, \quad \mathrm{Var}[\boldsymbol{y}] = \frac{\boldsymbol{\beta}(1+\boldsymbol{\upsilon})}{\boldsymbol{\upsilon}(\boldsymbol{\alpha}-1)} \tag{6.7}$$

$\mathrm{Var}[\boldsymbol{y}]$ represents the total uncertainty of model prediction, which is equal to the summation of the aleatoric uncertainty $\mathbb{E}[\boldsymbol{\sigma}^2]$ and epistemic uncertainty $\mathrm{Var}[\boldsymbol{\mu}]$ according to the law of total variance:

$$\begin{aligned}\mathrm{Var}[\boldsymbol{y}] &= \mathbb{E}[\mathrm{Var}[\boldsymbol{y}|\Psi]] + \mathrm{Var}[\mathbb{E}[\boldsymbol{y}|\Psi]] \\ &= \mathbb{E}[\boldsymbol{\sigma}^2] + \mathrm{Var}[\boldsymbol{\mu}]\end{aligned} \tag{6.8}$$

To fit the NIG distribution, the model is trained by maximising the sum of the marginal likelihoods of each human label $\boldsymbol{y}^{(m)}$. The negative log likelihood (NLL) loss can be computed as

$$\mathcal{L}^{\mathrm{NLL}}(\boldsymbol{\theta}) = -\frac{1}{M}\sum_{m=1}^{M}\log \mathrm{p}(\boldsymbol{y}^{(m)}|\Omega) = -\frac{1}{M}\sum_{m=1}^{M}\log\left[\mathrm{St}_{2\boldsymbol{\alpha}}\left(\boldsymbol{y}^{(m)}|\boldsymbol{\gamma}, \frac{\boldsymbol{\beta}(1+\boldsymbol{\upsilon})}{\boldsymbol{\upsilon}\,\boldsymbol{\alpha}}\right)\right] \tag{6.9}$$

where $\boldsymbol{\theta}$ is the model parameters. This is the proposed per-observation-based NLL loss, which takes each observed label into consideration for AER. This loss serves as the first part of the objective function for training a deep neural network model with parameter $\boldsymbol{\theta}$ to predict the hyperparameters $\{\boldsymbol{\gamma}, \boldsymbol{\upsilon}, \boldsymbol{\alpha}, \boldsymbol{\beta}\}$ to fit all observed labels of $\boldsymbol{x}$.

Calibrating the Uncertainty on Errors

The second part of the objective function regularises training by calibrating the uncertainty based on the incorrect predictions. A novel regulariser is formulated which contains two terms: $\mathcal{L}^{\mu}$ and $\mathcal{L}^{\sigma}$ that respectively regularises the errors on the estimation of the mean $\boldsymbol{\mu}$ and the variance $\boldsymbol{\sigma}^2$ of the Gaussian likelihood.

The first term $\mathcal{L}^{\mu}$ is proportional to the error between the model prediction and the average of the observations:

$$\mathcal{L}^{\mu}(\boldsymbol{\theta}) = \Phi\,|\bar{\boldsymbol{y}} - \mathbb{E}[\boldsymbol{\mu}]| \tag{6.10}$$

where $|\cdot|$ is $L1$ norm, $\bar{\boldsymbol{y}} = \frac{1}{M}\sum_{m=1}^{M}\boldsymbol{y}^{(m)}$ is the averaged label which is usually used as the ground truth in regression-based AER, and Φ is an uncertainty measure associated with the inferred posterior. The reciprocal of the total uncertainty is used as Φ in this chapter, which can be calculated as

$$\Phi = \frac{1}{\mathrm{Var}[\boldsymbol{y}]} = \frac{\boldsymbol{\upsilon}(\boldsymbol{\alpha}-1)}{\boldsymbol{\beta}(1+\boldsymbol{\upsilon})} \tag{6.11}$$

The regulariser imposes a penalty when there's an error in prediction and dynamically scales it by dividing by the total uncertainty of inferred posterior. It penalises the cases where the model produces an incorrect prediction with a small uncertainty, thus preventing the model from being over-confident. For instance, if the model produces an error with a small predicted variance, Φ is large, resulting in a large penalty. Minimising the regularisation term enforces the model to produce accurate prediction or increase uncertainty when the error is large.

In addition to imposing a penalty on the mean prediction as in Amini et al. (2020), a second term $\mathcal{L}^{\sigma}$ is proposed in order to calibrate the estimation of the aleatoric uncertainty. As discussed in the introduction, aleatoric uncertainty in AER is shown by the different emotion labels given to the same utterance by different human annotators. This chapter uses the variance of the observations to describe the aleatoric uncertainty in the emotion data. The second regularisation term is defined as:

$$\mathcal{L}^{\sigma}(\boldsymbol{\theta}) = \Phi\,|\bar{\boldsymbol{\sigma}}^2 - \mathbb{E}[\boldsymbol{\sigma}^2]| \tag{6.12}$$

where $\bar{\boldsymbol{\sigma}}^2 = \frac{1}{M}\sum_{m=1}^{M}(\boldsymbol{y}^{(m)} - \bar{\boldsymbol{y}})^2$.

6.1.3 Testing

Since NIG is the Gaussian conjugate prior, its posterior $p(\Psi|\mathcal{D})$ is in the same parametric family as the prior $p(\Psi|\Omega)$. Therefore, given a test utterance $\boldsymbol{x}_*$, the predictive posterior $p(\boldsymbol{y}_*|\mathcal{D})$ has the same form as the marginal likelihood $p(\boldsymbol{y}|\Omega)$, where $\mathcal{D}$ denotes the training set.

$$p(\boldsymbol{y}_*|\mathcal{D}) = \int p(\boldsymbol{y}_*|\Psi)p(\Psi|\mathcal{D})\,d\Psi \tag{6.13}$$

$$p(\boldsymbol{y}|\Omega) = \int p(\boldsymbol{y}|\Psi)p(\Psi|\Omega)\,d\Psi \tag{6.14}$$

In DEER, the predictive posterior and posterior are both conditioned on Ω, written as $p(\boldsymbol{y}_*|\mathcal{D},\Omega)$ and $p(\Psi|\mathcal{D},\Omega)$ to be precise. Also, the information of $\mathcal{D}$ is contained in Ω_* since $\Omega_* = f_{\hat{\theta}}(\boldsymbol{x}_*)$ and $\hat{\boldsymbol{\theta}}$ is the optimal model parameters obtained by training on $\mathcal{D}$. Then the predictive posterior can be written as $p(\boldsymbol{y}_*|\Omega_*)$. Given the conjugate prior, the predictive posterior in DEER can be computed by directly substituting the predicted Ω_* into the expression of marginal likelihood derived in Eqn. (6.6), skipping the step of calculating the posterior.

6.1.4 Summary and Comparison to Categorical Cases

For an AER task that consists of K emotion attributes, DEER trains a deep neural network model to simultaneously predict the hyperparameters $\{\Omega_1,\ldots,\Omega_K\}$ associated with the K attribute-specific NIG distributions, where $\Omega_k = \{\gamma_k, \upsilon_k, \boldsymbol{\alpha}_k, \beta_k\}(k = 1,\ldots,K)$. A DEER model thus has $4K$ output units. The system is trained by minimising the total loss *w.r.t.* $\boldsymbol{\theta}$ as:

$$\mathcal{L}_{\text{total}}(\boldsymbol{\theta}) = \sum_{k=1}^{K} \epsilon_k \mathcal{L}_k(\boldsymbol{\theta}) \tag{6.15}$$

$$\mathcal{L}_k(\boldsymbol{\theta}) = \mathcal{L}_k^{\text{NLL}}(\boldsymbol{\theta}) + \lambda_k \left[\mathcal{L}_k^{\mu}(\boldsymbol{\theta}) + \mathcal{L}_k^{\sigma}(\boldsymbol{\theta})\right] \tag{6.16}$$

where ϵ_k is the weight satisfying $\sum_{k=1}^{K} \epsilon_k = 1$, λ_k is the scale coefficient that trades off the training between data fit and uncertainty regularisation.

At test-time, the predictive posteriors are K separate Student's t-distributions $p(\boldsymbol{y}|\Omega_1)$, $p(\boldsymbol{y}|\Omega_2)$, ... , $p(\boldsymbol{y}|\Omega_K)$, each of the same form as derived in Eqn. (6.6). Apart from obtaining a distribution over the emotion attribute of the speaker, DEER also allows analytic computation of the uncertainty terms, as summarised in Table 6.1.

Table 6.1 Summary of the uncertainty terms for DEER.

Term	Expression
Predicted mean	$\mathbb{E}[\boldsymbol{y}] = \mathbb{E}[\boldsymbol{\mu}] = \boldsymbol{\gamma}$
Predicted variance (Total uncertainty)	$\text{Var}[\boldsymbol{y}] = \frac{\beta(1+\upsilon)}{\upsilon(\alpha-1)}$
Aleatoric uncertainty	$\mathbb{E}[\boldsymbol{\sigma}^2] = \frac{\beta}{\alpha-1}$
Epistemic uncertainty	$\text{Var}[\boldsymbol{\mu}] = \frac{\beta}{\upsilon(\alpha-1)}$

Table 6.2 compares the evidential deep learning approaches for discrete emotion classes (the EDL* method proposed in Chapter 5) and continuous emotion attributes (the DEER method described above). The multinomial likelihood with the Dirichlet prior is used for discrete emotion labels while the Gaussian likelihood with the NIG prior is used for continuous emotion attributes.

Table 6.2 Comparison of EDL approaches for discrete and continuous annotations.

Label y	Likelihood	Prior	Marginal Likelihood
Discrete	Multinomial $\text{P}(\boldsymbol{y}\|\boldsymbol{\pi})$	$\text{Dir}(\boldsymbol{\pi}\|\boldsymbol{\alpha})$	$\text{P}(\boldsymbol{y}\|\boldsymbol{\alpha}) = \int \text{P}(\boldsymbol{y}\|\boldsymbol{\pi}) \text{Dir}(\boldsymbol{\pi}\|\boldsymbol{\alpha})\text{d}\boldsymbol{\pi}$
Continuous	Gaussian $\text{p}(\boldsymbol{y}\|\Psi)$	$\text{NIG}(\Psi\|\Omega)$	$\text{p}(\boldsymbol{y}\|\Omega) = \int \text{p}(\boldsymbol{y}\|\Psi) \text{NIG}(\Psi\|\Omega)\text{d}\Psi$

6.2 Experimental Setup and Metrics

6.2.1 Dataset

The MSP-Podcast and IEMOCAP datasets (see Section 3.3.2) are used in this chapter. The annotations of both datasets use $K = 3$ with valence, arousal (also called activation), and dominance as the emotion attributes. MSP-Podcast uses a seven-point Likert scale for attribute annotation. Release 1.8 was used in the experiments, which contains 73,042 utterances from 1,285 speakers amounting to more than 110 hours of speech. The average variance of the labels assigned to each sentence is 0.975, 1.122, 0.889 for valence, arousal, and dominance respectively. The standard splits for training (44,879 segments), validation (7,800 segments) and testing (15,326 segments) were used in the experiments.

The IEMOCAP corpus uses a five-point Likert scale. The average variance of the labels assigned to each sentence is 0.130, 0.225, 0.300 for valence, arousal, and dominance

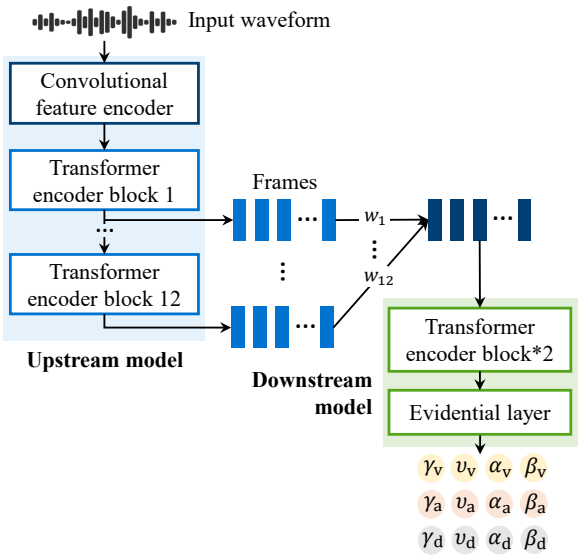


Fig. 6.1 Illustration of the model structure. Weights $w_1, \ldots, w_{12}$ for the weighted sum of the 12 Transformer encoder outputs are trainable and satisfy $\sum_{i=1}^{12} w_i = 1$.

respectively. Unless otherwise mentioned, systems on IEMOCAP were evaluated by training on Session 1-4 and testing on Session 5.

6.2.2 Model Structure

The model structure used in this chapter follows the upstream-downstream framework (Yang et al., 2021, Bommasani et al., 2021), as illustrated in Fig. 6.1. WavLM Base+ (Chen et al., 2022) was used as the upstream foundation model[2] which has 12 Transformer encoder blocks with 768-dimensional hidden states and 8 attention heads. The parameters of the pretrained model are frozen and the weighted sum of the outputs of the 12 Transformer encoder blocks are used as the speech embeddings and fed into the downstream model.

The downstream model consists of two 128-dimensional Transformer encoder blocks with 4-head self-attention, followed by an evidential layer that contains four output units for each of the three attributes, which has a total of 12 output units. The model contains 0.3M trainable parameters. A softplus activation[3] is applied to $\{\boldsymbol{\upsilon}, \boldsymbol{\alpha}, \boldsymbol{\beta}\}$ to ensure $\boldsymbol{\upsilon}, \boldsymbol{\alpha}, \boldsymbol{\beta} > 0$ with an additional +1 added to $\boldsymbol{\alpha}$ to ensure $\boldsymbol{\alpha} > 1$. A linear activation is used for $\boldsymbol{\gamma} \in \mathbb{R}$. The proposed DEER model is trained to simultaneously learn three evidential distributions for the three attributes. The weights in Eqn. (6.15) are set as $\epsilon_v = \epsilon_a = \epsilon_d = 1/3$. The scale coefficients are set to $\lambda_v = \lambda_a = \lambda_d = 0.1$ for Eqn. (6.16).

[2] Available at: https://huggingface.co/microsoft/wavlm-base-plus
[3] $\text{softplus}(x) = \log(1 + \exp(x))$

6.2.3 Baselines

The proposed method is compared to the following three baseline systems:

- GP: a Gaussian process (Williams and Rasmussen, 2006) with a radial basis function kernel, trained by maximising the per-observation-based marginal likelihood.
- MCDP: a Monte Carlo dropout (Gal and Ghahramani, 2016) system with a dropout rate of 0.4. During inference, the system was forwarded 50 times with different dropout random seeds to obtain 50 samples.
- Ensemble: an ensemble (Lakshminarayanan et al., 2017) of 10 systems initialised and trained with 10 different random seeds.

The MCDP and ensemble baselines use the same model structure as the DEER system, except that the evidential output layer was replaced by a standard FC output layer with three output units to predict the values of valence, arousal and dominance respectively. Following prior work (AlBadawy and Kim, 2018, Atmaja and Akagi, 2020b, Sridhar and Busso, 2020b), the concordance correlation coefficient (CCC) loss,

$$\mathcal{L}_{\text{ccc}} = 1 - \rho_{\text{ccc}} \tag{6.17}$$

was used for training the MCDP and ensemble baselines where ρ_{ccc} is defined in Eqn. (6.18). The CCC loss is computed based on the sequence within each mini-batch of training data. The CCC loss has been shown by previous studies to improve the continuous emotion predictions compared to the RMSE loss (Povolny et al., 2016, Trigeorgis et al., 2016, Le et al., 2017). For MCDP and ensemble, the predicted distribution of the emotion attributes are estimated based on the obtained samples by kernel density estimation.

6.2.4 Evaluation Metrics

Mean Prediction

Following prior work in continuous emotion recognition (Ringeval et al., 2015, 2017, Sridhar and Busso, 2020a, Leem et al., 2022), the concordance correlation coefficient (CCC) was used to evaluate the predicted mean. CCC combines the Pearson's correlation coefficient with the square difference between the mean of the two compared sequences:

$$\rho_{\text{ccc}} = \frac{2\rho\,\boldsymbol{\sigma}_{\text{ref}}\boldsymbol{\sigma}_{\text{hyp}}}{\boldsymbol{\sigma}_{\text{ref}}^2 + \boldsymbol{\sigma}_{\text{hyp}}^2 + \left(\boldsymbol{\mu}_{\text{ref}} - \boldsymbol{\mu}_{\text{hyp}}\right)^2}, \tag{6.18}$$

where ρ is the Pearson correlation coefficient between a hypothesis sequence (system predictions) and a reference sequence, where $\boldsymbol{\mu}_{\text{hyp}}$ and $\boldsymbol{\mu}_{\text{ref}}$ are the mean values, and $\boldsymbol{\sigma}^2_{\text{hyp}}$ and $\boldsymbol{\sigma}^2_{\text{ref}}$ are the variance values of the two sequences. Hypotheses that are well correlated with the reference but shifted in value are penalised in proportion to the deviation. The value of CCC ranges from -1 (perfect disagreement) to 1 (perfect agreement).

The root mean square error (RMSE) averaged over the test set is also reported. RMSE can be computed as follows:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\boldsymbol{y}_i^{\text{hyp}} - \boldsymbol{y}_i^{\text{ref}})^2} \tag{6.19}$$

where N is the number of samples, $\boldsymbol{y}_i^{\text{hyp}}$ is the prediction of the i^{th} sample, $\boldsymbol{y}_i^{\text{ref}}$ is the ground truth of the i^{th} sample. Since the average of the human labels, $\bar{\boldsymbol{y}}$, is defined as the ground truth in both datasets, $\bar{\boldsymbol{y}}$ were used as the reference in computing the CCC and RMSE. However, using $\bar{\boldsymbol{y}}$ also indicates that these metrics are less informative when the aleatoric uncertainty is large.

Uncertainty Estimation

Uncertainty estimation ability is measured by negative log likelihood (NLL), which is computed by fitting data to the predictive posterior $\text{p}(\boldsymbol{y})$. In this chapter, NLL(avg) defined as $-\log \text{p}(\bar{\boldsymbol{y}})$ and NLL(all) defined as $-\frac{1}{M}\sum_{m=1}^{M}\log \text{p}(\boldsymbol{y}^{(m)})$ are both used. NLL(avg) measures how much the averaged label $\bar{\boldsymbol{y}}$ fits into the predicted posterior distribution, and NLL(all) measures how much every single human label $\boldsymbol{y}^{(m)}$ fits into the predicted posterior. A lower NLL indicates better uncertainty estimation.

6.3 Experimental Results

6.3.1 Baseline Comparisons

The proposed DEER method are compared to the baselines described in Section 6.2.3. The results are shown in Table 6.3. The proposed DEER system outperforms the baselines on most of the attributes and the overall values. In particular, DEER outperforms all baselines consistently in the NLL(all) metric, indicating its superior performance in distribution estimation.

Table 6.3 Comparison with the baselines. "v", "a", "d" stands for valence, arousal, dominance. "↑" denotes the higher the better, "↓" denotes the lower the better. The best results in each column shown in bold. The second best results underlined.

	CCC ↑			RMSE ↓			NLL(avg) ↓			NLL(all) ↓		
MSP-Podcast	**v**	**a**	**d**	**v**	**a**	**d**	**v**	**a**	**d**	**v**	**a**	**d**
GP	0.342	0.595	0.486	0.811	**0.673**	**0.566**	1.447	1.408	1.297	1.727	1.808	1.592
MCDP	0.476	0.667	0.594	0.874	0.702	0.623	1.680	1.300	**1.071**	2.050	2.027	1.776
Ensemble	**0.511**	0.679	0.608	0.855	0.692	0.615	1.864	1.384	1.112	2.096	2.066	1.795
DEER	0.506	**0.698**	**0.613**	**0.772**	0.680	0.576	**1.334**	**1.285**	1.156	**1.696**	**1.692**	**1.577**
IEMOCAP	**v**	**a**	**d**	**v**	**a**	**d**	**v**	**a**	**d**	**v**	**a**	**d**
GP	0.535	0.717	0.512	0.763	0.479	0.657	1.209	**0.791**	1.047	1.295	1.205	1.380
MCDP	0.539	0.724	0.568	0.786	0.561	0.702	1.291	0.849	1.133	1.549	1.325	1.747
Ensemble	0.580	0.754	0.560	0.778	0.476	0.686	1.296	0.864	1.110	1.584	1.218	1.749
DEER	**0.596**	**0.756**	**0.569**	**0.755**	**0.457**	**0.638**	**1.070**	0.795	**1.035**	**1.275**	**1.053**	**1.283**

Table 6.4 Cross comparison of the CCC value on MSP-Podcast and IEMOCAP. "v", "a", "d" stands for valence, arousal, dominance. "Version" of MSP-Podcast denotes the release version of the dataset, and only the results from the same dateset version are comparable. "Test set" of IEMOCAP denotes the train/test split. "Ses05" denotes training on Session 1-4 and testing on Session 5. "5CV" denotes leave-one-session-out 5-fold cross validation.

	Paper	**Version**	**v**	**a**	**d**	**Average**
MSP-podcast	Ghriss et al. (2022)	1.6	0.412	0.679	0.564	0.552
	Mitra et al. (2022)	1.6	0.57	0.75	0.67	0.663
	Srinivasan et al. (2022)	1.6	0.627	0.757	0.671	0.685
	DEER	1.6	**0.629**	**0.777**	**0.684**	**0.697**
	Leem et al. (2022)	1.8	0.212	0.572	0.505	0.430
	DEER	1.8	**0.506**	**0.698**	**0.613**	**0.606**
	Paper	**Setting**	**v**	**a**	**d**	**Average**
IEMOCAP	Atmaja and Akagi (2020a)	Ses05	0.421	0.590	0.484	0.498
	Atmaja and Akagi (2021)	Ses05	0.553	0.579	0.465	0.532
	DEER	Ses05	**0.596**	**0.756**	**0.569**	**0.640**
	Srinivasan et al. (2022)	5CV	0.582	0.667	0.545	0.598
	DEER	5CV	**0.625**	**0.720**	**0.548**	**0.631**

6.3.2 Cross Comparison of Mean Prediction

Table 6.4 compares results obtained with those previously published in terms of the CCC value. Previous papers have reported results on both version 1.6 and 1.8 of the MSP-Podcast dataset. Experiments were also conducted on MSP-Podcast version 1.6 for comparison. Version 1.6 is a subset of version 1.8 and contains 34,280 segments for training, 5,958 segments for validation and 10,124 segments for testing. For IEMOCAP, apart from training

on Session 1-4 and testing on Session 5 (Ses05), the proposed system was also evaluated by a 5-fold cross-validation (5CV) based on a "leave-one-session-out" strategy. In each fold, one session was left out for testing and the others were used for training. The configuration is speaker-exclusive for both settings. As shown in Table 6.4, the proposed DEER systems achieved SOTA results on both versions of MSP-Podcast and both test settings of IEMOCAP.

6.4 Analysis

This section provides analysis of the DEER method including the effect of the aleatoric regulariser and per-observation-based likelihood loss, visualisation of predicted uncertainty, a reject option based on the predicted uncertainty, and fusion with text modality.

6.4.1 Effect of the Aleatoric Regulariser

First, by setting the aleatoric regulariser $\mathcal{L}^{\sigma}$ in Eqn. (6.16) to zero, an ablation study of the effect of the proposed extra regularisation term $\mathcal{L}^{\sigma}$ was performed. The results are given in the "$\mathcal{L}^{\sigma} = 0$" rows in Table 6.5. In this case, only $\mathcal{L}^{\mu}$ is used to regularise $\mathcal{L}^{\text{NLL}}$ and the results are compared to those trained using the complete loss defined in Eqn. (6.16), which are shown in the "$\mathcal{L}$ in Eqn. (6.16)" rows. From the results, $\mathcal{L}^{\sigma}$ improves the performance in CCC and NLL(all), but not in NLL(avg), as expected.

Table 6.5 Results using variants of the DEER loss in Eqn. (6.16). "v" , "a", "d" stands for valence, arousal, dominance. "↑" denotes the higher the better, "↓" denotes the lower the better. The "$\mathcal{L}$ in Eqn. (6.16)" row systems used the complete total loss of DEER. The "$\mathcal{L}^{\sigma} = 0$" row systems had no $\mathcal{L}^{\sigma}$ regularisation term in the total loss. The "$\mathcal{L}^{\text{NLL}} = \bar{\mathcal{L}}^{\text{NLL}}$" row systems replaced the individual human labels with $\bar{\mathcal{L}}^{\text{NLL}}$ in the total loss.

	CCC ↑			RMSE ↓			NLL(avg) ↓			NLL(all) ↓		
MSP-Podcast	**v**	**a**	**d**	**v**	**a**	**d**	**v**	**a**	**d**	**v**	**a**	**d**
$\mathcal{L}$ in Eqn. (6.16)	0.506	0.698	0.613	0.772	0.680	0.576	1.334	1.285	1.156	1.696	1.692	1.577
$\mathcal{L}^{\sigma} = 0$	0.451	0.687	0.607	0.784	0.679	0.580	1.345	1.277	1.159	1.706	1.705	1.586
$\mathcal{L}^{\text{NLL}} = \bar{\mathcal{L}}^{\text{NLL}}$	0.473	0.682	0.609	0.808	0.673	0.566	1.290	1.060	0.899	2.027	2.089	1.969
IEMOCAP	**v**	**a**	**d**	**v**	**a**	**d**	**v**	**a**	**d**	**v**	**a**	**d**
$\mathcal{L}$ in Eqn. (6.16)	0.596	0.755	0.569	0.755	0.457	0.638	1.070	0.795	1.035	1.275	1.053	1.283
$\mathcal{L}^{\sigma} = 0$	0.582	0.752	0.553	0.772	0.466	0.655	1.180	0.773	1.061	1.408	1.069	1.294
$\mathcal{L}^{\text{NLL}} = \bar{\mathcal{L}}^{\text{NLL}}$	0.585	0.759	0.555	0.786	0.444	0.633	1.001	0.727	1.036	1.627	1.329	1.441

6.4.2 Effect of the Per-Observation-Based $\mathcal{L}^{\text{NLL}}$

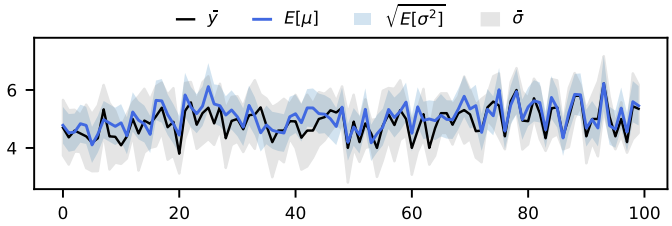
Next, the effect of the proposed per-observation-based NLL loss defined in Eqn. (6.9), $\mathcal{L}^{\text{NLL}}$, is compared to an alternative. Instead of using $\mathcal{L}^{\text{NLL}}$,

$$\bar{\mathcal{L}}^{\text{NLL}} = -\log \text{p}(\bar{\boldsymbol{y}}|\Omega) \tag{6.20}$$

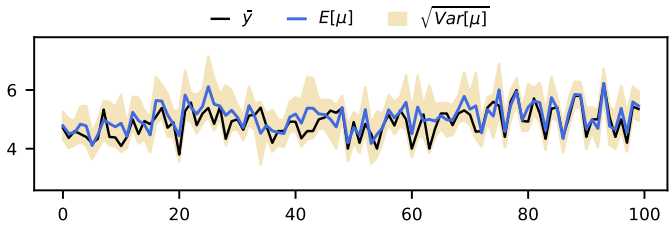
is used to compute the total loss during training, and the results are given in the "$\mathcal{L}^{\text{NLL}} = \bar{\mathcal{L}}^{\text{NLL}}$" rows in Table 6.5. While $\mathcal{L}^{\text{NLL}}$ considers the likelihood of fitting each individual observation into the predicted posterior, $\bar{\mathcal{L}}^{\text{NLL}}$ only considers the averaged observation. Therefore, it is expected that using $\bar{\mathcal{L}}^{\text{NLL}}$ instead of $\mathcal{L}^{\text{NLL}}$ yields a smaller NLL(avg) but larger NLL(all), which have been validated by the results in the table.

6.4.3 Visualisation

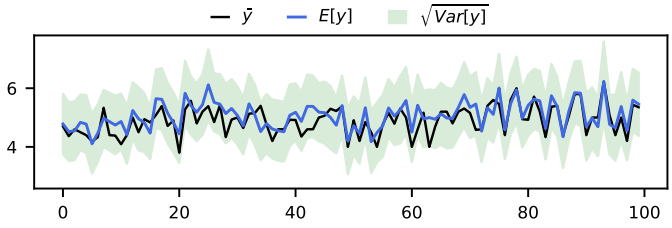
Based on a randomly selected subset test set of MSP-Podcast version 1.8, the aleatoric, epistemic and total uncertainty of the dominance attribute predicted by the proposed DEER system are shown in Fig. 6.2.



(a) Aleatoric uncertainty



(b) Epistemic uncertainty



(c) Total uncertainty

Fig. 6.2 Visualisation of (a) aleatoric (b) epistemic (c) total uncertainty of dominance for MSP-Podcast. x-axis is the test utterance index.

Fig. 6.2 (a) shows the predicted mean ± square root of the predicted aleatoric uncertainty ($\mathbb{E}[\boldsymbol{\mu}] \pm \sqrt{\mathbb{E}[\boldsymbol{\sigma}^2]}$) and the average label ± the standard deviation of the human labels ($\bar{\boldsymbol{y}} \pm \bar{\boldsymbol{\sigma}}$). It can be seen that the predicted aleatoric uncertainty (blue) overlaps with the label standard deviation (grey) and the overlapping is more evident when the mean predictions are accurate (*i.e.*, samples around index 80-100).

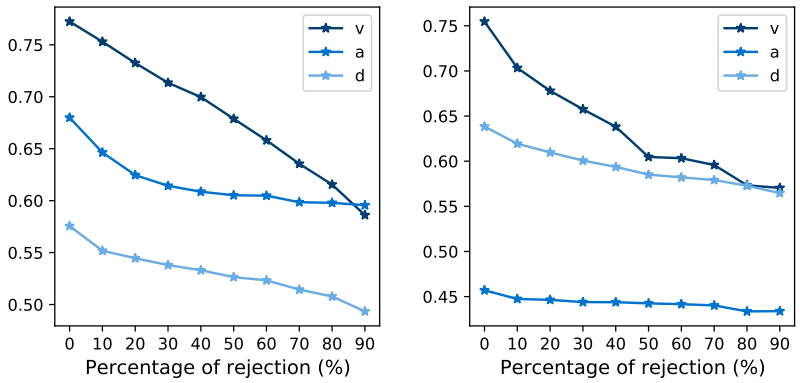
Fig. 6.2 (b) shows the predicted mean ± square root of the predicted epistemic uncertainty ($\mathbb{E}[\boldsymbol{\mu}] \pm \sqrt{\text{Var}[\boldsymbol{\mu}]}$). The epistemic uncertainty is high when the predicted mean deviates from the target (*i.e.*, samples around index 40-50) while low when the predicted mean matches the target (*i.e.*, samples around index 80-100).

Fig. 6.2 (c) shows the predicted mean ± square root of the total uncertainty ($\mathbb{E}[\boldsymbol{y}] \pm \sqrt{\text{Var}[\boldsymbol{y}]}$) which combines the aleatoric and epistemic uncertainty. The total uncertainty is high either when the input utterance is complex or the model is not confident.

6.4.4 Reject Option

A reject option was applied to analyse the uncertainty estimation performance, where the system has the option to accept or decline a test sample based on the uncertainty prediction. Since the evaluation of CCC is based on the whole sequence rather than individual samples, its computation would be affected when the sequence is modified by rejection (Wu et al., 2022a). Therefore, the reject option is performed based on RMSE.

The confidence is measured by the total uncertainty given in Eqn. (6.7). Fig. 6.3 shows the performance of the proposed DEER system with a reject option on MSP-Podcast and IEMOCAP. A percentage of utterances with the largest predicted variance were rejected. The



(a) MSP-Podcast (b) IEMOCAP

Fig. 6.3 Reject option of RMSE based on predicted variance for (a) MSP-Podcast and (b) IEMOCAP.

results at 0% rejection corresponds to the RMSE achieved on the entire test data. As the percentage of rejection increases, test coverage decreases and the average RMSE decreases showing the predicted variance succeeded in confidence estimation. The system then trades off between the test coverage and performance.

6.4.5 Fusion with Text Modality for AER

This section examines whether text information is useful for emotion attribute prediction. A bi-modal experiment is presented that explicitly[4] incorporates text information into the model. The text transcriptions of the speech data were obtained from a publicly available ASR model "wav2vec2-base-960h" [5] which finetuned the Wav2vec 2.0 (see Section 2.2.4) model on 960 hours LibriSpeech data[6] (Panayotov et al., 2015). Transcriptions were first encoded by a RoBERTa model (see Section 2.2.3) and fed into another two-layer Transformer encoder. As shown in Fig. 6.4, outputs from the text Transformer were concatenated with the outputs from the audio Transformer encoder and fed into the evidential output layer.

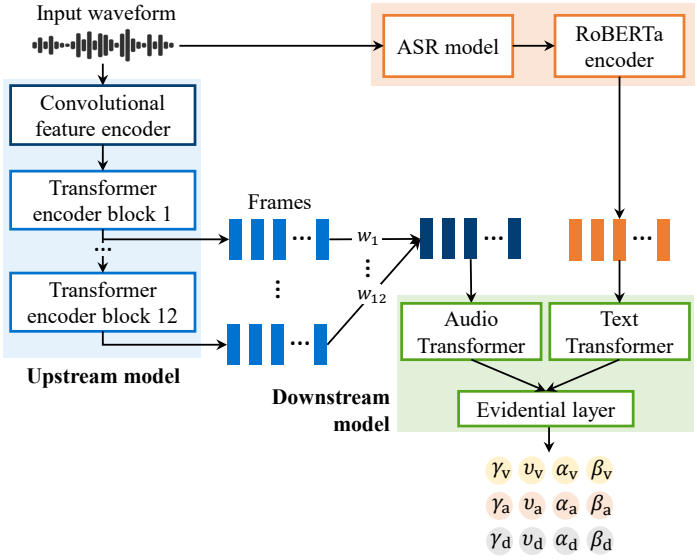


Fig. 6.4 Model structure for bi-modal experiments.

The results are shown in Table 6.6. Incorporating text information improves the estimation of valence but not necessarily for arousal and dominance. Similar phenomena were observed by Triantafyllopoulos et al. (2022). A possible explanation is that text is effective for sentiment analysis (positive or negative) but may not be as informative as audio to determine

[4]Some level of semantic information could already be implicitly encoded by the speech foundation model.
[5]Available at: https://huggingface.co/facebook/wav2vec2-base-960h
[6]Details about the LibriSpeech dataset can be found in Appendix B.1.

Table 6.6 CCC value for bi-modal experiments. “A” and “T” stands for audio and text. “v”, “a”, and “d” stand for valence, arousal, and dominance. Release 1.8 is used for MSP-Podcast. “Ses05” setup used for IEMOCAP that trains on Session 1-4 and tests on Session 5.

	MSP-podcast			IEMOCAP		
Modality	**v**	**a**	**d**	**v**	**a**	**d**
A	0.506	0.698	0.613	0.596	0.756	0.569
A+T	0.559	0.699	0.614	0.609	0.754	0.575

a speaker’s level of excitement. CCC for dominance improves more for IEMOCAP than MSP-Podcast possibly because IEMOCAP is an acted dataset and the emotion may be exaggerated compared with MSP-Podcast which contains naturalistic emotion.

6.5 Chapter Summary

Two types of uncertainty exist in AER: (i) aleatoric uncertainty arising from the inherent ambiguity of emotion and personal variations in emotion expression; (ii) epistemic uncertainty associated with the estimated network parameters given the observed data. This chapter introduces DEER for estimating those uncertainties in emotion attributes. Treating observed attribute-based annotations as samples drawn from a Gaussian distribution, DEER places a normal-inverse-gamma (NIG) prior over the Gaussian likelihood. A novel training loss is proposed which combines a per-observation-based NLL loss with a regulariser on both the mean and the variance of the Gaussian likelihood. Experiments on the MSP-Podcast and IEMOCAP datasets show that DEER can produce state-of-the-art results in estimating both the mean value and the distribution of emotion attributes. The use of NIG, the conjugate prior to the Gaussian distribution, leads to tractable analytic computation of the marginal likelihood as well as aleatoric and epistemic uncertainty associated with attribute prediction. Uncertainty estimation is analysed by visualisation and a reject option.

Beyond the scope of AER, DEER could also be applied to other tasks with subjective evaluations yielding inconsistent labels, which will be discussed in Chapter 7. It is noted that the proposed method made a Gaussian assumption on the likelihood function for the analytic computation of the uncertainties. Chapter 7 introduces an alternative method that does not restrict the distribution to be a certain type.

Chapter 7

Subjective Human Evaluations

Chapter 5 and Chapter 6 have discussed the variability in emotion annotation. Such variability is not confined to emotion perception but is also prevalent in other subjective tasks that involve human evaluation. In this chapter, we extend the scope from emotion recognition to a general framework for modelling variability in subjective human evaluations.

Human evaluation is fundamental to machine learning research, guiding processes such as data annotation and model assessment, which for instance include perceptual quality evaluation of synthesised speech, text, and images (Ma et al., 2015, Patton et al., 2016, Talebi and Milanfar, 2018, Lo et al., 2019, Borade and Netak, 2020, Ramesh and Sanampudi, 2022), annotation generation for weak supervision (Ratner et al., 2016, Wu et al., 2022b), and model optimisation based on human preferences (Schatzmann et al., 2007, Asri et al., 2016, Gür et al., 2018, Ruiz et al., 2019, Shi et al., 2019, Lin et al., 2021). Collecting human annotations or evaluations often requires substantial resources and may expose human annotators to distressing and harmful content in sensitive tasks (*e.g.*, toxic speech detection, suicidal risk prediction, and depression detection). This inspires the exploration of human annotator simulation (HAS) as a scalable and cost-effective alternative, which facilitates large-scale dataset evaluation, benchmarking, and system comparisons.

Variability is a unique aspect of real-world human evaluation, since individual variations in cognitive biases, cultural backgrounds, and personal experiences (Hirschberg et al., 2003, Wiebe et al., 2004, Haselton et al., 2015) can lead to variability in human interpretation (Lotfian and Busso, 2019, Mathew et al., 2021, Maniati et al., 2022). HAS aims to incorporate the variability present in human evaluation rather than solely relying on majority opinions, which mitigates potential biases and over-representation in scenarios where dominant opinions could potentially overshadow minority viewpoints (Dixon et al., 2018, Hutchinson et al., 2020), thus promoting fairness and inclusivity.

This chapter investigates HAS for the automatic generation of human-like annotations that takes into account the variability in human evaluation[1]. A novel meta-learning framework that treats HAS as a zero-shot density estimation problem is introduced, which allows for the efficient generation of human-like annotations for unlabelled test inputs. Under this framework, two new model classes, conditional integer flows and conditional softmax flows, are proposed to account for ordinal and categorical annotations respectively, which are common types of annotations in human evaluation tasks. The proposed methods show superior capability and efficiency to predict the aggregated behaviours of human annotators, match the distribution of human annotations, and simulate the level of inter-annotator agreement on three real-world human evaluation tasks: emotion recognition, toxic speech detection, and speech quality assessment.

The rest of the chapter is organised as follows. Section 7.1 introduces the background of HAS along with problem formulation and related work. Section 7.2 describes the proposed framework for variability-aware HAS including models for ordinal and categorical annotations in Section 7.2.2 and Section 7.2.3 respectively. Three real-word human evaluation tasks that are used to assess the performance of the proposed method are described in Section 7.3. Section 7.4 introduces the experimental setup and evaluation metrics. The results and analysis are presented in Section 7.5, followed by the chapter summary.

7.1 Human Annotator Simulation (HAS)

7.1.1 The Sources of Variability in Human Evaluation

Human perception refers to the process by which individuals interpret and make sense of the sensory information they receive from the environment. It involves the integration of sensory data, cognitive processes, emotions, and previous experiences. Subjective perception emphasises that each individual's perception of the world is unique and influenced by their internal mental states, beliefs, attitudes, and past experiences. As a result, people can interpret and react to the same stimuli differently, leading to diverse and subjective perceptions.

Each person's sensory organs, such as their eyes and ears, may have slight variations in sensitivity and acuity, leading to different perceptions of the same stimuli. Cognitive biases, the inherent mental shortcuts or tendencies that influence how humans perceive and process information, can lead to difference in judgement and decision-making. People's past experiences, cultural norms, and upbringing also shape their perceptions. Different cultural

[1] Part of this chapter has been published as a conference paper (Wu et al., 2024a). See Appendix A for more detail.

backgrounds can lead to distinct interpretations of the same event, leading to diverse reactions. The variability in humans can be manifest in various tasks such as colour perception, emotion recognition, art appreciation, and feedback preferences.

Embracing and understanding the variability of human perception is vital for various research fields such as psychology, neuroscience, human-computer interaction, *etc.*, and has practical implications in designing human-centred systems and promoting empathy and diversity. It helps create products and interfaces that cater to diverse user needs and preferences in fields like human-computer interaction and user experience design. Being aware of the variability of perception is crucial in ethical decision-making. It helps ensure that different perspectives and cultural sensitivities are considered, which in turn helps identify and address potential biases that might disproportionately affect certain groups or lead to unfair outcomes.

7.1.2 The Variability in Human Evaluation is Valuable

As discussed in Section 7.1.1, each individual's perception of the world is unique and influenced by their physical state and cognitive biases, which leads to diverse and subjective interpretations. Such subjectivity can be manifest in various tasks such as emotion recognition (Hirschberg et al., 2003, Mihalcea and Liu, 2006), perceptual quality assessment (Wiebe et al., 2004, Seshadrinathan et al., 2010, Zen and Vanderdonckt, 2016), and user experience evaluation (Zen and Vanderdonckt, 2016). It has been argued that achieving a deterministic "ground truth" in subjective tasks like human evaluation is not feasible, nor essential (Alm, 2011, Wu et al., 2022e). Therefore, we advocate for methodologies that focus on modelling annotators' subjective interpretations, rather than seek to reduce the variability in annotations: instead of only predicting the majority opinion, it is important to account for human perception variability when designing a human annotator simulator. Below are three examples that demonstrate the importance of modelling variability in HAS:

Revealing data ambiguity. Incorporating the variability in human perception allows HAS systems to reveal potential ambiguity or complexity in data, providing valuable insight for further analysis[2].

Mitigating bias and over-representation. Incorporating the variability in human judgements prevents HAS from being biased towards a certain perspective and ignoring minority

[2]Taking emotion perception as an example, there are certain cases that convey fairly clear emotional expressions (*e.g.*, laughing) and most annotators agree that the speaker is happy. However, there also exist cases where the emotion is more subtle and human opinions can easily diverge. For instance, in a dyadic situation where two people disagree, with the speaker being the one who compromises (*i.e.*, the case illustrated in Fig. 5.16), some people would perceive the emotion as frustrated while others may interpret it as angry. These types of data contain ambiguous emotion that is inherently more complex to deal with.

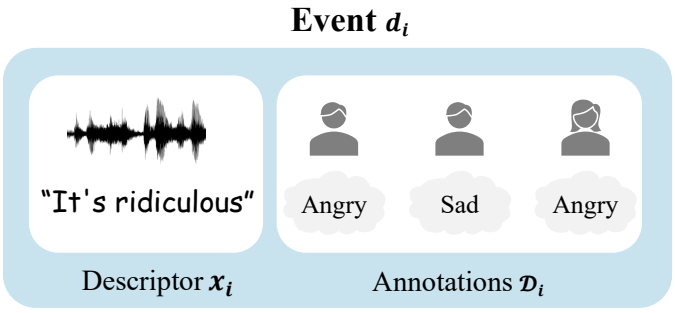


Fig. 7.1 Definition of an event for HAS.

viewpoints, leading to a more inclusive representation of opinions where all viewpoints are given due consideration.

Improving model alignment. Optimisation based on human feedback has led to superior performance on tasks such as text generation (Christiano et al., 2017, Ouyang et al., 2022, Rafailov et al., 2023), which aligns the behaviour of language models with human preferences. HAS could be helpful in this task, as it is an efficient and cost-effective alternative to generating human feedback.

7.1.3 Problem Formulation and Related Work

Problem Formulation

Denote an event as $\boldsymbol{d}_i$, which consists of a descriptor (*e.g.,* an utterance or text) $\boldsymbol{x}_i$ and a set of M_i human annotations $\mathcal{D}_i = \{\boldsymbol{\eta}_i^{(m)}\}_{m=1}^{M_i}$ for $\boldsymbol{x}_i$. The definition of an event is illustrated in Fig. 7.1. Note that different events may be labelled by different sets of annotators. Given a dataset of training events $\mathcal{D} = \{(\boldsymbol{x}_i, \mathcal{D}_i)\}_{i=1}^{N}$, HAS aims to model the conditional annotation distribution $\mathrm{p}(\boldsymbol{\eta}_i|\boldsymbol{x}_i)$ given the observations $\mathcal{D}_i$ of $\boldsymbol{\eta}_i$ provided by different annotators. For a unseen test descriptor $\boldsymbol{x}_*$, a HAS system can predict $\mathrm{p}(\boldsymbol{\eta}_*|\boldsymbol{x}_*)$ to simulate human-like annotations $\mathcal{D}_* = \{\boldsymbol{\eta}_*^{(m)}\}_{m=1}^{M_*}$ in a way that reflects how it would be labelled by human annotators.

Related Work

Prior work mainly investigated three approaches to simulating human annotations. Although some of these techniques have already been discussed in previous chapters, they are re-introduced in this section for a better understanding and comparison.

The first approach uses a single proxy variable $\boldsymbol{\eta}_i'$ (*e.g.*, majority vote) to summarise all annotations for each descriptor $\boldsymbol{x}_i$ (Kim et al., 2013, Djuric et al., 2015, Patton et al., 2016,

Poria et al., 2017). This creates a proxy dataset $\mathcal{D}' = \{(\boldsymbol{x}_i, \boldsymbol{\eta}'_i)\}_{i=1}^{N}$ and converts HAS into a supervised learning problem, which is usually solved by fitting a discriminative model to estimate the conditional distribution for the proxy variable. During testing, given an unseen descriptor $\boldsymbol{x}_*$, the model predicts the proxy variable $\boldsymbol{\eta}'_*$ for $\boldsymbol{x}_*$. Clearly, modelling a single proxy variable as in this approach fails to take into account the subjectivity and diversity in human behaviour and perception. Other work incorporated the variance of human annotations into the proxy variable (Deng et al., 2012, Prabhakaran et al., 2012, Plank et al., 2014, Dang et al., 2017, Han et al., 2017, Leng et al., 2021). However, all these approaches still focus on obtaining the "correct" label (*e.g.*, aiming for improved prediction accuracy) and minimising the discrepancy among annotators (*e.g.*, reducing "noise" in annotations) rather than embracing inter-annotator disagreements.

The second approach explicitly models the behaviour of different annotators using different individual models in an ensemble or different heads in a single model (Fayek et al., 2016, Chou and Lee, 2019, Davani et al., 2022). This approach is computationally feasible only when the number of annotators is relatively small and when a sufficient quantity of annotation is available for each annotator, which is not applicable to large crowd-sourced datasets (Lotfian and Busso, 2019, Mathew et al., 2021) that are common in real-world applications.

The third approach approximates subjective probability distributions using Markov chain Monte Carlo with people (Sanborn and Griffiths, 2007, Harrison et al., 2020), which requires human annotators to be involved in the process in a dynamic setting. These methods present the descriptor $\boldsymbol{x}_*$ to human participants and asks them to provide a sequence of decisions $\mathcal{D}_*$ following the Metropolis-Hasting acceptance rule (Metropolis et al., 1953, Hastings, 1970). The annotation distribution $\mathrm{p}(\boldsymbol{\eta}_*|\boldsymbol{x}_*)$ is then estimated based on $\mathcal{D}_*$. In other words, this requires access to human annotations $\mathcal{D}_*$ for estimating the annotation distribution for each $\boldsymbol{x}_*$, and there is no obvious way to transfer information between different events. Therefore, these methods cannot be applied to simulate annotation distributions for unlabelled test descriptors.

7.2 A Meta-Learning Framework for Zero-Shot HAS

This chapter proposes a novel framework for HAS that meta-learns a flow model to estimate the human annotation distribution $\mathrm{p}(\boldsymbol{\eta}|\boldsymbol{x})$ across all training events $\mathcal{D}$. The proposed model learns to learn (*i.e.*, meta-learns) how to estimate the underlying distribution of human annotations $\mathcal{D}_i$ for any given descriptor $\boldsymbol{x}_i$ by leveraging the diverse human annotations, rather than designing a proxy variable to summarise $\mathcal{D}_i$ as in the first approach described in

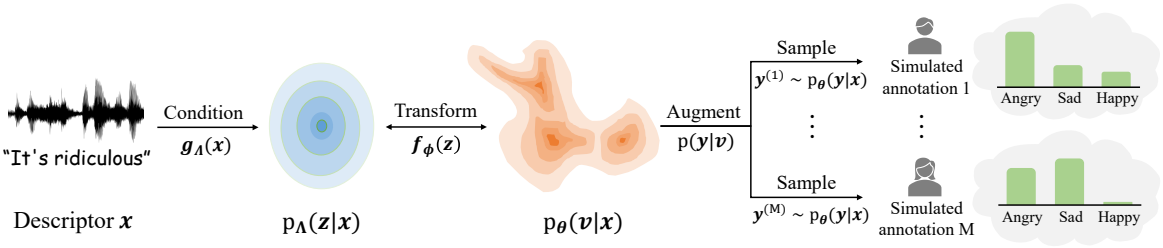


Fig. 7.2 Diagram for the proposed zero-shot human annotator simulation framework.

Section 7.1.3. Unlike the second approach in Section 7.1.3 which separately models each individual human annotator with a different model, the proposed method is compatible with large crowd-sourced datasets since it amortises across annotators with a single flow model. Moreover, the proposed model is a zero-shot human annotation simulator which can estimate the human annotation distribution $p(\boldsymbol{\eta}_*|\boldsymbol{x}_*)$ for any unseen test descriptor $\boldsymbol{x}_*$ without access to any human annotations $\mathcal{D}_*$ for $\boldsymbol{x}_*$, in contrast to the third method in Section 7.1.3 which requires human annotators to be dynamically involved in the process of labelling $\boldsymbol{x}_*$.

7.2.1 A Latent Variable Model for HAS

The proposed meta-learning framework for HAS is realised using a latent variable model[3]:

$$p_{\boldsymbol{\theta}}(\boldsymbol{y}|\boldsymbol{x}) = \int p(\boldsymbol{y}|\boldsymbol{v})p_{\boldsymbol{\phi}}(\boldsymbol{v}|\boldsymbol{z})p_{\boldsymbol{\Lambda}}(\boldsymbol{z}|\boldsymbol{x})d\boldsymbol{v}d\boldsymbol{z}, \tag{7.1}$$

where the conditional prior $p_{\boldsymbol{\Lambda}}(\boldsymbol{z}|\boldsymbol{x})$ learns to summarise useful information about $\boldsymbol{x}$ and encode the possible disagreements over $\boldsymbol{x}$ among different human annotators, which is helpful for the likelihood $p_{\boldsymbol{\phi}}(\boldsymbol{y}|\boldsymbol{z}) = \int p(\boldsymbol{y}|\boldsymbol{v})p_{\boldsymbol{\phi}}(\boldsymbol{v}|\boldsymbol{z})d\boldsymbol{v}$ to simulate human-like annotations.

Fig. 7.2 illustrates the proposed framework. Specifically, the conditional prior is modelled by a conditional factorised Gaussian distribution $p_{\boldsymbol{\Lambda}}(\boldsymbol{z}|\boldsymbol{x}) = \mathcal{N}(\boldsymbol{z}|\boldsymbol{\mu}_{\boldsymbol{\Lambda}}(\boldsymbol{x}), \text{diag}(\boldsymbol{\sigma}^2_{\boldsymbol{\Lambda}}(\boldsymbol{x})))$ whose mean $\boldsymbol{\mu}_{\boldsymbol{\Lambda}}(\boldsymbol{x})$ and variance $\boldsymbol{\sigma}^2_{\boldsymbol{\Lambda}}(\boldsymbol{x})$ are parameterised by a neural network with parameters $\boldsymbol{\Lambda}$. The intermediate variable $\boldsymbol{v}$ is obtained by a deterministic invertible transformation $p_{\boldsymbol{\phi}}(\boldsymbol{v}|\boldsymbol{z}) = \delta(\boldsymbol{v} - \boldsymbol{f}_{\boldsymbol{\phi}}(\boldsymbol{z}))$, where $\boldsymbol{f}_{\boldsymbol{\phi}}(\boldsymbol{z})$ is parameterised by an invertible neural network with parameters $\boldsymbol{\phi}$, and $\delta(\cdot)$ is the multivariate Dirac delta function. This results in a conditional normalising flow (CNF):

$$p_{\boldsymbol{\theta}}(\boldsymbol{v}|\boldsymbol{x}) = \int \delta(\boldsymbol{v} - \boldsymbol{f}_{\boldsymbol{\phi}}(\boldsymbol{z}))p_{\boldsymbol{\Lambda}}(\boldsymbol{z}|\boldsymbol{x})d\boldsymbol{z} = p_{\boldsymbol{\Lambda}}\left(\boldsymbol{f}^{-1}_{\boldsymbol{\phi}}(\boldsymbol{v})\middle|\boldsymbol{x}\right)\left|\det\left(\frac{\partial \boldsymbol{f}^{-1}_{\boldsymbol{\phi}}(\boldsymbol{v})}{\partial \boldsymbol{v}}\right)\right|, \tag{7.2}$$

[3]For clarity, different notations are used for human annotations $\boldsymbol{\eta}$ and model outputs $\boldsymbol{y}$.

where $\det(\cdot)$ denotes the determinant operator, $\partial \boldsymbol{f}_{\phi}^{-1}(\boldsymbol{v})/\partial \boldsymbol{v}$ denotes the Jacobian matrix of $\boldsymbol{f}_{\phi}^{-1}(\boldsymbol{v})$, and $\boldsymbol{\theta} \coloneqq \{\phi, \boldsymbol{\Lambda}\}$ denotes all parameters in this base CNF. This modelling choice has the advantage of having a tractable marginal likelihood as in Eqn. (7.2) while not restricting the intermediate variable $\boldsymbol{v}$ to a specific type of distribution as in previous methods, *e.g.*, Gaussian (Han et al., 2017) and Student's-t (Chapter 6) distributions, thus offering enhanced tractability, flexibility and generality. In addition, samples can be efficiently drawn from this model by first drawing $\boldsymbol{z} \sim \mathrm{p}_{\boldsymbol{\Lambda}}(\boldsymbol{z}|\boldsymbol{x})$ from the conditional prior and then computing the deterministic flow transformation $\boldsymbol{v} = \boldsymbol{f}_{\phi}(\boldsymbol{z})$.

Finally, the output variable $\boldsymbol{y}$ is obtained by augmenting the intermediate variable $\boldsymbol{v}$ using the transformation $\mathrm{p}(\boldsymbol{y}|\boldsymbol{v})$, in order to accommodate different types of annotation. For continuous annotations, the identity transformation $\mathrm{p}(\boldsymbol{y}|\boldsymbol{v}) = \delta(\boldsymbol{y} - \boldsymbol{v})$ is used, which exactly recovers the base CNF model. However, real-world human evaluation tasks often involve discrete annotations that are either ordinal or categorical. In the following sections, two new model classes with meta-learning objectives are introduced to accommodate these annotation types.

7.2.2 Conditional Integer Flows for Ordinal Annotations

I-CNF Modelling

Discrete ordinal annotations are often used in K-point rating systems, where the ratings are integer-valued with a clear ordering. A new class of models is proposed, named conditional integer flows (I-CNFs), which augment the base CNFs by quantising the continuous intermediate variable $\boldsymbol{v}$ to its nearest integer by using a rounding transformation $\mathrm{p}(\boldsymbol{y}|\boldsymbol{v}) = \mathbb{I}(\boldsymbol{y} - 1/2 < v \leq \boldsymbol{y} + 1/2)$, where $\mathbb{I}(\cdot)$ is the indicator function. Let o be an ordinal variable that represents the ordinal human rating for an input $\boldsymbol{x}$. The marginal likelihood of I-CNF is given by

$$\mathrm{p}_{\boldsymbol{\theta}}(o = \boldsymbol{y}|\boldsymbol{x}) = \int_{-\infty}^{\infty} \mathbb{I}(\boldsymbol{y} - 1/2 < \boldsymbol{v} \leq \boldsymbol{y} + 1/2)\mathrm{p}_{\boldsymbol{\theta}}(\boldsymbol{v}|\boldsymbol{x})\mathrm{d}\boldsymbol{v} = \int_{\boldsymbol{y}-1/2}^{\boldsymbol{y}+1/2} \mathrm{p}_{\boldsymbol{\theta}}(\boldsymbol{v}|\boldsymbol{x})\mathrm{d}\boldsymbol{v}, \tag{7.3}$$

where $\mathrm{p}_{\boldsymbol{\theta}}(\boldsymbol{v}|\boldsymbol{x})$ is the marginal likelihood of the base CNF defined in Eqn. (7.2). Since the marginal likelihood of I-CNF given in Eqn. (7.3) is analytically intractable due to the rounding transformation, it is approximated using numerical integration. In practice, the rectangular rule is found to work well in terms of both performance and efficiency in this setting, where the density of $\mathrm{p}_{\boldsymbol{\theta}}(\boldsymbol{v}|\boldsymbol{x})$ within the interval $\boldsymbol{v} \in (\boldsymbol{y} - 1/2, \boldsymbol{y} + 1/2]$ is approximated

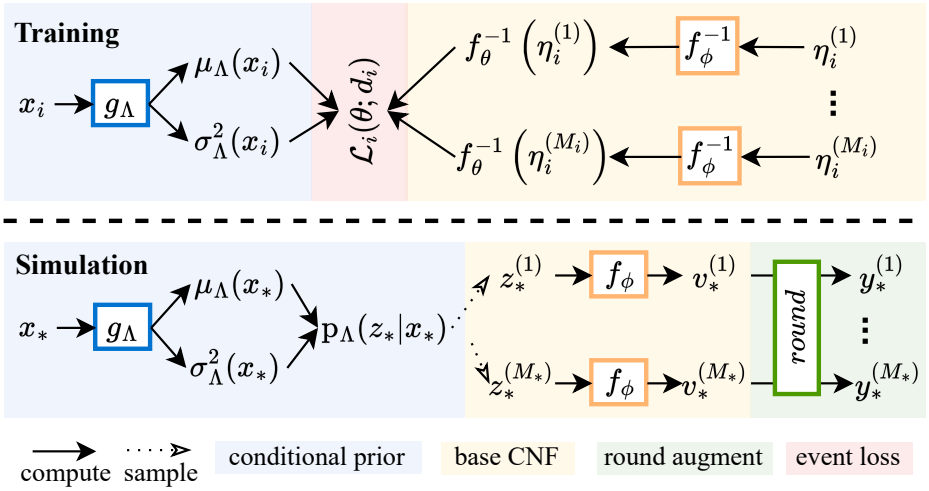


Fig. 7.3 Illustration for I-CNF training and simulation workflow.

by the midpoint density value:

$$\int_{y-1/2}^{y+1/2} \mathrm{p}_{\boldsymbol{\theta}}(\boldsymbol{v}|\boldsymbol{x})\mathrm{d}\boldsymbol{v} \approx \left(\left(\boldsymbol{y}+\frac{1}{2}\right)-\left(\boldsymbol{y}-\frac{1}{2}\right)\right)\cdot \mathrm{p}_{\boldsymbol{\theta}}\left(\frac{(\boldsymbol{y}-1/2)+(\boldsymbol{y}+1/2)}{2}\middle|\boldsymbol{x}\right) = \mathrm{p}_{\boldsymbol{\theta}}(\boldsymbol{y}|\boldsymbol{x}). \tag{7.4}$$

This means that Eqn. (7.2) can be used as a proxy to evaluate the likelihood of I-CNF. The structure of proposed I-CNF is illustrated in Fig. 7.3.

Meta-Learning I-CNF

Using the numerical approximation given in Eqn. (7.4), the loss $\mathcal{L}(\boldsymbol{\theta};\boldsymbol{d}_i)$ for I-CNF on a single event $\boldsymbol{d}_i$ can be defined as the average negative log marginal likelihood of Eqn. (7.2) evaluated on the human annotations $\mathcal{D}_i = \{\boldsymbol{\eta}_i^{(m)}\}_{m=1}^{M_i}$ given the corresponding input $\boldsymbol{x}_i$:

$$\mathcal{L}(\boldsymbol{\theta};\boldsymbol{d}_i) = -\frac{1}{M_i}\sum_{m=1}^{M_i}\left(\log \mathrm{p}_{\boldsymbol{\Lambda}}\left(f_{\boldsymbol{\phi}}^{-1}(\boldsymbol{\eta}_i^{(m)})\middle|\boldsymbol{x}_i\right) + \log\left|\det\left(\frac{\partial f_{\boldsymbol{\phi}}^{-1}(\boldsymbol{\eta}_i^{(m)})}{\partial \boldsymbol{\eta}_i^{(m)}}\right)\right|\right). \tag{7.5}$$

Following the episodic training scheme (Vinyals et al., 2016, Snell et al., 2017, Chen et al., 2023b), density estimation on each dataset is treated as a learning problem and randomly sample a subset of such learning problems to train on at each step during meta-training. This results in a meta-learning objective across the training events in $\mathcal{D}$:

$$\mathcal{L}_{\text{meta}}(\boldsymbol{\theta};\mathcal{D}) = \mathbb{E}_{\boldsymbol{d}_i\sim \mathrm{p}(\mathcal{D})}[\mathcal{L}(\boldsymbol{\theta};\boldsymbol{d}_i)], \tag{7.6}$$

where $\mathrm{p}(\mathcal{D})$ denotes the uniform distribution over $\mathcal{D}$. Intuitively, this objective maps each human annotation to the latent space of the corresponding descriptor by the I-CNF during meta-training, which helps the model to build a diverse latent representation that captures the variability in human annotations across different descriptors.

At test time, the I-CNF can simulate human-like annotations for an unseen, unlabelled input $\boldsymbol{x}_*$ by first drawing $\boldsymbol{v}_*^{(m)} \sim \mathrm{p}_{\boldsymbol{\theta}}(\boldsymbol{v}|\boldsymbol{x}_*)$ from the base CNF then applying the rounding function $\boldsymbol{y}_*^{(m)} = \lfloor \boldsymbol{v}_*^{(m)} \rfloor$, for $m = 1, \cdots, M_*$, where M_* denotes the number of annotations to be simulated.

7.2.3 Conditional Softmax Flows for Categorical Annotations

S-CNF Modelling

To account for non-ordinal categorical annotations (*e.g.*, emotion categories), a new class of models called conditional softmax flows (S-CNFs) is proposed, which augments the base CNFs by applying the softmax function $\mathrm{p}(\boldsymbol{y}|\boldsymbol{v}) = \delta\left(\boldsymbol{y} - \mathrm{softmax}(\boldsymbol{v})\right)$ to transform the continuous intermediate variable $\boldsymbol{v}$ into categorical probabilities $\boldsymbol{y}$. Let c be a categorical variable with probability $\mathrm{P}(c = k|\boldsymbol{y}) = \boldsymbol{y}_k$ $(k = 1, \cdots, K)$ that represents the categorical human annotation for an input $\boldsymbol{x}$, with $\mathrm{P}(c = k|\boldsymbol{v}) = \int \boldsymbol{y}_k \delta\left(\boldsymbol{y} - \mathrm{softmax}(\boldsymbol{v})\right) \mathrm{d}\boldsymbol{y} = \mathrm{softmax}(\boldsymbol{v})_k$. The marginal likelihood of S-CNF is given by

$$\mathrm{P}_{\boldsymbol{\theta}}(c = k|\boldsymbol{x}) = \int \mathrm{P}(c = k|\boldsymbol{v})\mathrm{p}_{\boldsymbol{\theta}}(\boldsymbol{v}|\boldsymbol{x})\mathrm{d}\boldsymbol{v} = \int \mathrm{softmax}(\boldsymbol{v})_k \mathrm{p}_{\boldsymbol{\theta}}(\boldsymbol{v}|\boldsymbol{x})\mathrm{d}\boldsymbol{v}, \tag{7.7}$$

where $\mathrm{p}_{\boldsymbol{\theta}}(\boldsymbol{v}|\boldsymbol{x})$ is the marginal likelihood of the base CNF defined in Eqn. (7.2). Since the marginal likelihood of the S-CNF given in Eqn. (7.7) is analytically intractable due to the softmax transformation, it is approximated using variational inference (Wainwright et al., 2008) with a learnable mean-field Gaussian variational posterior $\mathrm{q}_{\boldsymbol{\Omega}}(\boldsymbol{v}|\boldsymbol{y}) = \mathcal{N}(\boldsymbol{v}|\boldsymbol{\mu}_{\boldsymbol{\Omega}}(\boldsymbol{y}), \mathrm{diag}(\boldsymbol{\sigma}^2_{\boldsymbol{\Omega}}(\boldsymbol{y})))$, which can be seen as a probabilistic inverse of the softmax transformation $\mathrm{p}(\boldsymbol{y}|\boldsymbol{v})$. Applying Jensen's inequality to the log marginal likelihood of the S-CNF in Eqn. (7.7), a tractable evidence lower bound (ELBO) is obtained:

$$\log \mathrm{P}_{\boldsymbol{\theta}}(c = k|\boldsymbol{x}) \geq \mathbb{E}_{\mathrm{q}_{\boldsymbol{\Omega}}(\boldsymbol{v}|\boldsymbol{y})}\left[\log \mathrm{P}(c = k|\boldsymbol{v}) + \log \mathrm{p}_{\boldsymbol{\theta}}(\boldsymbol{v}|\boldsymbol{x}) - \log \mathrm{q}_{\boldsymbol{\Omega}}(\boldsymbol{v}|\boldsymbol{y})\right]. \tag{7.8}$$

It is worth noting that the softmax flow likelihood $\mathrm{P}(c = k|\boldsymbol{v}) = \mathrm{softmax}(\boldsymbol{v})_k$ places non-zero probability mass for every category $k = 1, \cdots, K$, which is different from argmax flow (Hoogeboom et al., 2021) whose likelihood only places probability mass for a single category. From a modelling perspective, softmax flow has a greater capacity to represent the variability and uncertainty in human annotations. From an optimisation perspective, the

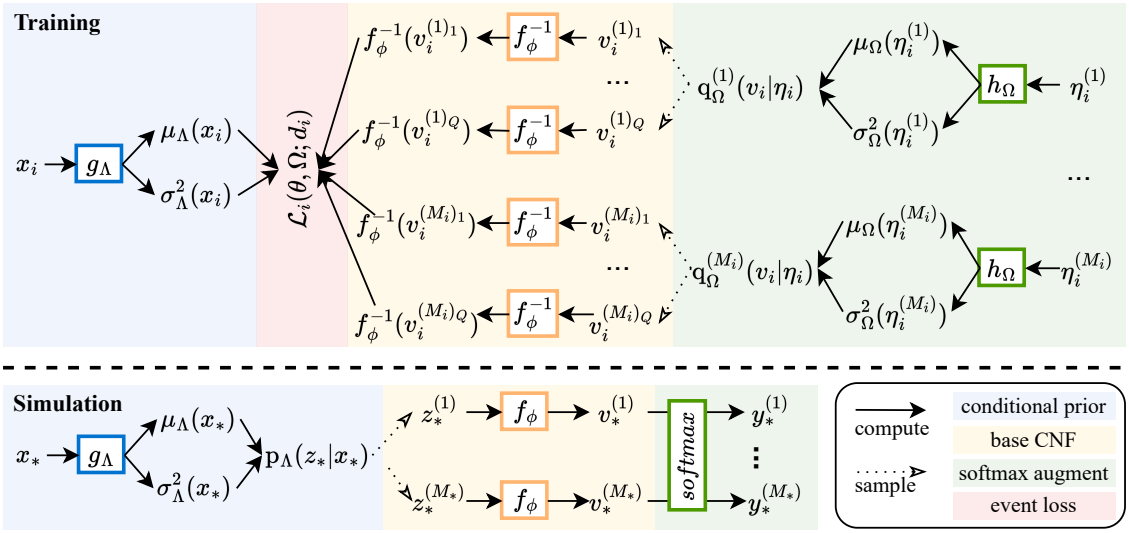


Fig. 7.4 Illustration for S-CNF training and simulation workflow.

ELBO for softmax flow is always well-defined, whereas the ELBO for argmax flow is not defined when the model output does not match the human annotation, since the log-likelihood would be $\log(0)$ in this case, which requires additional thresholding tricks to fix (Hoogeboom et al., 2021).

Meta-Learning S-CNF

Using the variational approximation defined in Eqn. (7.8), the loss $\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\Omega}; \boldsymbol{d}_i)$ for S-CNF on a single event $\boldsymbol{d}_i$ can be defined as the average negative ELBO evaluated on the set of the human annotations $\mathcal{D}_i = \{\boldsymbol{\eta}_i^{(m)}\}_{m=1}^{M_i}$ for the corresponding descriptor $\boldsymbol{x}_i$: $\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\Omega}; \boldsymbol{d}_i) =$

$$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\Omega}; \boldsymbol{d}_i) = -\frac{1}{M_i} \sum_{m=1}^{M_i} \mathbb{E}_{\mathrm{q}_{\boldsymbol{\Omega}}(\boldsymbol{v}|\boldsymbol{\eta}_i^{(m)})} \left[\sum_{k=1}^{K} \boldsymbol{\eta}_{i,k}^{(m)} \log \mathrm{P}(c_i = k|\boldsymbol{v}) + \log \mathrm{p}_{\boldsymbol{\theta}}(\boldsymbol{v}|\boldsymbol{x}_i) - \log \mathrm{q}_{\boldsymbol{\Omega}}(\boldsymbol{v}|\boldsymbol{\eta}_i^{(m)}) \right], \tag{7.9}$$

where the expectation over the variational posterior is approximated by Monte Carlo simulation with the reparameterisation trick (Kingma and Welling, 2014). As in Section 7.2.2, we follow the episodic training scheme with a meta-learning objective $\mathcal{L}_{\text{meta}}(\boldsymbol{\theta}, \boldsymbol{\Omega}; \mathcal{D}) = \mathbb{E}_{\boldsymbol{d}_i \sim \mathrm{p}(\mathcal{D})}[\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\Omega}; \boldsymbol{d}_i)]$ for meta-training and use a similar flow sampling scheme but apply the softmax function $\boldsymbol{y}_*^{(m)} = \mathrm{softmax}(\boldsymbol{v}_*^{(m)})$ to the samples $\boldsymbol{v}_*^{(m)}$ from the base CNF at test time. Note that each sample of S-CNF is a categorical distribution with probabilities $\boldsymbol{y}_*^{(m)}$.

The structure of proposed S-CNF is illustrated in Fig. 7.4. The procedure of sampling from and optimising S-CNF are summarised in Algorithm 1 and 2.

Algorithm 1 Sampling from S-CNF

Input: $\boldsymbol{x}$
Output: Categorical probability $\boldsymbol{y}$
Compute $\boldsymbol{\mu}_{\boldsymbol{\Lambda}}(\boldsymbol{x}), \boldsymbol{\sigma}^2_{\boldsymbol{\Lambda}}(\boldsymbol{x}) = \boldsymbol{g}_{\boldsymbol{\Lambda}}(\boldsymbol{x})$
Sample $\boldsymbol{z} \sim \mathcal{N}(\boldsymbol{\mu}_{\boldsymbol{\Lambda}}(\boldsymbol{x}), \text{diag}(\boldsymbol{\sigma}^2_{\boldsymbol{\Lambda}}(\boldsymbol{x}))$
Compute $\boldsymbol{v} = \boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{z})$
Compute $\boldsymbol{y} = \text{softmax}(\boldsymbol{v})$

Algorithm 2 Optimising S-CNF

Input: $\boldsymbol{x}, \mathcal{D} = \{\boldsymbol{\eta}^{(1)}, \cdots, \boldsymbol{\eta}^{(M)}\}$
Output: ELBO $\mathcal{L}^{\text{ELBO}}$ on dataset $\mathcal{D}$
for $m = 1, \cdots, M$ **do**
  Compute $\boldsymbol{\mu}_{\boldsymbol{\Omega}}(\boldsymbol{\eta}^{(m)}), \boldsymbol{\sigma}^2_{\boldsymbol{\Omega}}(\boldsymbol{\eta}^{(m)}) = \boldsymbol{h}_{\boldsymbol{\Omega}}(\boldsymbol{\eta}^{(m)})$
  for $j = 1, \cdots, Q$ **do**
    Sample $\boldsymbol{v}_j \sim \boldsymbol{q}_{\boldsymbol{\Omega}}(\boldsymbol{v}|\boldsymbol{\eta}^{(m)})$
    Compute $\mathcal{L}_j^{(m)} = -\sum_{k=1}^{K} \boldsymbol{\eta}_k^{(m)} \log \text{P}(c = k|\boldsymbol{v}_j) + \log \text{p}_{\boldsymbol{\theta}}(\boldsymbol{v}_j|\boldsymbol{x}) - \log \boldsymbol{q}_{\boldsymbol{\Omega}}(\boldsymbol{v}_j|\boldsymbol{\eta}^{(m)})$
  end for
  Compute $\mathcal{L}_m^{\text{ELBO}} = \frac{1}{Q} \sum_{j=1}^{Q} \mathcal{L}_j^{(m)}$
end for
Compute $\mathcal{L}^{\text{ELBO}} = \frac{1}{M} \sum_{m=1}^{M} \mathcal{L}_m^{\text{ELBO}}$

7.3 Evaluation Tasks

The proposed meta-learned zero-shot density estimation method for HAS from Section 7.2 was evaluated by three representative real-world human evaluation tasks for speech and natural language processing. I-CNF was evaluated on emotion attribute prediction and speech quality assessment where ordinal annotations are normally used. S-CNF was evaluated on emotion category annotation and toxic speech detection where categorical annotations are used.

7.3.1 Emotion Annotation

The proposed method is first evaluated on AER. Both discrete emotion class labelling and continuous emotion attribute prediction are studied in this chapter. The proposed method can enhance the fairness of emotion annotation as it better handles different opinions among human annotators.

Emotion Dataset

The MSP-Podcast dataset (see Section 3.3.2) was used for the emotion annotation tasks. Release 1.6 was used, which contains 50k+ utterances from 1k+ speakers consisting of 80+ hours of speech. The standard splits of training (34,280 segments), validation (5,958

segments) and test (10,124 segments) are used. Each utterance was labelled by at least 5 human annotators, and there are 6.7 annotations per utterance on average.

The emotion class labels were grouped into five categories: “angry”, “sad”, “happy”, “neutral”, and “other”. In MSP-Podcast, each annotator can choose from ten emotion classes to label the primary emotion of an utterance: “angry”, “sad”, “happy”, “surprise”, “fear”, “disgust”, “contempt”, “neutral”, “other”. Although only one option is allowed, they can say “other” and define their own emotion class which can be more than one. During label processing, the original “other” class is split into sub-classes depending on the manual defined label and merged with the predefined labels. The grouping used here was as follows: (i) “Angry” includes “angry”, “disgust”, “contempt”, “annoyed”; (ii) “Sad” includes “sad”, “frustrated”, “disappointed”, “depressed”, “concerned”; (iii) “Happy” includes “happy”, “excited”, “amused”; (iv) “Neutral” includes “neutral”; (v) “Other” includes all other emotion sub-classes not listed above. It is worth noting that 16.5% of the utterances in this dataset do not have a majority emotion class, showing strong disagreement among the human annotators.

For emotion attribute annotation, annotators label the attributes in terms of valence, arousal, and dominance on a 7-point Likert scale.

7.3.2 Toxic Speech Detection

Toxic speech detection aims to filter out harmful and offensive language in written or spoken communications, such as insults, threats and harassment, which can lead to emotional distress, cyberbullying, and hostile online environments. Developing effective toxic detection methods is crucial for creating safer and more respectful online environments and promoting positive interactions and healthy communications among users. The proposed method incorporates interpretations from different human annotators, leading to a comprehensive understanding of hate speech, which is a good substitute for human annotators to reduce their exposure to distressing and harmful content.

Toxic Speech Dataset

The HateXplain dataset (Mathew et al., 2021) is used in this experiment, which contains over 20k text posts from Twitter and Gab. These posts are labelled using crowd-sourcing with the commonly used 3-category annotation: hate, offensive, normal. Each post is annotated by three annotators. Cases where all the three annotators choose a different class (919 out of 20,148 posts) were originally excluded from the standard split of the dataset. These cases are incorporated into training, validation, and test sets in an 8:1:1 ratio to better reflect the

inter-annotator disagreements, resulting in 16,118 posts for training, 2,014 for validation, and 2,016 for testing in total.

7.3.3 Speech Quality Assessment

Speech quality assessment plays an important role in the development of speech processing systems such as text-to-speech (TTS) synthesis. Speech quality is a complex, subjective psychoacoustic outcome of human perception. The mean opinion score (MOS) is a commonly used metric to evaluate the speech quality in TTS, which is obtained by having human listeners rate the perceived quality of the synthesised speech on a numerical scale typically ranging from 1 to 5, where a higher score indicates better-perceived speech quality, then average the scores across all listeners. Apart from estimating the MOS (*i.e.*, the average score), it is also worthwhile considering the annotator scoring range and distribution to take into account the subjective nature of individual preferences, perceptions and biases. The proposed method is a cost-effective alternative to the time-consuming and expensive human assessment of speech quality which models the subjectivity that different human listeners may have.

TTS MOS Dataset

The SOMOS dataset (Maniati et al., 2022) is used in this experiment. The speech dataset consists of 20k+ utterances generated from 200 TTS systems along with the natural LJ Speech[4] (Ito and Johnson, 2017) and 2,000 unique sentences which on average have 10 words. The SOMOS dataset is annotated using crowd-sourcing. Each audio segment is evaluated by at least 17 unique annotators out of 987 participated human annotators, and there are 17.9 annotations per segment on average. The human annotators were asked to evaluate the naturalness of each audio sample on a 5-point Likert scale from 1 (very unnatural) to 5 (completely natural). The standard split provided by the dataset is used, which contains 141,100 training segments, 3,000 validation segments and 3,000 test segments.

7.4 Experimental Setup and Metrics

7.4.1 Backbone Architecture

A neural-network-based encoder $g_{\mathbf{\Lambda}}$ is built to model $\boldsymbol{\mu}_{\mathbf{\Lambda}}(\boldsymbol{x}), \boldsymbol{\sigma}^2_{\mathbf{\Lambda}}(\boldsymbol{x})$ given input $\boldsymbol{x}$ where $\mathbf{\Lambda}$ is the model parameters. $g_{\mathbf{\Lambda}}$ follows an upstream-downstream paradigm. The upstream

[4]Details about the LJ Speech dataset can be found in Appendix B.4.

Table 7.1 Configuration of the model structure (number of layers * layer dimension).

Task	Input modality	g_{Λ}-upstream	g_{Λ}-downstream	f_{θ}	h_{Ω}
Emotion class labelling	speech	WavLM Base+	2*128	3*64	1*64
Toxic speech detection	text	RoBERTa Base	2*128	3*64	1*64
Speech quality assessment	speech	WavLM Base+	2*128	3*16	/
Emotion attribute prediction	speech	WavLM Base+	2*128	3*64	/

uses a foundation model pretrained on a large amount of unlabelled data to learn universal representations. The downstream model uses the learned representation from the upstream model for specific applications.

For tasks involving speech as input (*i.e.*, emotion annotation, speech quality assessment), WavLM Base+[5] (see Section 2.2.4) was used as the upstream model. The parameters of the pretrained WavLM were frozen and the weighted sum of the outputs of the 12 Transformer encoder blocks were used as the speech embeddings feeding into the downstream model. RoBERTa Base[6] (see Section 2.2.3) was used as upstream model to encode text input for toxic speech detection, which has 12 Transformer layers, 768 hidden units, and 12 attention heads. The RoBERTa encoder was frozen.

The downstream model consisted of two Transformer encoder blocks followed by two FC layers. The Transformer encoder layers have a dimension of 128 and four attention heads. The output layer contains two heads to predict the mean and standard deviation of the latent distribution $\mathrm{p}_{\boldsymbol{\Lambda}}(\boldsymbol{z}|\boldsymbol{x})$.

The invertible flow model $\boldsymbol{f_{\theta}}$ uses the real NVP blocks (Dinh et al., 2017). The variational encoder for S-CNF $\boldsymbol{h_{\Omega}}$ contains a FC layer and two output heads for the mean and standard deviation of the variational distribution $\boldsymbol{q_{\Omega}}(\boldsymbol{v}|\boldsymbol{y})$. More detail can be found in Table 7.1.

7.4.2 Baselines

The proposed I-CNF and S-CNF were compared to baselines of various types such as ensemble methods, Bayesian methods, and conditional generative models:

- Deep ensemble (Ensemble) (Lakshminarayanan et al., 2017) which consists of 10 systems initialised and trained using different random seeds;
- Monte Carlo dropout (MCDP) (Gal and Ghahramani, 2016) with a dropout rate of 0.4;
- Bayes-by-backprop (BBB) (Blundell et al., 2015) with a standard Gaussian prior;

[5]Available at: https://huggingface.co/microsoft/wavlm-base-plus
[6]Available at: https://huggingface.co/roberta-base

- Conditional variational autoencoder (CVAE) (Kingma and Welling, 2014) which has the same $\boldsymbol{g}_{\boldsymbol{\Lambda}}$ structure as S-CNF for modelling $\mathrm{p}(\boldsymbol{z}|\boldsymbol{x})$ and two 64-d FC layers for the encoder and the decoder;
- Conditional argmax flow (A-CNF) (Hoogeboom et al., 2021) with an identical model structure to S-CNF;
- Gaussian process (GP) (Williams and Rasmussen, 2006) with a radial basis function kernel which takes features extracted from the upstream model as input and is trained by maximising the per-observation-based marginal likelihood;
- EDL-based systems described in Chapter 5 and Chapter 6 for categorical and continuous labels respectively, which are trained by maximising the per-observation-based marginal likelihood with a modified regularisation term.

Ensemble, MCDP, BBB and EDL used the same model structure as $\boldsymbol{g}_{\boldsymbol{\Lambda}}$ apart from removing the output head for predicting variance of latent distribution. $M_* = 100$ samples were used to compute evaluation metrics at test time. The Ensemble only consists of 10 systems due to its expensive computational cost.

The system was trained for 30 epochs and the model with the best validation performance was used for testing. The number of ELBO samples was set to 20. Experiments were run for three different seeds and the standard error is reported along with the average.

7.4.3 Evaluation Metrics

Several metrics are adopted to measure the empirical performance of the HAS system in terms of mean/majority prediction, distribution matching, and human variability simulation.

Mean/Majority Prediction

For ordinal annotations, the root mean squared error is used to evaluate the quality of the mean prediction for all test inputs: $\mathrm{RMSE}^{\bar{\boldsymbol{y}}} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\bar{\boldsymbol{y}}_i - \bar{\boldsymbol{\eta}}_i)^2}$, where $\bar{\boldsymbol{y}}_i = \frac{1}{M_*}\sum_{m=1}^{M_*}\boldsymbol{y}_i^{(m)}$, $\bar{\boldsymbol{\eta}}_i = \frac{1}{M_i}\sum_{m=1}^{M_i}\boldsymbol{\eta}_i^{(m)}$, and N is the number of test samples. For categorical annotations, the classification accuracy (ACC) for the majority vote is evaluated for all test inputs that have majority human annotations.

Distribution Matching

The negative log likelihood (NLL) is used to evaluate how well the model estimates the human annotation distribution: $\mathrm{NLL}^{\mathrm{all}} = -\frac{1}{N}\sum_{i=1}^{N}\left(\frac{1}{M_i}\sum_{m=1}^{M_i}\log \mathrm{p}_{\boldsymbol{\theta}}(\boldsymbol{\eta}_i^{(m)}|\boldsymbol{x}_i)\right)$.

Inter-Annotator Disagreement Simulation

Apart from evaluating the goodness of fit, additional metrics are adopted to explicitly measure how well the model simulates the variability and disagreements in human annotations:

- The root mean squared error of the standard deviations of the annotations for all test inputs:

$$\text{RMSE}^s = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\sigma_i - s_i)^2}, \tag{7.10}$$

  where for ordinal annotations

$$\sigma_i = \sqrt{\frac{1}{M_i}\sum_{m=1}^{M_i}(\boldsymbol{\eta}_i^{(m)} - \bar{\boldsymbol{\eta}}_i)^2}, \quad s_i = \sqrt{\frac{1}{M_*}\sum_{m=1}^{M_*}(\boldsymbol{y}_i^{(m)} - \bar{\boldsymbol{y}}_i)^2} \tag{7.11}$$

  and for categorical annotations

$$\sigma_i = \frac{1}{K}\sum_{k=1}^{K}\sqrt{\frac{1}{M_i}\sum_{m=1}^{M_i}(\boldsymbol{\eta}_{i,k}^{(m)} - \bar{\boldsymbol{\eta}}_{i,k})^2}, \quad s_i = \frac{1}{K}\sum_{k=1}^{K}\sqrt{\frac{1}{M_*}\sum_{m=1}^{M_*}(\boldsymbol{y}_{i,k}^{(m)} - \bar{\boldsymbol{y}}_{i,k})^2}. \tag{7.12}$$

  where K denotes the number of classes, $\bar{\boldsymbol{\eta}}_i$ is the average of human annotations for event $\boldsymbol{d}_i$, $\boldsymbol{y}_i^{(m)}$ is a simulated annotation, $\bar{\boldsymbol{y}}_i$ is the average of simulated annotations for event $\boldsymbol{d}_i$, and M_* is the number of simulated annotations;

- The absolute error of the average standard deviations of the annotations for all test inputs: $\mathcal{E}(\bar{s}) = |\bar{\sigma} - \bar{s}|$, where $\bar{\sigma} = \sum_{i=1}^{N}\sigma_i$ and $\bar{s} = \sum_{i=1}^{N} s_i$;

- The absolute error of inter-annotator disagreement levels. For categorical annotations, Fleiss' kappa ($\boldsymbol{\kappa}$) (Fleiss, 1971) is adopted where $\boldsymbol{\kappa}$ is a real number between -1 and $+1$, with -1 indicating no observed agreement and $+1$ indicating perfect agreement. The absolute error between the kappas of human annotations ($\boldsymbol{\kappa}$) and simulated annotations ($\hat{\boldsymbol{\kappa}}$) for all test inputs is reported: $\mathcal{E}(\hat{\boldsymbol{\kappa}}) = |\hat{\boldsymbol{\kappa}} - \boldsymbol{\kappa}|$. For ordinal annotations, intra-class correlation coefficient (ICC) (Shrout and Fleiss, 1979) is adopted which ranges from 0 to 1. The absolute error $\mathcal{E}(\text{ICC})$ between the ICC(1,k) of human annotations and simulated annotations ($\hat{\boldsymbol{\kappa}}$) is reported.

7.5 Experimental Results

7.5.1 I-CNF for Ordinal Annotations

Performance

Table 7.2 and Table 7.3 report the test results for all compared methods for speech quality assessment and emotion attribute prediction respectively. The proposed I-CNF achieves competitive performance for mean prediction. More importantly, I-CNF obtains the best performance for distribution match (in terms of NLL^{all}) and inter-annotator disagreement simulation (measured by RMSE^{s}, $\mathcal{E}(\bar{s})$ and $\mathcal{E}(\text{ICC})$) among all of the compared methods for both tasks.

Table 7.2 Test performance on the speech quality assessment task. "↓" denotes the lower the better. The best value in each column is shown in bold, and the second-best value is underlined.

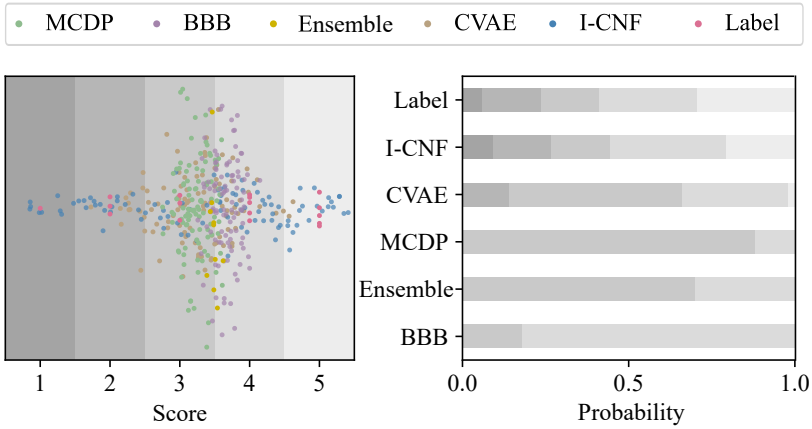
	$\text{RMSE}^{\bar{y}} \downarrow$	$\text{NLL}^{\text{all}} \downarrow$	$\text{RMSE}^{s} \downarrow$	$\mathcal{E}(\bar{s}) \downarrow$	$\mathcal{E}(\text{ICC}) \downarrow$
GP	**0.359±0.001**	1.693±0.000	0.472±0.000	0.412±0.000	0.433±0.000
EDL	0.449±0.023	<u>1.636±0.001</u>	<u>0.375±0.022</u>	<u>0.356±0.025</u>	<u>0.107±0.029</u>
MCDP	<u>0.390±0.013</u>	1.787±0.008	0.783±0.035	0.742±0.031	0.495±0.010
Ensemble	0.410±0.008	1.858±0.000	0.740±0.007	0.704±0.006	0.136±0.028
BBB	0.613±0.011	1.934±0.015	0.944±0.017	0.918±0.017	0.480±0.003
CVAE	0.419±0.013	1.703±0.022	0.598±0.033	0.561±0.035	0.214±0.028
I-CNF	<u>0.392±0.016</u>	**1.609±0.003**	**0.251±0.007**	**0.123±0.013**	**0.079±0.015**

Table 7.3 Test performance on the emotion attribute annotation task. "↓" denotes the lower the better. The best value in each column is shown in bold, and the second-best value is underlined.

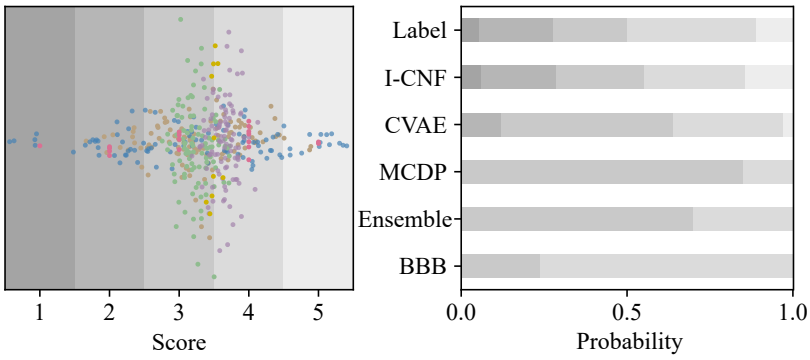
	$\text{RMSE}^{\bar{y}} \downarrow$	$\text{NLL}^{\text{all}} \downarrow$	$\text{RMSE}^{s} \downarrow$	$\mathcal{E}(\bar{s}) \downarrow$	$\mathcal{E}(\text{ICC}) \downarrow$
GP	<u>0.667±0.000</u>	2.928±0.000	<u>0.408±0.000</u>	0.415±0.000	0.169±0.000
EDL	0.755±0.002	<u>1.911±0.005</u>	0.465±0.039	0.504±0.037	0.172±0.017
MCDP	0.887±0.007	5.545±0.026	0.610±0.005	0.474±0.006	0.087±0.014
Ensemble	0.923±0.017	6.280±0.084	0.836±0.017	0.718±0.019	<u>0.057±0.003</u>
BBB	0.720±0.014	5.332±0.034	0.643±0.001	0.516±0.001	0.241±0.003
CVAE	0.704±0.004	4.906±0.005	0.502±0.003	<u>0.324±0.003</u>	0.192±0.003
I-CNF	**0.665±0.006**	**1.707±0.030**	**0.296±0.019**	**0.132±0.002**	**0.032±0.012**

Case Study

To better illustrate the properties of the annotations simulated by different methods, simulated distributions were visualised against the ground-truth distributions for two representative examples of speech quality assessment in Fig. 7.5. It can be seen that the proposed I-CNF is the only method which gives an accurate distribution match and good inter-annotator disagreement simulation in both cases. In contrast, all the other methods tend to either produce annotations centred around the mean score or collapse to one score (typically 3 or 4). More case study examples can be found in Appendix E.1.



(a) Utterance "conv_2007_0090_057"



(b) Utterance "conv_2007_0090_060"

Fig. 7.5 Visualisation of simulated annotations on the speech quality assessment task for case study. For the visualisation purpose, the points that have same x values are spread along y-axis according to density to avoid overlapping.

7.5.2 S-CNF for Categorical Annotations

Performance

Table 7.4 and Table 7.5 report the test results for all of the compared methods for emotion class labelling and toxic speech detection. Ensemble achieves the best majority prediction accuracy (ACC) at the cost of training 10 independent systems. The proposed S-CNF achieves the second-best majority prediction accuracy with only a tenth of the computational cost of Ensemble. More importantly, S-CNF is the best at matching the distributions of human annotations (in terms of NLL^{all}) and simulating inter-annotator disagreements (measured by RMSE^{s}, $\mathcal{E}(\bar{s})$ and $\mathcal{E}(\hat{\kappa})$) among all of the compared methods.

Table 7.4 Test performance on the emotion category annotation task. CVAE collapses to one category for all inputs. "↑" denotes the higher the better. "↓" denotes the lower the better. The best value in each column is shown in bold, and the second-best value is underlined.

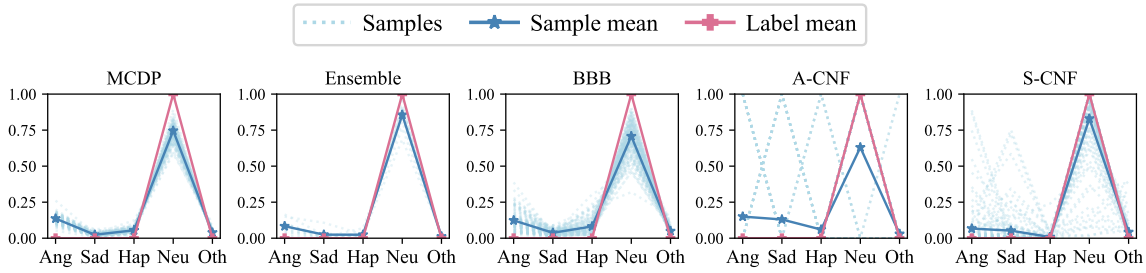
	ACC ↑	**NLL**$^{\text{all}}$ ↓	**RMSE**^{s} ↓	$\mathcal{E}(\bar{s})$ ↓	$\mathcal{E}(\hat{\kappa})$ ↓
EDL	0.583±0.005	<u>1.417±0.003</u>	0.266±0.004	0.175±0.002	<u>0.123±0.012</u>
MCDP	0.582±0.003	1.423±0.012	0.294±0.001	0.193±0.000	0.467±0.005
Ensemble	**0.603±0.002**	1.458±0.004	0.271±0.003	0.160±0.004	0.344±0.017
BBB	0.565±0.010	1.459±0.011	0.289±0.005	0.187±0.008	0.511±0.034
CVAE	0.275±0.000	1.661±0.000	0.333±0.000	0.244±0.000	—
A-CNF	0.583±0.002	1.430±0.006	<u>0.239±0.001</u>	<u>0.097±0.002</u>	0.382±0.015
S-CNF	<u>0.591±0.002</u>	**1.403±0.011**	**0.218±0.000**	**0.020±0.002**	**0.068±0.021**

Table 7.5 Test performance on the toxic speech detection task. CVAE collapses to one category for all inputs. "↑" denotes the higher the better. "↓" denotes the lower the better. The best value in each column is shown in bold, and the second-best value is underlined.

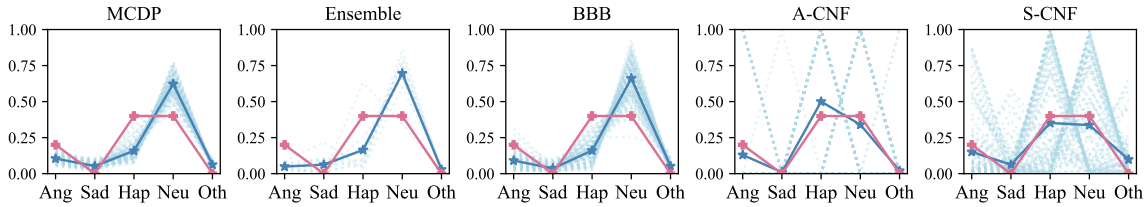
	ACC ↑	**NLL**$^{\text{all}}$ ↓	**RMSE**^{s} ↓	$\mathcal{E}(\bar{s})$ ↓	$\mathcal{E}(\hat{\kappa})$ ↓
EDL	0.670±0.006	0.908±0.003	<u>0.276±0.001</u>	0.093±0.001	0.092±0.009
MCDP	0.656±0.009	0.951±0.032	0.300±0.002	0.129±0.003	0.143±0.008
Ensemble	**0.682±0.002**	0.909±0.012	0.289±0.001	0.100±0.003	<u>0.064±0.006</u>
BBB	0.670±0.001	0.949±0.021	0.300±0.009	0.127±0.022	0.207±0.051
CVAE	0.406±0.000	1.150±0.000	0.345±0.000	0.208±0.000	—
A-CNF	0.628±0.003	<u>0.892±0.011</u>	0.297±0.001	<u>0.087±0.008</u>	0.198±0.027
S-CNF	<u>0.673±0.002</u>	**0.837±0.008**	**0.263±0.001**	**0.002±0.001**	**0.026±0.012**

Case Study

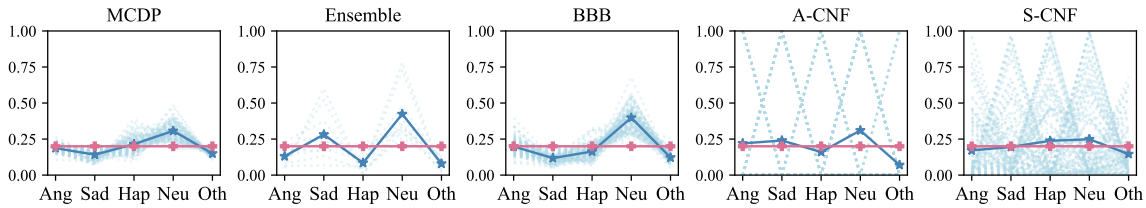
To better illustrate the properties of the annotations simulated by different methods, the simulated distributions against the ground-truth distributions for three representative examples are visualised in Fig. 7.6 (more case study examples can be found in Appendix E.2). Overall, the mean of the samples generated by S-CNF aligns the best with the average human label, indicating its superior performance in estimating the aggregated behaviour of human annotators. Interestingly, the samples generated by S-CNF are the most diverse among all compared methods, which manage to simulate the variability of the behaviour of different individual human annotators. In sharp contrast, the samples generated by all the other methods are highly concentrated around their sample means. The visualised result for each example is analysed below:



(a) Utterance "MSP-PODCAST_0114_0263.wav"



(b) Utterance "MSP-PODCAST_0167_0179_0001.wav"



(c) Utterance "MSP-PODCAST_0574_0476.wav"

Fig. 7.6 Visualisation of simulated annotations on the emotion category annotation task for case study. The y-axis corresponds to the probability mass. Each sample is a categorical distribution. The probability mass values of different categories in each categorical distribution are connected for the purpose of better visualisation. CVAE is omitted because it collapses to one category for all inputs.

(a) Human annotators reach a consensus in this case. The majority of samples generated by S-CNF exhibit prominent peaks aligned with the ground-truth emotion class "neutral". In contrast, many samples generated by A-CNF peak at other emotion classes.

(b) Human opinions diverge in this case. The majority of samples generated by S-CNF are sharp categorical distributions peaking at one of the two majority emotion classes "happy" and "neutral". Additionally, a few samples generated by S-CNF peak at the emotion class "angry", which manages to simulate the minority viewpoint held by some annotators. Very few human annotators attribute this utterance to the emotion classes "sad" and "other", and S-CNF likewise produces scarce samples peaking at these classes.

(c) Five human annotators give distinct emotion labels in this case, resulting in a tie in the label means. The tie comes from annotators' diverse individual perceptions of the emotion rather than consensus on its ambiguity. S-CNF is the only model that can simulate both the diverse behaviours of different individual annotators and the aggregated behaviour of all annotators since the individual samples are sharp categorical distributions peaking at one of the five emotion classes and the mean of the samples aligns well with the label mean.

7.5.3 Computational Time Cost

The computational time cost of all of the methods that have been compared for the three tasks studied in the chapter are shown in Table 7.6 and Table 7.7 . Denote M_* as the number of annotations to be simulated. The ensemble model with M_* members involves training and testing M_* individual models, which increases the training time by $M_*\times$ and the inference time by $M_*\times$. MCDP and BBB require M_* forward passes during inference to generate M_* samples and therefore cost $M_*\times$ inference time. All other methods require a single forward pass. In contrast to neural-network-based methods of complexity $O(n^2)$, the training and inference of GP involves matrix inversion of complexity $O(n^3)$.

7.5.4 Analysis

This section provides analysis in order to give a better understanding of the proposed method. The results of one run are reported due to computational costs.

Table 7.6 Computational time cost (sec) of speech quality assessment and emotion attribute annotation. Due to training complexity, the number of annotations it simulate M_* is set to 10 for ensemble while 100 for all other methods.

	Speech quality assessment		**Emotion attribute annotation**	
	Training	**Inference**	**Training**	**Inference**
GP	3.88±0.01E+03	6.27±0.07E+01	1.00±0.00E+04	2.61±0.03E+02
EDL	2.92±0.01E+03	5.17±0.19E+01	7.69±0.03E+03	1.91±0.00E+02
MCDP	1.37±0.32E+03	3.64±1.29E+03	3.89±0.01E+03	1.76±0.03E+04
Ensemble	1.39±0.00E+04	5.10±0.05E+02	3.91±0.01E+04	1.67±0.00E+03
BBB	1.51±0.00E+03	5.33±0.02E+03	4.25±0.01E+03	1.81±0.01E+04
CVAE	1.41±0.00E+03	5.27±0.06E+01	4.13±0.00E+03	2.26±0.05E+02
I-CNF	1.34±0.07E+03	5.10±0.02E+01	3.98±0.08E+03	1.76±0.00E+02

Table 7.7 Computational time cost (sec) of emotion class annotation and toxic speech detection. Due to training complexity, the number of annotations it simulate M_* is set to 10 for ensemble while 100 for all other methods.

	Emotion category annotation		**Toxic speech detection**	
	Training	**Inference**	**Training**	**Inference**
EDL	6.78±0.01E+03	2.90±0.01E+02	1.90±0.01E+02	2.67±0.02E+01
MCDP	7.20±0.10E+03	1.82±0.01E+04	2.42±0.02E+02	5.99±0.02E+02
Ensemble	1.46±0.00E+05	1.67±0.01E+03	2.39±0.01E+03	4.00±0.04E+01
BBB	7.55±0.01E+03	1.79±0.01E+04	3.22±0.01E+02	5.79±0.01E+02
A-CNF	7.04±0.02E+03	2.31±0.07E+02	3.14±0.04E+02	1.40±0.11E+01
S-CNF	6.99±0.00E+03	2.12±0.02E+02	2.63±0.02E+02	1.37±0.09E+01

Analysis of Standard Deviation of Simulated Samples

It has been observed in previous sections that flow models tend to have a larger difference between RMSE^s and $\mathcal{E}(\bar{s})$. This section provides a detailed analysis of this observation. Let N be the number of test utterances. Three standard deviation (std) related metrics are computed: (i) RMSE between std of predictions and human labels: $\text{RMSE}^s = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(s_i - \sigma_i)^2}$; (ii) Mean absolute error between std of predictions and std of human labels: $\text{MAE}^s = \frac{1}{N}\sum_{i=1}^{N}|s_i - \sigma_i|$; (iii) Absolute error between average std of predictions and average std of human labels $\mathcal{E}(\bar{s}) = |\bar{s}_i - \bar{\sigma}_i|$. The results are shown in Table 7.8.

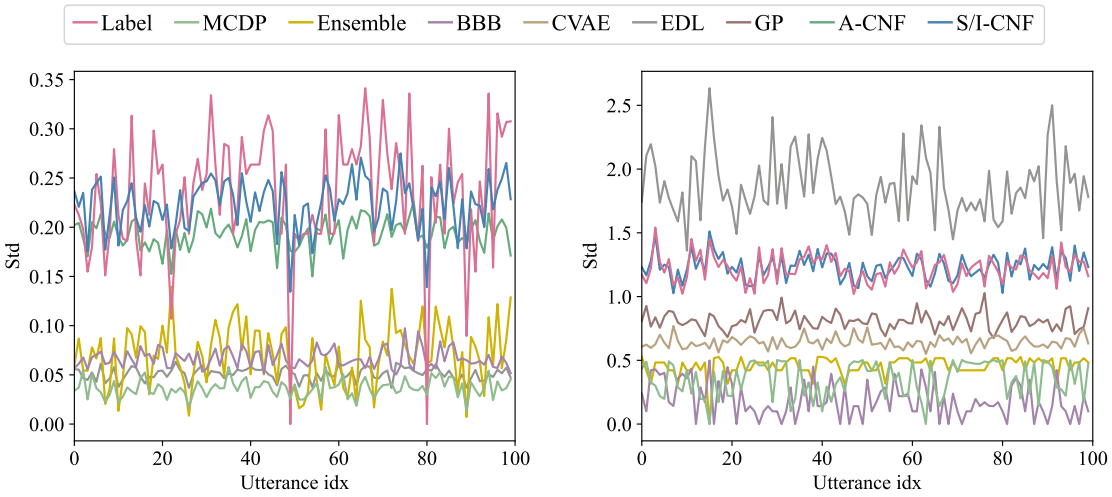
Table 7.8 Analysis of standard deviation of simulated samples.

	Emotion class labelling			**Speech quality**		
	RMSE^s	**MAE**^{s}	$\mathcal{E}(\bar{s})$	**RMSE**^{s}	**MAE**^{s}	$\mathcal{E}(\bar{s})$
MCDP	0.305	0.233	0.206	0.809	0.762	0.762
Ensemble	0.277	0.222	0.166	0.747	0.703	0.703
BBB	0.284	0.226	0.178	0.952	0.917	0.917
CVAE	0.333	0.244	0.244	0.574	0.535	0.534
EDL	0.267	0.200	0.133	0.381	0.368	0.368
GP		/		0.472	0.419	0.412
A-CNF	0.223	0.209	0.046		/	
S/I-CNF	0.218	0.198	0.015	0.229	0.184	0.067

The flow model tends to have larger discrepancy between MAE^s and $\mathcal{E}(\bar{s})$. According to the triangular inequality:

$$\mathcal{E}(\bar{s}) = \left| \frac{1}{N}\sum_{i=1}^{N} s_i - \frac{1}{N}\sum_{i=1}^{N} \sigma_i \right| = \left| \frac{1}{N}\sum_{i=1}^{N} (s_i - \sigma_i) \right| \leq \frac{1}{N}\sum_{i=1}^{N} |s_i - \sigma_i| = \text{MAE}^s \tag{7.13}$$

which shows that $\mathcal{E}(\bar{s})$ is a lower bound of MAE^s. The equality condition is satisfied when all samples are uniformly either greater than or less than the compared value. Therefore, a larger discrepancy between these two values indicates that the standard deviation of some samples exceeds that of the labels, while for others, it is lower. A smaller discrepancy indicates that the standard deviation of samples tend to be consistently larger of smaller than that of the labels. In Fig. 7.7, 100 test utterances were randomly selected and the std of samples generated by different models are plotted, which supports the above conclusion. The proposed S-CNF and I-CNF has the best performance for matching the diversity of human annotations.

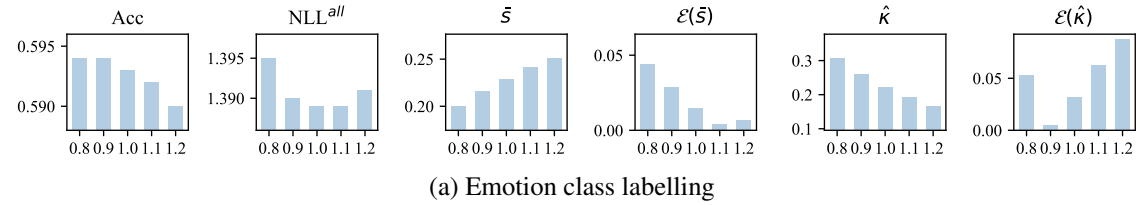


(a) Std for emotion class labelling. (b) Std for speech quality evaluation.

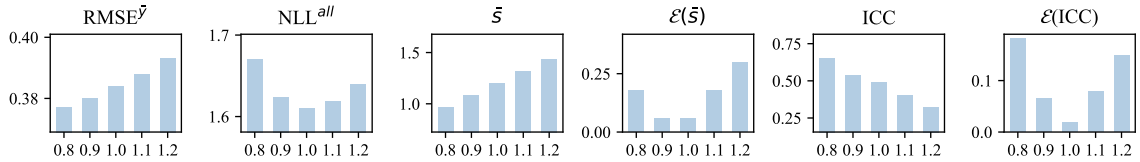
Fig. 7.7 Standard deviation of simulated samples.

Adjusting Diversity of CNFs by Prior Tempering

One advantage of the CNF approach is that the sample diversity can be easily controlled on demand without re-training by tempering the standard deviation of $\mathrm{p}_{\boldsymbol{\Lambda}}(\boldsymbol{z}|\boldsymbol{x})$ at test time. Fig. 7.8 explores the effect of prior tempering on performance. More detail is shown in Table 7.9. Overall, the trend is clear that the simulated annotations become more diverse as the temperature increases. The default temperature value 1 used during training (*i.e.*, no tempering) achieves the best trade-off among majority prediction accuracy (ACC), distribu-



(a) Emotion class labelling



(b) Speech quality assessment

Fig. 7.8 The effect of prior tempering on the performance of S-CNF and I-CNF. The x-axis corresponds to the prior temperature. The other two tasks show similar trend and is omitted here.

Table 7.9 Adjusting the diversity of CNFs by prior tempering (T).

T	Emotion class ACC	Emotion class $\bar{s}$	Toxic speech ACC	Toxic speech $\bar{s}$	Speech quality $\text{RMSE}^{\bar{y}}$	Speech quality $\bar{s}$	Emotion attribute $\text{RMSE}^{\bar{y}}$	Emotion attribute $\bar{s}$
0.8	0.594	0.200	0.671	0.125	0.377	0.963	0.650	0.735
0.9	0.594	0.216	0.675	0.157	0.380	1.083	0.654	0.924
1.0	0.593	0.229	0.673	0.188	0.384	1.201	0.658	1.171
1.1	0.592	0.241	0.671	0.216	0.388	1.322	0.662	1.314
1.2	0.590	0.251	0.669	0.242	0.393	1.440	0.665	1.570

Table 7.10 Adjusting the diversity of MCDP models by dropout rate (DP).

DP	Emotion class ACC	Emotion class $\bar{s}$	Toxic speech ACC	Toxic speech $\bar{s}$	Speech quality $\text{RMSE}^{\bar{y}}$	Speech quality $\bar{s}$	Emotion attribute $\text{RMSE}^{\bar{y}}$	Emotion attribute $\bar{s}$
0.1	0.583	0.040	0.661	0.049	0.385	0.180	0.899	0.432
0.2	0.589	0.040	0.666	0.061	0.412	0.236	0.884	0.486
0.3	0.590	0.045	0.654	0.081	0.408	0.294	0.868	0.504
0.4	0.585	0.051	0.662	0.085	0.367	0.227	0.871	0.527
0.5	0.589	0.053	0.662	0.088	0.356	0.278	0.880	0.595

tion matching (NLL^{all}), and inter-annotator disagreement simulation (in terms of $\mathcal{E}(\bar{s})$ and $\mathcal{E}(\hat{\kappa})$). In addition, as compared in Table 7.10, prior tempering in CNF is more efficient and covers a wider range of dynamics than adjusting the dropout rate in MCDP.

7.6 Limitations and Ethics Statement

Before concluding the chapter, it is worth discussing the potential concerns associated with training models on labels generated by models. AI models, being data-driven, are heavily reliant on the quality of training data. Biased, incomplete, or inaccurate data could put the model at the risk of reinforcing and even amplifying these biases, leading to unintended consequences. The complexity of modern large neural network models can lead to a lack of interpretability and explainability, making it difficult to understand how they arrive at their decisions. This black box problem can limit the ability to identify and mitigate potential biases or errors in the system. Despite the rapid evolution of AI technology, over-reliance on these systems remains a cause for concern.

7.7 Chapter Summary

Human annotator simulation (HAS) serves as a cost-effective substitute for human evaluation such as data annotation and system assessment. Human perception and behaviour during human evaluation exhibits inherent variability due to diverse cognitive processes and subjective interpretations, which should be taken into account in modelling to better mimic the way people perceive and interact with the world. This chapter introduces a novel meta-learning framework that treats HAS as a zero-shot density estimation problem, which incorporates human variability. This overcomes the drawbacks of prior work and allows for the efficient generation of human-like annotations for unlabelled test inputs. In this framework, a meta-learning objective has been derived for two new model classes, conditional integer flows and conditional softmax flows, to account for ordinal and categorical annotations, respectively. The proposed method consistently and significantly outperforms a wide range of methods on three real-world human evaluation tasks, showing a superior ability and efficiency to predict the aggregated behaviour of human annotators, match the distribution of human annotations, and simulate inter-annotator disagreements. It is hoped that this work could help mitigate unfair biases and over-representation in HAS and reduce the exposure of human annotators to potentially harmful content, thus promoting ethical AI practices.

Chapter 8

Detection of Mental Disorders and Cognitive Diseases

In recent years, awareness and concern for mental health, such as Alzheimer's disease (AD) and depression, have significantly increased. This chapter discusses the use of speech information for automatic detection of depression and Alzheimer's disease. Depression is a widespread mental disorder that is manifest through ongoing sadness, a lack of interest in activities previously enjoyed, and also diminished thinking capabilities. According to WHO, depression affects about 280 million people in the world (World Health Organisation, 2024) involving all age groups and cultural backgrounds, potentially causing persistent thoughts of death or suicidal ideation (Edition et al., 2013). Despite this prevalence, the diagnosis of depression are by nature difficult and time consuming as there is no single clinical characterisation of a depressed individual. At present, there is no objective measure for depression detection with clinical utility (Cummins et al., 2015). Dementia is a category of neurodegenerative diseases that entails a long-term and usually gradual decrease of cognitive functioning. Alzheimer's disease is the leading cause of dementia, impacting millions of people across the world (Mattson, 2004).

The precise diagnosis of AD and depression is essential for their effective management and the initiation of timely intervention. This has driven forward research in the automatic detection of AD (Ivanov et al., 2013, Mirheidari et al., 2018, Cui et al., 2023) and depression (Moore II et al., 2007, Yu et al., 2015, He et al., 2021, Wu et al., 2023c). Studies have explored a variety of hand-crafted features, including acoustic aspects like pitch variation, syllable rate, and spectrogram analysis (Moore II et al., 2007, Low et al., 2010, Ooi et al., 2012, Ivanov et al., 2013, Yu et al., 2015, Mirheidari et al., 2018), along with linguistic aspects such as part-of-speech information, sentence structure, and vocabulary diversity (Bucks et al., 2000, Fraser et al., 2016, Yang et al., 2016, Gong and Poellabauer, 2017). The advent

of deep learning pretrained speech and language models, such as WavLM, Whisper, and BERT, has offered promising results for extracting features relevant to diagnosing AD and depression (Balagopalan et al., 2020, Syed et al., 2020, Wu et al., 2022c, Yuan et al., 2020, Cui et al., 2023, Wu et al., 2023c).

The rest of this chapter is organised as follows. Section 8.1 introduces background information including approaches to detecting depression and AD. Section 8.2 presents the benchmark datasets used in this chapter. Section 8.3 discusses current challenges in automatic detection of mental disorders. Section 8.4 uses foundation models for detecting depression via spontaneous speech. Section 8.5 applies the EDL methods introduced in Chapter 5 to confidence estimation for automatic diagnosis of depression and AD.

8.1 Background

8.1.1 Depression

Depression (also called major depressive disorder or clinical depression) is a common but serious mood disorder. Typical symptoms include depressed mood and/or markedly diminished interest or pleasure in combination with four of: (i) psychomotor retardation or agitation; (ii) diminished ability to think/concentrate or increased indecisiveness; (iii) fatigue or loss of energy; (iv) insomnia or hypersomnia; (v) significant weight loss or weight gain; (vi) feelings of worthlessness or excessive/inappropriate guilt; (vii) recurrent thoughts of death or recurrent suicidal ideation (Edition et al., 2013). Whilst many people feel some form of depression in their life, it is considered an illness when an individual has these symptoms for longer than a two-week period.

Diagnosis of depression is complex. It relies heavily on the ability, desire and honesty of a patient to communicate their symptoms, moods or cognitions when, by definition, their outlook and motivation are impaired. This makes diagnostic information time consuming to gather and requires a large degree of clinical training, practice, and certification to produce acceptable results. Currently there is no objective measure, with clinical utility, for depression (Cummins et al., 2015). Depression is commonly assessed by clinical interview along with rating scales such as Hamilton rating scale for depression (Hamilton, 1986), Beck depression index (Beck et al., 1996), patient health questionnaire (Kroenke et al., 2001), *etc*.

To enhance current diagnostic methods, an objective screening mechanism is necessary. A wide range of biological markers have been investigated to be associated with depression such as neurotransmitter dysfunction (Luscher et al., 2011) and genetic abnormalities (Gatt et al., 2009). However, no specific bio-marker has been found to date. Recent advances have

been made in using affective computing and social signal processing as a diagnostic tool for depression (Cohn et al., 2009, Cummins et al., 2013, Joshi et al., 2013, Scherer et al., 2013, Williamson et al., 2013) which rely in particular on facial and body tracking algorithms to capture characteristic behavioural changes relating to depression.

Automatically detecting mental illness using speech, more specifically non-verbal paralinguistic cues, has gained popularity in recent years. Speech can be collected cheaply, remotely, non-invasively and non-intrusively which makes it an attractive candidate for use in an automated system. Depressed individuals usually exhibit decreased verbal activity productivity, a diminished prosody and monotonous and "lifeless" sounding speech (Hall et al., 1995, Sobin and Sackeim, 1997). A wide range of audio features have been trialled for automatic depressed speech classification including prosodic, voice quality, spectral, glottal features and the combination of them (Moore II et al., 2007, Low et al., 2010, Cummins et al., 2011, Ooi et al., 2012, Alghowinem et al., 2012, 2013a,b). Apart from audio information, text (Williamson et al., 2016, Yang et al., 2016, Sun et al., 2017, Gong and Poellabauer, 2017) and video (Pampouchidou et al., 2017, He et al., 2021) also provide useful information for depression detection. Inspired by emerging deep learning techniques, various neural network structure and the integration of multi-modal features through deep learning models has been found to be promising for depression detection (Ma et al., 2016, Yang et al., 2017, Al Hanai et al., 2018, Haque et al., 2018, Ray et al., 2019).

8.1.2 Alzheimer's Disease

Alzheimer's disease (AD) is a chronic neurodegenerative disease with a progressive pattern of cognitive and functional impairment. Symptoms like language impairment, memory loss, self-neglect, and behaviour issues are found in patients as the disease worsens (Mattson, 2004). AD is currently incurable, but timely intervention can effectively decelerate progression. Therefore, the detection of AD is crucial and attracts extensive attention worldwide (Ritchie et al., 2017).

Conventional methods for AD detection are mainly based on clinical tests for cognitive decline and independence in everyday activities (Velayudhan et al., 2014). The most commonly used neuropsychological assessments include mini-mental state examination (Folstein et al., 1975), clinical dementia rating (Morris, 1991), general practitioner assessment of cognition (Brodaty et al., 2002), Montreal cognitive assessment (Nasreddine et al., 2005) and hierarchical dementia scale-revised (Cole et al., 2015). However, these diagnostic processes are constrained due to time requirements and accessibility of resources. As the number of people diagnosed with AD is rapidly increasing, the high prevalence of the disease and the

high costs associated with traditional approaches to detection stimulates the research on automatic detection of AD (Zeisel et al., 2020).

AD can be detected from brain imaging exams such as computed tomography (CT) , magnetic resonance imaging (MRI) or positron emission tomography (PET). Many current AD detection studies use medical imaging with various machine learning techniques and deep neural network models (Li et al., 2007, Termenon et al., 2013, Moradi et al., 2015, Zhang et al., 2015, Mirzaei et al., 2016, Sarraf and Tofighi, 2016, Lu et al., 2018, Pellegrini et al., 2018, Ortiz et al., 2018).

Throughout the course of AD, patients have been observed suffering a loss of lexical-semantic skills, including suffering anomia, reduced word comprehension, object naming problems, semantic paraphasia, and a reduction in vocabulary and verbal fluency (Bayles and Boone, 1982, Forbes-McKay and Venneri, 2005). Speech in patients with AD is mostly characterised by a low speech rate and frequent hesitations at the phonetic and phonological level (Kavé and Levy, 2003). Since spoken language is an easily captured signal that can reflect the speaker's cognitive abilities, researchers have been motivated to investigate the use of speech and language features as biomarkers for AD detection (Szatloczki et al., 2015, Weiner et al., 2019). Various machine learning models have been adopted for AD detection using acoustic information (Yu et al., 2015, Ivanov et al., 2013, Luz et al., 2018, Haider et al., 2019), linguistic information (Fraser et al., 2016, Mirheidari et al., 2018, Li et al., 2021b, Ye et al., 2021, Syed et al., 2020) and their combination (Haider et al., 2019, Pulido et al., 2020, Luz et al., 2020) and shows the early symptoms of AD are detectable from speech and language features.

It is known that cognitive impairments caused by dementia affects the speech production system (Ross et al., 1990). A growing body of research has demonstrated that quantifiable indicators of cognitive decline associated with AD are detectable in spontaneous speech (de la Fuente Garcia et al., 2020). Existing methods of automatic AD detection from spontaneous speech can roughly be divided into methods based on speech (Ivanov et al., 2013, Yu et al., 2015, Luz et al., 2018, Haider et al., 2019) and methods based on transcripts derived from speech (Mirheidari et al., 2018, Li et al., 2021b, Ye et al., 2021, Syed et al., 2020). Conventional audio-based methods exploit a variety of acoustic features such as pitch variance, syllable rate, phoneme-based measures, formant-based articulatory coordination features, log-Mel spectrogram, MFCC features, and other paralinguistic features (Yu et al., 2015, Ivanov et al., 2013, Mirheidari et al., 2018, Luz, 2017, Meghanani et al., 2021). The performance could be further improved by considering (dis)fluency features and speech pause distributions such as turn-taking patterns and speech rate (Luz et al., 2018, 2020, Yuan et al., 2020, Campbell et al., 2021, Pastoriza-Domínguez et al., 2022). The transcript-based

methods adopt various type of features derived from text content such as part-of-speech information, grammatical constituents, vocabulary richness (Bucks et al., 2000, Fraser et al., 2016), as well as word vector representations (Mirheidari et al., 2018, Guerrero-Cristancho et al., 2020). In addition to the use of hand-crafted features, pretrained language models based on deep neural networks, such as BERT and RoBERTa, have shown promising performance as feature extractors for AD detection in recent years (Balagopalan et al., 2020, Yuan et al., 2020, Syed et al., 2020, Rohanian et al., 2021b, Syed et al., 2021). Studies show that transcript-based methods tend to achieve better performance than audio-based methods (Luz et al., 2020, Balagopalan et al., 2020, Li et al., 2021b) and the combination of audio and text modalities usually leads to improved performance (Luz et al., 2020, Syed et al., 2020, Cummins et al., 2020, Balagopalan et al., 2020, Sarawgi et al., 2020). Model ensembles have also been studied to improve the robustness of the system where different classifiers are combined to produce the final decision (Cummins et al., 2020, Rohanian et al., 2021b, Qiao et al., 2021)

Although linguistic features are effective in detecting AD, the preparation for manual transcripts is time-consuming and costly. A fully automatic pipeline for AD detection replacing human transcribers by ASR systems is highly desirable. The major challenge is that the quality of speech transcription may degrade significantly when the speech is from language impaired patients. This requires a more effective way of integrating ASR models into the analysis and detection processes. Work on this include using commercial ASR systems (Rohanian et al., 2021a, Qiao et al., 2021), pretrained ASR models (Zhu et al., 2021), ASR systems adapted to spontaneous speech and elderly speech (Pan et al., 2021, Li et al., 2021b, Ye et al., 2021), and ASR models finetuned for AD (Qin et al., 2021).

8.2 Corpora for Depression and AD

This section describes the datasets used for detection of depression and AD in this chapter.

8.2.1 DAIC-WOZ

The distress analysis interview corpus - wizard-of-oz (DAIC-WOZ) database (DeVault et al., 2014) is part of a larger corpus, the distress analysis interview corpus (Gratch et al., 2014), that contains clinical interviews designed to support the diagnosis of psychological distress conditions such as anxiety, depression, and post-traumatic stress disorder. These interviews were collected as part of a larger effort to create a computer agent that interviews people and identifies verbal and non-verbal indicators of mental illness. Data collected include 189 audio

and video recordings and extensive questionnaire responses. This part of the corpus includes wizard-of-oz interviews, conducted by an animated virtual interviewer called Ellie, controlled by a human interviewer in another room. The data has been transcribed and annotated for a variety of verbal and non-verbal features. Among the 189 recordings, ~30% are labelled as depressed.

8.2.2 ADReSS

The ADReSS dataset (Luz et al., 2020), released with the Alzheimer's dementia recognition through spontaneous speech (ADReSS) challenge, consists of a statistically balanced, acoustically enhanced set of recordings of spontaneous speech sessions along with segmentation and detailed time-stamped transcriptions. It consists of speech recordings and transcripts of spoken picture descriptions elicited from participants through the "cookie theft" picture from the Boston diagnostic aphasia exam. The recorded speech has been segmented for voice activity using a simple voice activity detection algorithm based on signal energy threshold. The segmented dataset contains 1,955 speech segments from 78 non-AD subjects and 2,122 speech segments from 78 AD subjects. The average number of speech segments produced by each participant was 24.86. The recordings were acoustically enhanced with stationary noise removal and audio volume normalisation was applied across all speech segments to control for variations caused by recording conditions such as microphone placement.

8.3 Current Challenges in Automatic Detection of Depression and AD

8.3.1 Variability in Manifestations

Variability in manifestations causes difficulties in finding a unified representation for mental disorders. Typical sources of unwanted forms of variability (referred to as nuisance factors) includes (i) biological trait primitives such as race, ethnicity, gender, age; (ii) cultural trait primitives such as first language, dialect, and sociolect; (iii) emotional signals such as anger, fear, and energetic states; (iv) social signals such as conversational sounds, intimacy and dominance; (v) voice pathology such as speech disorders, intoxication and respiratory tract infection; (vi) co-existence of other forms of paralinguistic information. The between-class (healthy vs depressed) acoustic variability is diluted by linguistic information and speaker characteristics.

8.3.2 Data Scarcity

One major challenge in automatic mental illness detection is data scarcity. Medical corpora are usually limited in terms of both number of speakers and duration. Privacy concerns surrounding sensitive health information restricts access to large-scale, comprehensive datasets. Healthcare data are often protected by stringent regulations and ethical considerations, limiting data sharing and collaboration across institutions. Additionally, the annotation difficulty associated with labelling medical data, particularly for nuanced conditions like depression and AD, poses a significant challenge. Manual annotation by medical experts is time-consuming, expensive, and subject to inter-rater variability. Moreover, detection of mental illness mostly relies on interview data which is a sparse scenario. Compared to emotion recognition where data samples are labelled at segment-level (*i.e.,* one label per utterance), interview-based datasets are mainly labelled at session-level (*i.e.,* one label per interview). This means that given same amount the speech data, the effective number of samples are far less for depression and AD datasets than emotion datasets. Data scarcity affects the robustness and generalisability of automatic diagnosis systems, especially given variability in depression manifestation discussed above.

8.3.3 Data Imbalance

Apart from the limited size, medical datasets often also suffer from severe data imbalance where positive cases tend to be far fewer than negative cases (*i.e.*, healthy control group). Data augmentation and balancing approaches are current fields of interest, and F1 is commonly used instead of accuracy for evaluation which balances the trade-off between precision and recall.

8.3.4 Reliability and Confidence Estimation

Confidence estimation is crucial for a trustworthy automatic diagnostic systems which informs the clinician about the confidence of model predictions and helps reduce the risk of misdiagnosis. Deep learning models often suffer from calibration issues, leading to high confidence in incorrect predictions (*i.e.*, confidently wrong). While confidence estimation techniques have been applied in areas like speech recognition (Wessel et al., 2001, Jiang, 2005, Yu et al., 2011, Li et al., 2021c) and dialogue systems (Tur et al., 2005), their application in detecting mental illnesses through speech analysis remains largely unexplored.

8.4 Speech-Based Depression Detection using Foundation Models

Despite encouraging progress (Moore II et al., 2007, Low et al., 2010, Ooi et al., 2012, Williamson et al., 2016, Al Hanai et al., 2018) indicating that correlations of depression are detectable in spontaneous speech, speech-based depression detection (SDD) is still challenging due to the variability in depression manifestations and lack of training data.

Recently, foundation models have sparked a research paradigm shift in many fields of artificial intelligence (see Section 2.2.2). It has been shown that self-supervised learning (SSL) representations, the intermediate layer output of an SSL pretrained foundation model, are often useful for many downstream tasks (Yang et al., 2021). In particular, speech foundation models, such as Wav2vec 2.0 (W2V2), HuBERT, and WavLM, are attracting increasing attention and have achieved SOTA results in many speech processing tasks, including ASR and AER (Zhang et al., 2022b, Morais et al., 2022), *etc*. Despite this great success, SSL representations have not been extensively studied for SDD.

This section studies the use of SSL-pretrained speech foundation models to handle the challenges in SDD[1]. This approach allows the data sparsity issue to be handled via the large amount of unlabelled data used for SSL pretraining. Such unlabelled data can be produced by many speakers that cover a wide range of speaker variability and hence can help to model speaker-dependent depression manifestation variability. A block-wise analysis is first performed to compare the SSL representations from different layers of different foundation models and to understand which types of information is more effective in SDD. Next, foundation models are finetuned for the ASR and AER tasks separately, to investigate knowledge transfer from ASR and AER to SDD and the effect of finetuning on the intermediate layers. Three different speech foundation models, W2V2, HuBERT and WavLM, are compared. ASR transcriptions are encoded by RoBERTa, a text foundation model, and incorporated. The ensemble with multiple foundation models gives SOTA results on the benchmark DAIC-WOZ dataset.

The remainder of the section is organised as follows. Section 8.4.1 introduces the backbone structure. Section 8.4.2 describes the proposed data augmentation method. The experimental setup is presented in Section 8.4.3. Sections 8.4.4 and 8.4.5 present a block-wise analysis of speech foundation models and the use of ASR transcriptions in depression detection respectively. The foundation models are combined in Section 8.4.6, followed in Section 8.4.7 by a summary.

[1] Part of this section has been published as a conference paper (Wu et al., 2023c). See Appendix A for more detail.

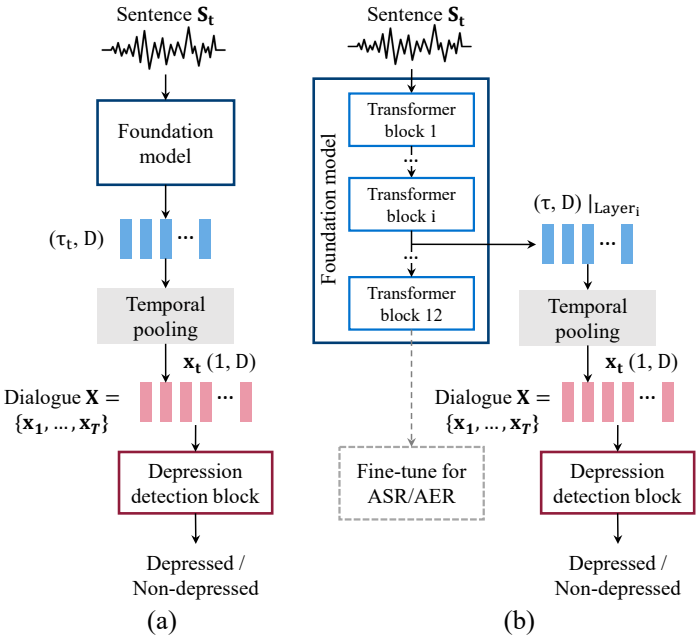


Fig. 8.1 (a) Model structure. (b) The block-wise analysis framework.

8.4.1 Model Structure

In this section, SDD is formulated as a binary classification task that determines whether the speaker is depressed or not. The SDD system takes a dialogue $\mathbf{X}$ (*i.e.*, a clinical interview) as input, which consists of a sequence of sentences $\mathbf{X} = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_T\}$ where T is the number of sentences in the dialogue. The model structure is illustrated in Fig. 8.1 (a) which contains a foundation model followed by a depression detection block. The foundation model takes a sentence $\mathbf{S}_t$ as input (*e.g.*, speech waveform or text) and produces a vector of size (τ_t, D) where τ_t is the number of frames in $\mathbf{S}_t$ and D is the feature dimension. Temporal pooling (mean pooling was used in this section) is then applied to the output of the foundation model, producing a D dimensional (-dim) vector $\boldsymbol{x}_t$ for each sentence. The depression detection block then takes a dialogue consisting of a T-length sequence with D-dim vectors as inputs to perform the diagnosis.

Three pretrained foundation models were used in this section: wav2vec 2.0[2] (W2V2), HuBERT[3], and WavLM[4] (see Section 2.2.4). The Base versions were used for all three foundation models which contain twelve 768-dim Transformer encoder blocks and about 95M parameters. The depression detection block consists of two 128-dim Transformer

[2]Available at: https://huggingface.co/facebook/wav2vec2-base
[3]Available at: https://huggingface.co/facebook/hubert-base-ls960
[4]Available at: https://huggingface.co/microsoft/wavlm-base-plus

encoder blocks with four attention heads each, followed by a FC output layer. The depression detection block has 0.3M parameters.

8.4.2 Sub-Dialogue Shuffling

As explained in Section 8.3.2, depression is usually assessed by clinical interview and labelled at the session-level, which is a very data sparse scenario. For instance, the DAIC-WOZ datasets consists of 50+ hours of speech recordings that correspond to merely 189 samples. Furthermore, data imbalance is another severe issue since the positive cases are much fewer than negative cases (28% vs 72% in training). Therefore, it is crucial to use data augmentation to alleviate both data scarcity and imbalance issues for SDD.

In this section, the training set is augmented using sub-dialogue shuffling, which samples a sub-dialogue $\boldsymbol{x}_{s:e}$ from each complete dialogue $\boldsymbol{x}_{1:T}$, where s and e are the randomly selected start and end utterance indexes. The details are given in Algorithm 3. First, the number of positive and negative samples in the training set are counted and M^+ is set which is the desired number of sub-dialogues for each positive dialogue (lines 1-3 of Algorithm 3). To augment while balancing the training samples, M^- is computed based on N^+, N^-, and M^+ (line 4). Then, M^+ and M^- sub-dialogues are generated for each complete dialogue belonging to the positive and negative classes respectively (lines 8-10 of Algorithm 3). ϵ_l and

Algorithm 3 Sub-dialogue shuffling

1: $N^+ \leftarrow$ Number of positive samples in the training set
2: $N^- \leftarrow$ Number of negative samples in the training set
3: Set number of sub-dialogues for each positive sample M^+
4: $M^- \leftarrow N^+ \times M^+ / N^-$
5: Set ϵ_l, ϵ_h satisfying $0 < \epsilon_l < \epsilon_h <= 1$
6: **for** Dialogue $\mathbf{X}^{(n)}, n = 1, 2, \ldots, N$ **do**
7: $\quad T \leftarrow \text{len}(\mathbf{X}^{(n)})$
8: $\quad$ **if** $\mathbf{X}^{(n)}$ is positive **then** $M \leftarrow M^+$
9: $\quad$ **else** $M \leftarrow M^-$
10: $\quad$ **end if**
11: $\quad$ **for** Sub-dialogue $\mathbf{X}^{(n)_m}, m = 1, 2, \ldots, M$ **do**
12: $\quad\quad$ Sample ϵ uniformly from $[\epsilon_l, \epsilon_h)$
13: $\quad\quad d \leftarrow \epsilon T - 1$
14: $\quad\quad$ Sample s randomly from range $[0, T - d)$
15: $\quad\quad e \leftarrow s + d$
16: $\quad\quad \mathbf{X}^{(n)_m} \leftarrow \mathbf{X}^{(n)}_{s:e}$
17: $\quad$ **end for**
18: **end for**

ϵ_h are two variables that determine the length range of the sub-dialogues. When generating a sub-dialogue, its length d is first defined by a coefficient randomly drawn from $[\epsilon_l, \epsilon_h)$ (lines 12-13). The start index s is then randomly chosen from its available range and the end index is then determined (lines 14-16).

8.4.3 Experimental Setup

The DAIC-WOZ dataset (see Section 8.2.1) is used in this section. For a fair comparison to prior work (Gong and Poellabauer, 2017, Al Hanai et al., 2018, Ravi et al., 2022, Wu et al., 2022c), results on the development subset are reported. 30 out of 107 interviews within the training set and 12 out of 35 interviews within the development set are labelled as depressed. Classification performance is evaluated by the F1 score, which is the harmonic mean of the precision and recall. Precision computes the proportion of positive identifications being actually correct, which is defined as follows:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{8.1}$$

where TP (true positives) is the number of samples correctly predicted as "positive", and FP (false positives) is the number of samples wrongly predicted as "positive". Recall computes the proportion of actual positives being identified correctly, which is defined as follows:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{8.2}$$

where FN (false negatives) is the number of samples wrongly predicted as "negative". F1 score is then defines as:

$$\text{F1} = \frac{2}{\frac{1}{\text{Precision}} + \frac{1}{\text{Recall}}} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{8.3}$$

which measures the trade-off between precision and recall and is thus suitable for evaluate the classification performance when the class distribution is imbalanced. The model was initialised and trained for 20 different random seeds and both the highest (F1(max)) and the average (F1(avg)) value are reported, along with the standard deviation (F1(std)) across seeds.

8.4.4 Experiment: Block-Wise Analysis of Foundation Models

It has been previously found that the output of different encoder blocks of a speech foundation model contains different levels of information (Pasad et al., 2021, Zheng et al., 2022). The block-wise evolution of the representations follows an acoustic-linguistic hierarchy, where the shallowest layers encode acoustic features, followed by the word meaning information, and phonetic and word identities. The analysis of the intermediate block representations could provide insights to better understand the information relevant to SDD. This section performs such an analysis. The model structure used for block-wise analysis is shown in Fig. 8.1 (b). Each time output from one intermediate Transformer block from the foundation model is used for downstream SDD.

Effect of Data Augmentation

The effect of data augmentation was first investigated using the output of the last (12^{th}) Transformer block of the pretrained WavLM model ($\text{WavLM}^{\text{PT}}_{12}$). Augmenting data trades off between generating more data and matching the true data distribution. As shown in Table 8.1, the F1 score increases and standard deviation decreases as the number of sub-dialogues for each positive sample M^+ increases up until 1000, then F1 decreases and the standard deviation increases. The model runs the risk of overfitting the training data if each original sequence is replicated too many times. The following experiments used $M^+ = 500$, balancing performance and training time.

Table 8.1 SDD results with increased number of augmented utterances on DAIC-WOZ. $\text{WavLM}^{\text{PT}}_{\text{L12}}$ used as input. M^+ is the number of sub-dialogues for each positive sample.

M^+	100	200	500	1000	1500
F1(avg)	0.451	0.583	0.647	0.679	0.669
F1(max)	0.640	0.700	0.714	0.762	0.727
F1(std)	0.131	0.082	0.033	0.027	0.031

Pretrained SSL Representations

The parameters of the three pretrained foundation models (W2V2^{PT}, $\text{HuBERT}^{\text{PT}}$, WavLM^{PT}) were frozen and the SDD results using different intermediate blocks of the models are shown in Table 8.2. F1(avg) of the intermediate blocks of three models are plotted in Fig. 8.2. For all three models, F1 first improves as the layer number increases and then F1 decreases. Overall, WavLM^{PT} produces an F1 score higher than the other two models. The features

Table 8.2 SDD results using the outputs from different intermediate blocks of different pretrained foundation models on DAIC-WOZ. "id" indicates index of intermediate blocks. Highest F1 value in each column shown in bold.

W2V2^{PT}				HuBERT^{PT}				WavLM^{PT}			
id	**F1(avg)**	**F1(max)**	**F1(std)**	**id**	**F1(avg)**	**F1(max)**	**F1(std)**	**id**	**F1(avg)**	**F1(max)**	**F1(std)**
2	0.531	0.615	0.044	2	0.557	0.615	0.033	2	0.545	0.636	0.033
4	0.549	0.667	0.055	4	0.582	0.621	0.020	4	0.571	0.629	0.029
6	0.597	**0.700**	0.056	6	0.606	0.667	0.046	6	0.630	0.692	0.034
8	**0.627**	0.667	0.043	8	0.628	0.714	0.049	8	**0.700**	**0.750**	0.024
10	0.536	0.667	0.060	10	**0.667**	**0.762**	0.052	10	0.685	0.720	0.031
12	0.519	0.636	0.066	12	0.610	0.696	0.034	12	0.647	0.714	0.033

extracted from the 10th -block give the highest F1 for HuBERT^{PT} while the features extracted from the 8th-block have the overall best performance for W2V2^{PT} and WavLM^{PT}. It has been found (Pasad et al., 2021) that the first few W2V2 Transformer blocks show increased similarity with Mel filterbank energy (MFB) features, indicating that shallow layers encode acoustic information much like MFB. Word meaning information is mainly encoded in middle blocks, especially around the 8th-block (Pasad et al., 2021). Hence it can be inferred that features contain word meaning information are useful for SDD.

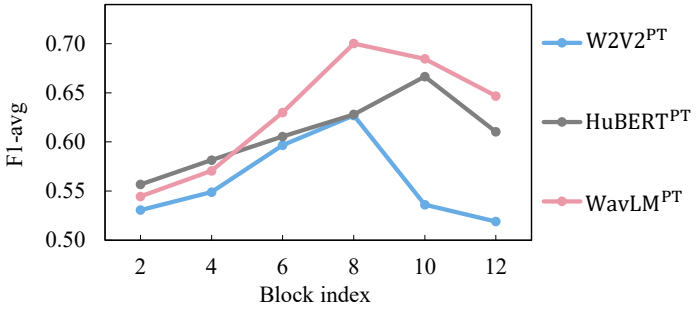


Fig. 8.2 Trends of DAIC-WOZ F1(avg) values at different blocks for the pretrained foundation models.

ASR and AER Finetuned Representations

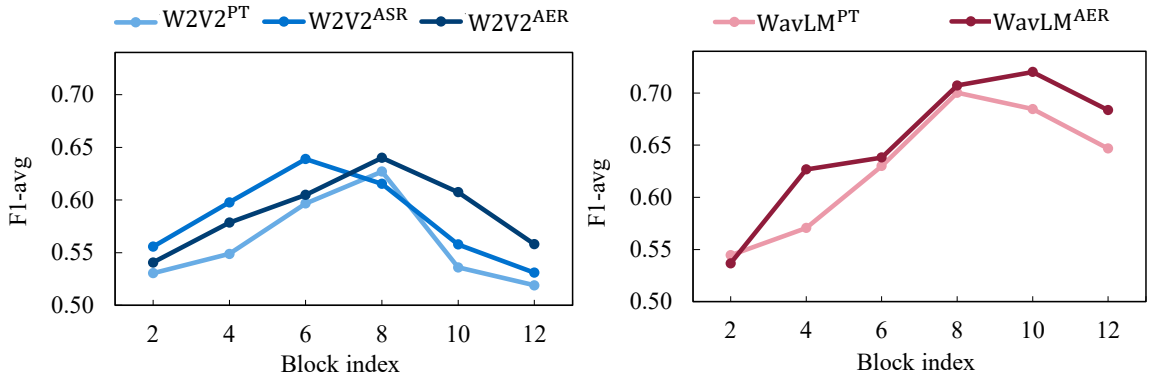
This section investigates how finetuning changes the findings in the previous section. It has been implied in the previous section that the intermediate layer containing information correlated with word meaning is effective to SDD. It has also been found (Wu et al., 2022c) that emotion information is also useful to SDD. Thus, our foundation models are finetuned based on data for ASR and AER tasks. Three finetuned systems are investigated in this paper with parameters frozen after finetuning:

Table 8.3 SDD results using the outputs from different intermediate blocks of different foundation models finetuned for ASR and AER on DAIC-WOZ. Highest F1 value in each column shown in bold.

W2V2^{ASR}				W2V2^{AER}				WavLM^{AER}			
id	**F1(avg)**	**F1(max)**	**F1(std)**	**id**	**F1(avg)**	**F1(max)**	**F1(std)**	**id**	**F1(avg)**	**F1(max)**	**F1(std)**
2	0.556	0.696	0.051	2	0.541	0.615	0.050	2	0.537	0.600	0.022
4	0.598	**0.700**	0.052	4	0.579	0.643	0.043	4	0.627	0.690	0.027
6	**0.639**	0.690	0.045	6	0.605	**0.737**	0.041	6	0.638	0.667	0.027
8	0.615	0.649	0.025	8	**0.640**	0.688	0.036	8	0.707	**0.786**	0.032
10	0.558	0.645	0.040	10	0.608	0.696	0.058	10	**0.720**	0.769	0.036
12	0.531	0.615	0.054	12	0.558	0.667	0.045	12	0.684	0.750	0.032

- W2V2^{ASR}: W2V2 base model finetuned for ASR on the 960 hours of LibriSpeech data[5].
- W2V2^{AER}: W2V2 base model finetuned on 110 hours of MSP-Podcast dataset for AER by adding two extra FC layers[6].
- WavLM^{AER}: WavLM base model finetuned in the same way as W2V2^{AER} for AER[7].

The SDD results of the finetuned models are shown in Table 8.3. Comparing the results of W2V2^{ASR} and W2V2^{AER} with W2V2^{PT} , as shown in Fig. 8.3 (a), the peak of W2V2^{ASR} is further towards earlier blocks while the peak of W2V2^{AER} is towards later blocks. As



(a) W2V2 finetuned on ASR and AER. (b) WavLM finetuned on AER.

Fig. 8.3 Trends of DAIC-WOZ F1(avg) values at different blocks for the foundation models finetuned for ASR and AER.

[5]Available at: https://huggingface.co/facebook/wav2vec2-base-960h

[6]The concordance correlation coefficients (see Section 6.2.4) for valence, activation, and dominance are 0.418, 0.658, 0.562 for W2V2^{AER}.

[7]The concordance correlation coefficients (see Section 6.2.4) for valence, activation, and dominance are 0.445, 0.667, 0.597 for WavLM^{AER}.

shown in Fig. 8.3 (b), the performance of WavLM$^{\text{AER}}$ also improves over WavLM$^{\text{PT}}$ on later layers. The finetuned foundation models presumably learn more task-specific information. For a W2V2 model finetuned with character-level CTC loss (see Section 4.1.3), the output of the last few layers are more directly related to the word identities. Finetuning the foundation model for AER improves the overall performance, indicating that emotion and depression share some paralinguistic indicators encoded by the finetuned models.

8.4.5 Experiment: The Use of ASR Transcriptions

It has been shown that text information is effective for SDD (Williamson et al., 2016, Dinkel et al., 2019). However, reference transcriptions are usually not available in practice. This section uses an ASR system to transcribe the depression detection interview and investigates the performance of using erroneous transcriptions in SDD. Automatic transcriptions were obtained from the final output of the W2V2$^{\text{ASR}}$ model which has a WER of 3.4% on LibriSpeech "test-clean" set and 8.6% on "test-other" set but 40.9% on DAIC-WOZ. The ASR and reference transcripts were encoded by a text foundation model, the RoBERTa Base model[8] (see Section 2.2.3) and fed into the depression detection block. The SDD results with ASR generated hypotheses and reference transcriptions are compared in Table 8.4 (RoBERTa$^{\text{Hyp}}$, RoBERTa$^{\text{Ref}}$). Replacing the reference transcriptions with ASR generated hypotheses leads to a decrease of 0.36 in average F1 score and also a larger standard deviation.

Utterance-level representations derived from RoBERTa$^{\text{Hyp}}$ were combined with those derived from the 6$^{\text{th}}$-block representations of the ASR-finetuned W2V2 model (W2V2$_6^{\text{ASR}}$) by concatenation. From Table 8.4, this combination produced better SDD results than using the reference transcriptions alone.

Table 8.4 Comparison of using reference and ASR transcriptions for SDD on DAIC-WOZ, where Cat$\{\cdot,\cdot\}$ refers to a concatenation.

System	F1(avg)	F1(max)	F1(std)
RoBERTa$^{\text{Hyp}}$	0.599	0.667	0.042
RoBERTa$^{\text{Ref}}$	0.635	0.667	0.029
Cat{RoBERTa$^{\text{Hyp}}$,W2V2$_6^{\text{ASR}}$}	**0.648**	**0.714**	0.028

[8] Available at: https://huggingface.co/roberta-base

Table 8.5 Results of combining different speech and text foundation models on DAIC-WOZ, where Cat{·, ·} refers to a concatenation.

System	F1(avg)	F1(max)	F1(std)
$\text{RoBERTa}^{\text{Hyp}}$	0.599	0.667	0.042
$\text{WavLM}_{8}^{\text{PT}}$	0.700	0.750	0.024
$\text{WavLM}_{10}^{\text{AER}}$	0.720	0.769	0.036
$\text{Cat}\{\text{WavLM}_{8}^{\text{PT}}, \text{RoBERTa}^{\text{Hyp}}\}$	0.725	0.759	0.021
$\text{Cat}\{\text{WavLM}_{10}^{\text{AER}}, \text{RoBERTa}^{\text{Hyp}}\}$	0.756	0.800	0.023

Table 8.6 Ensemble of foundation models. Cat{·, ·} refers to a concatenation.

System	Ensemble 1	Ensemble 2
$\text{W2V2}_{6}^{\text{PT}}$	√	
$\text{HuBERT}_{10}^{\text{PT}}$	√	
$\text{WavLM}_{8}^{\text{PT}}$	√	√
$\text{WavLM}_{10}^{\text{AER}}$		√
$\text{Cat}\{\text{WavLM}_{10}^{\text{AER}}, \text{RoBERTa}^{\text{Hyp}}\}$		√
F1(avg)	0.800	0.829
F1(max)	0.857	0.886

8.4.6 Experiment: Combinations of Foundation Models

This section studies further combinations of SSL representations derived from both speech and text foundation models. Similar to the experiments in Table 8.4, speech SSL representations were combined with the ASR transcriptions by a concatenation, and the results are shown in Table 8.5. Combining speech and ASR-hypothesis-based text representations can improve F1(avg) and F1(max) as well as reduce F1(std), which improves both SDD classification performance and stability.

Finally, the use of a system ensemble by voting is investigated. Two ensembles are tested:

1. The ensemble of systems based on three speech foundation models: $\text{W2V2}_{6}^{\text{PT}}$, $\text{HuBERT}_{10}^{\text{PT}}$, $\text{WavLM}_{8}^{\text{PT}}$
2. The ensemble of systems from three modalities: $\text{WavLM}_{8}^{\text{PT}}$ (audio modality), $\text{WavLM}_{10}^{\text{AER}}$ (emotion modality), $\text{Cat}\{\text{WavLM}_{10}^{\text{AER}}, \text{RoBERTa}^{\text{Hyp}}\}$ (text modality)

The results of using ensembles are shown in Table 8.6. Reference transcriptions are not used in these ensembles and our best-performing depression detection systems require only the speech input. Table 8.7 cross compares our results with those published in literature.

Table 8.7 Cross comparison on DAIC-WOZ development subset.

Paper	F1(avg)	F1(max)
Gong and Poellabauer (2017)	-	0.70
Al Hanai et al. (2018)	-	0.77
Shen et al. (2022)	-	0.85
Ravi et al. (2022)	0.69	-
Wu et al. (2022c)	-	0.87
Proposed	0.83	**0.89**

Ravi et al. (2022) used W2V2 and reported the average result across five models. Reference transcriptions were used by Gong and Poellabauer (2017), Al Hanai et al. (2018), Shen et al. (2022), Wu et al. (2022c). The comparison shows that the ensemble of foundation models produced competitive performance for depression detection based on speech input only.

8.4.7 Summary

This section studies the use of SSL representations in speech-based depression detection. An analysis of SSL representations derived from different layers of pretrained foundation models is first presented for SDD, which provides insight to suitable indicator for depression detection. Knowledge transfer is then performed from ASR and AER to SDD by finetuning the foundation models. Results show that finetuning pretrained speech foundation models for AER improves SDD performance, indicating that some indicators are shared between AER and SDD. SDD performance when using ASR transcriptions matches that of using reference transcriptions when combined with the hidden representations derived from an ASR-finetuned foundation model. By integrating representations from multiple foundation models, SOTA SDD results are achieved on the DAIC-WOZ dataset without using the reference transcriptions. A similar approach has been applied to AD detection (Cui et al., 2023) where W. W. is the co-first author. Details can be found in Appendix A.2.

8.5 Confidence Estimation for Detection of AD and Depression

Estimating confidence levels is key in medical tasks. While supportive in diagnosis, deep learning models often suffer from calibration issues, leading to high confidence in incorrect predictions (*i.e.*, confidently wrong). Confidence estimation can increase the reliability and interpretability of diagnostics powered by deep learning, offering clinicians a clearer

understanding of how much trust to place in automated predictions to reduce the risk of misdiagnosis. It can also facilitate the identification of ambiguous and borderline cases, necessitating the input of clinical expertise. While confidence estimation techniques have been applied in areas like speech recognition (Wessel et al., 2001, Jiang, 2005, Yu et al., 2011, Li et al., 2021c) and dialogue systems (Tur et al., 2005), their application in detecting mental illnesses through speech analysis remains largely unexplored.

This section investigates confidence estimation for automatic AD and depression detection based on speech recordings from clinical interviews[9]. Standard deep neural network classifiers are often trained to maximise the categorical probability of the correct class using the cross-entropy loss, whose output logit values are converted into pseudo-categorical probability distributions as the predictions using a softmax function. This standard framework is known to have unreliable uncertainty estimation (Gal and Ghahramani, 2016, Guo et al., 2017). In this section, the EDL-based uncertainty estimation approach developed in Chapter 5 is adapted for confidence estimation which introduces a dynamic Dirichlet prior distribution to model the second-order probability over the predictive distribution. The dynamic Dirichlet prior is predicted by a deep neural network trained by minimising the Bayes risk of the prediction. Multiple evaluation metrics are adopted to evaluate the proposed method in terms of classification performance and confidence estimation. Results on the ADReSS and DAIC-WOZ datasets show that the proposed method outperforms the baselines in terms of both classification accuracy and model calibration. To the best of our knowledge, this is the first work that investigates confidence estimation for automatic AD and depression detection based on clinical interviews.

The rest of the section is organised as follows. Section 8.5.1 introduces the proposed approach of confidence estimation. The experimental setup and evaluation metrics are presented in Sections 8.5.2 and 8.5.3 respectively. The experimental results are given in Section 8.5.4, followed by the summary in Section 8.5.7.

8.5.1 Confidence Estimation Method

Confidence is defined as the probability corresponding to the predicted class in the predictive distribution[10]. A standard neural network classifier predicts a categorical distribution which represents the probabilistic assignment over the possible classes. In this section, the EDL approach proposed in Chapter 5 is modified and adapted for confidence estimation which

[9]Part of this section has been published as a conference paper (Wu et al., 2024c). See Appendix A for more detail.

[10]In this section, confidence is treated as a posterior probability. In some applications only relative confidence is required since a threshold will be fixed.

places a dynamic Dirichlet prior over the categorical distribution to measure the probability of the predictive distribution (*i.e.,* second-order probability) instead of a point estimate.

Modelling Second-order Probability

The problem setup follows that in Chapter 5, which is summarised below. Consider the target label as a one-hot vector $\boldsymbol{y}$ where $\boldsymbol{y}_k$ is one if class k is the correct class else zero. $\boldsymbol{y}$ is sampled from a categorical distribution $\boldsymbol{\pi}$ where each component $\boldsymbol{\pi}_k$ corresponds to the probability of sampling a label from class k. A Dirichlet prior is introduced to model the distribution over the categorical distribution:

$$\boldsymbol{y} \sim \mathrm{P}(\boldsymbol{y}|\boldsymbol{\pi}) = \mathrm{Cat}(\boldsymbol{y}|\boldsymbol{\pi}), \quad \boldsymbol{\pi} \sim \mathrm{p}(\boldsymbol{\pi}|\boldsymbol{\alpha}) = \mathrm{Dir}(\boldsymbol{\pi}|\boldsymbol{\alpha}). \tag{8.4}$$

where $\boldsymbol{\alpha}$ is the hyperparameter of the Dirichlet distribution. The output of a standard neural network classifier is a probability assignment over the possible classes. The Dirichlet distribution represents the density of each such probability assignment, hence it is modelling second-order probabilities and uncertainty. For a given input $\boldsymbol{x}_i$, the hyperparameter $\boldsymbol{\alpha}_i$ is predicted by a neural network $\boldsymbol{\alpha}_i = \boldsymbol{f}_{\boldsymbol{\Lambda}}(\boldsymbol{x}_i)$ where $\boldsymbol{\Lambda}$ is the model parameter vector.

Learning a Dynamic Dirichlet Prior

For brevity, superscript i is omitted in this section. Given a one-hot label $\boldsymbol{y}$ and predicted Dirichlet $\mathrm{Dir}(\boldsymbol{\pi}|\boldsymbol{\alpha})$, a neural network can be trained by minimising the Bayes risk with respect to the sum of squares loss between targets and the class predictor:

$$\begin{aligned}
\mathcal{L}^{\mathrm{BR}}(\boldsymbol{\Lambda}) &= \int \| \boldsymbol{y} - \boldsymbol{\pi} \|^2 \, \mathrm{p}(\boldsymbol{\pi}|\boldsymbol{\alpha}) \mathrm{d}\boldsymbol{\pi} \\
&= \sum_{k=1}^{K} \mathbb{E}\left[(\boldsymbol{y}_k)^2 - 2\boldsymbol{y}_k \boldsymbol{\pi}_k + (\boldsymbol{\pi}_k)^2 \right] \\
&= \sum_{k=1}^{K} \left(\boldsymbol{y}_k - \frac{\boldsymbol{\alpha}_k}{\boldsymbol{\alpha}_0} \right)^2 + \frac{\boldsymbol{\alpha}_k \left(\boldsymbol{\alpha}_0 - \boldsymbol{\alpha}_k \right)}{(\boldsymbol{\alpha}_0)^2 \left(\boldsymbol{\alpha}_0 + 1 \right)}.
\end{aligned} \tag{8.5}$$

Similar to Section 5.4, the KL divergence between the targets and prediction is introduced as an additional regularisation term. The total loss is then defined as:

$$\mathcal{L}(\boldsymbol{\Lambda}) = \mathcal{L}^{\mathrm{BR}}(\boldsymbol{\Lambda}) + \lambda \cdot \mathcal{KL}\left[\boldsymbol{y} \parallel \boldsymbol{\pi} \right] \tag{8.6}$$

with the coefficient λ set to 0.5.

Predictive Distribution and Confidence Estimation

Given $\mathrm{Dir}(\boldsymbol{\pi}|\boldsymbol{\alpha})$, the estimated probability of class k can be calculated by the expectation of the predicted Dirichlet prior. Since the Dirichlet distribution is the conjugate prior of the categorical distribution, the expectation is tractable:

$$\hat{\boldsymbol{\pi}}_k = \mathbb{E}[\boldsymbol{\pi}_k] = \frac{\boldsymbol{\alpha}_k}{\boldsymbol{\alpha}_0} \tag{8.7}$$

where $\boldsymbol{\alpha}_0 = \sum_{k=1}^{K} \boldsymbol{\alpha}_k$. Confidence is then computed as follows:

$$\hat{k} = \mathrm{argmax}_k \, \hat{\boldsymbol{\pi}} \tag{8.8}$$

$$p = \hat{\boldsymbol{\pi}}_{\hat{k}} \tag{8.9}$$

where $\hat{\boldsymbol{\pi}}$ is the predictive distribution, $\hat{k}$ is the predicted class, and p is the prediction confidence for a given input $\boldsymbol{x}$. The confidence is expected to reflect the probability of the output being actually correct.

8.5.2 Experimental Setup

Datasets

The ADReSS dataset (see Section 8.2.2) is used in this section for automatic AD detection. The standard split of train/test data provided by the corpus is used. 20% of the training data was further set aside for validation. The DAIC-WOZ data (see Section 8.2.1) is used for automatic depression detection. The standard split of train/dev/test data provided by the corpus is used.

Model Structure

The model structure is shown in Fig. 8.4. Following Cui et al. (2023), the recording was first transcribed using a pretrained Whisper model[11] (Radford et al., 2023), which has a word error rate of 32.8% on ADReSS and 20.4% on DAIC-WOZ. The transcription was then encoded by a pretrained BERT model[12] (Devlin et al., 2019). The model consists of two transformer encoder blocks of dimension 128 with four attention heads, followed by two FC layers and an output layer.

[11] Available at: https://huggingface.co/openai/whisper-small
[12] Available at: https://huggingface.co/bert-base-uncased

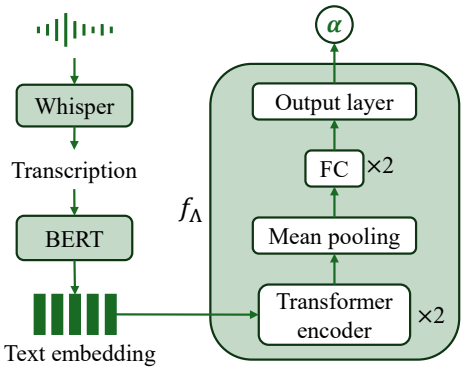


Fig. 8.4 Illustration of the model structure.

Baselines

The proposed method is compared with following baselines:

- L2: a standard classification network with softmax output activation trained by cross-entropy loss with weight decay.
- MCDP: a Monte Carlo dropout (Gal and Ghahramani, 2016) model with dropout rate of 0.3 which was forwarded 50 times during testing with different dropout random seeds to obtain 50 samples.
- BBB: a Bayes-by-backprop (Blundell et al., 2015) model which was forwarded 50 times during testing with different network weights to obtain 50 samples.
- Ensemble: a ensemble of five L2 models initialised and trained using different random seeds.

All of the baselines have the same backbone structure as the proposed method. The confidence is computed by Eqn. (8.9).

Implementation Details

Sub-dialogue shuffling proposed in Section 8.4.2 was applied to augment and balance the training set which samples sub-dialogues $\boldsymbol{x}_{s:e}$ from each complete dialogue $\boldsymbol{x}_{1:T}$, where s and e are the randomly selected start and end sentence indices. The number of sub-dialogues for positive samples was set to 100 for AD and 500 for depression. Piece-wise linear mappings (PWLMs) (Evermann and Woodland, 2000) were estimated on the validation set and then applied to the test set in order to better calibrate the confidence scores with accuracy. All experiments were run for 5 different seeds and the mean and standard error are reported.

8.5.3 Evaluation Metrics

A range of metrics are adopted to evaluate the proposed method in terms of classification performance and model calibration.

Classification Performance

The classification performance is evaluated by the accuracy (ACC) and F1 score (F1).

Model Calibration

Model calibration is evaluated by the expected calibration error (ECE), the normalised cross entropy (NCE), the area under the ROC curve (AUROC), and the area under the precision-recall curve (AUPRC). ECE, AUROC and AUPRC have been introduced in Section 5.3.3. The predicted confidence is used as the decision threshold for both AUROC and AUPRC.

Normalised cross entropy (NCE) (Siu et al., 1997) measures the quality of confidence scores. Confidence scores for all test samples $p = [p_1, \ldots, p_N]$ where $p_n \in [0, 1]$ are gathered and their corresponding target confidence $c = [c_1, \ldots, c_N]$ where $c_n \in \{0, 1\}$. The NCE is then given by

$$\mathrm{NCE}(\boldsymbol{c}, \boldsymbol{p}) = \frac{\mathcal{H}(\boldsymbol{c}) - \mathcal{H}(\boldsymbol{c}, \boldsymbol{p})}{\mathcal{H}(\boldsymbol{c})} \tag{8.10}$$

where $\mathcal{H}(\boldsymbol{c})$ is the entropy of the target confidence sequence and $-\mathcal{H}(\boldsymbol{c}, \boldsymbol{p})$ is the binary cross-entropy between the target and the estimated confidence scores. When confidence estimation is systematically better than the correct ratio ($\sum_{n=1}^{N} c_n/N$), NCE is positive. For perfect confidence scores, NCE is 1.

8.5.4 Experiments: Classification Performance

In this section, the proposed method is compared to the baselines described in Section 8.5.2 in terms of classification accuracy and confidence estimation.

Table 8.8 lists the classification accuracy and F1 scores for all of the compared methods. Comparing the proposed method to the baselines, it is shown that introducing a confidence measure does not degrade classification performance. The proposed method yields the best F1 and accuracy for AD detection as well as the highest accuracy for depression detection. Although the ensemble achieves the best prediction F1 score for depression detection, it involves training five individual systems. The proposed method achieves the second best accuracy with only a fifth of the computational cost of Ensemble during training. During testing, the Ensemble involves five individual forward passes of each base model. Both

Table 8.8 Comparison to the baselines in terms of classification accuracy and F1 score. Average and standard error of five runs are reported. The best value in each column is shown in bold and the second best is underlined.

	AD detection		**Depression detection**	
	F1	**ACC**	**F1**	**ACC**
L2	0.791±0.010	0.783±0.015	0.585±0.014	0.706±0.018
MCDP	0.786±0.010	0.779±0.014	0.572±0.006	0.695±0.034
BBB	0.738±0.008	0.721±0.025	0.580±0.012	0.715±0.024
Ensemble	<u>0.792±0.011</u>	<u>0.788±0.013</u>	**0.602±0.008**	<u>0.738±0.004</u>
Proposed	**0.807±0.013**	**0.800±0.019**	<u>0.600±0.008</u>	**0.745±0.008**

Table 8.9 Comparison to the baselines in terms of ECE. A smaller ECE value indicates better model calibration. Average and standard error of five runs are reported. The best value in each column is shown in bold and the second best is underlined.

	AD		**Depression**	
	ECE (w/o PWLM)	**ECE(w PWLM)**	**ECE (w/o PWLM)**	**ECE (w PWLM)**
L2	0.227±0.016	0.204±0.014	0.312±0.023	0.217±0.008
MCDP	0.229±0.022	0.207±0.005	0.302±0.026	0.252±0.009
BBB	0.216±0.025	0.195±0.013	<u>0.287±0.020</u>	<u>0.208±0.012</u>
Ensemble	<u>0.173±0.012</u>	<u>0.153±0.017</u>	0.370±0.007	0.219±0.008
Proposed	**0.163±0.011**	**0.137±0.004**	**0.207±0.009**	**0.183±0.009**

MCDP and BBB involves 50 forward passes to obtain 50 samples. In contrast, the proposed method only requires a single forward pass and is thus the most efficient during testing.

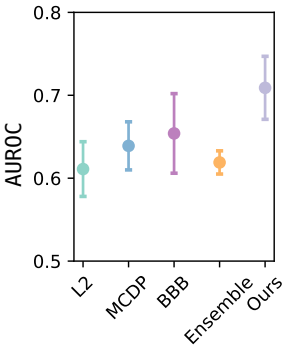
8.5.5 Experiments: Confidence Estimation

The ECE values and NCE values before and after applying a PWLM are shown in Table 8.9 and Table 8.10 respectively. It can be seen that applying a PWLM improves both ECE and NCE for most methods while it has larger impact on NCE than ECE. The proposed method performs the best before applying the PWLM and remains the best after applying the PWLM.

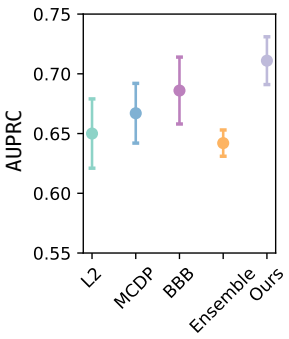
Since a PWLM is monotonic, NCE and ECE values will be affected while AUROC and AUPRC remain unchanged as the relative order of confidence scores is unchanged. The AUROC and AUPRC are compared in Fig. 8.5 and Fig. 8.6 for detection of AD and depression respectively. The proposed method performs the best in terms of both AUROC and AUPRC on both datasets, which further demonstrates its superior capability of confidence estimation.

Table 8.10 Comparison to the baselines in terms of NCE. A larger NCE value indicates better confidence estimation. Average and standard error of five runs are reported. The best value in each column is shown in bold and the second best is underlined.

	AD		Depression	
	NCE (w/o PWLM)	**NCE (w PWLM)**	**NCE (w/o PWLM)**	**NCE (w PWLM)**
L2	-0.138±0.047	0.103±0.011	-0.288±0.079	0.046±0.012
MCDP	-0.083±0.032	0.121±0.010	<u>-0.116±0.052</u>	<u>0.100±0.011</u>
BBB	-0.140±0.63	0.095±0.028	-0.171±0.077	0.073±0.017
Ensemble	<u>0.028±0.017</u>	<u>0.141±0.035</u>	-0.307±0.079	0.045±0.010
Proposed	**0.066±0.038**	**0.210±0.030**	**0.048±0.028**	**0.155±0.007**

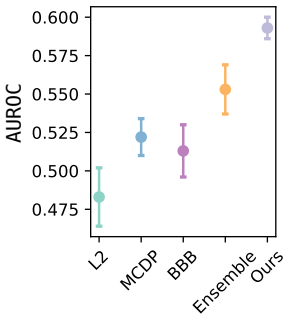


(a) AUROC

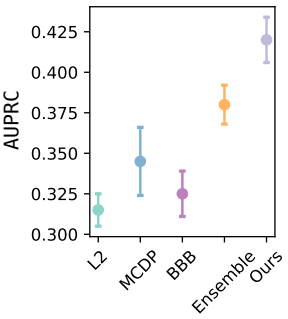


(b) AUPRC

Fig. 8.5 Comparison to the baselines in terms of AUROC and AUPRC for AD detection. Average of five runs are plotted along with standard error as error bars.



(a) AUROC



(b) AUPRC

Fig. 8.6 Comparison to the baselines in terms of AUROC and AUPRC for depression detection. Average of five runs are plotted along with standard error as error bars.

8.5.6 Analysis

A reject option was applied to analyse the confidence estimation performance, where the diagnosis system has the option to accept or decline a test sample based on the confidence prediction. As shown in Table 8.11, improved classification performance is observed for both AD and depression detection when a confidence threshold is increased from 0.5 to 0.8. This shows that the model provides more accurate predictions when it is more confident, which indicates the effectiveness of the proposed confidence estimation.

Table 8.11 A reject option based on confidence score.

Confidence threshold	AD		Depression	
	F1	ACC	F1	ACC
50%	0.807	0.800	0.600	0.745
80%	0.913	0.920	0.755	0.831

8.5.7 Summary

This section investigates confidence estimation of automatic detection of Alzheimer's disease and depression. EDL is adopted which places a dynamic Dirichlet prior over the categorical likelihood to model the second-order probability of model prediction. A range of metrics have been used to evaluate the performance for detection accuracy and confidence estimation. The results show that the proposed method clearly and consistently outperforms a range of baselines in terms of both classification accuracy and confidence estimation for both Alzheimer's disease detection and depression detection. It is hoped that the proposed method could help promote reliable and trustworthy automatic diagnostic systems. Although this work focuses on detection based on speech information from clinical interviews, the proposed method should also be able to be applied to other input modalities (*e.g.*, images) and the confidence estimation could also be useful when combining different systems.

8.6 Chapter Summary

Mental wellbeing has attracted increasing attention. This chapter studies automatic detection of Alzheimer's disease and depression via spontaneous speech. In order to tackle the data sparsity challenge of automatic diagnosis systems, foundation models pretrained on a large amount of speech data are used to transfer knowledge for speech-based automatic depression detection. By integrating representations from multiple foundation models, SOTA results have been achieved without the need for oracle transcriptions. In order to tackle the reliability challenge of automatic diagnosis systems, confidence estimation methods have been investigated for automatic detection of AD and depression. Evidential deep learning has been adapted which uses a dynamic Dirichlet prior to model the second-order probability of the predictive distribution. The results demonstrate that the proposed method outperforms a range of baselines for both classification accuracy and confidence estimation. However, it is important to acknowledge that confidence estimations in automatic diagnosis systems can hardly guarantee perfect accuracy. Therefore, it remains essential to include warnings to

manage user expectations and prevent over-reliance. Despite this, it is still hoped that this method could help promote reliable and trustworthy automatic diagnostic systems.

Chapter 9

Conclusions and Future Work

This thesis focuses on emotion understanding and automatic detection of mental illness and cognitive diseases from speech. The thesis begins with an introduction to deep learning in Chapter 2, with commonly used building blocks explained and training algorithms discussed, followed by the introduction to automatic emotion recognition (AER) in Chapter 3. Then, a novel integrated system is introduced in Chapter 4 which integrates AER with speaker diarisation and speech recognition in a jointly-trained system. Chapter 5 explores various ways to handle ambiguity in emotion annotations. A novel Bayesian approach based on evidential deep learning (EDL) is proposed which learns the emotional content as a distribution. The proposed EDL-based method is extended to dimensional emotion attributes in Chapter 6. Chapter 7 expands the scope beyond emotion and proposes a general framework for simulating human annotations for general subjective tasks, which takes the variability of human opinions into account. Chapter 8 introduces an automatic diagnosis system for depression and Alzheimer's disease (AD). The EDL-based approach is then modified to estimate the confidence of deep-learning-based automatic diagnosis systems. This chapter provides a review of contributions and offers promising prospects for future work.

9.1 Review of Contributions

Emotions are intrinsically part of human mental activity and play a key role in human decision handling, interaction and cognitive processes. Incorporating emotional capabilities into artificial intelligence (AI) can significantly enhance the quality and depth of human-AI interactions. This thesis starts with the discussion of AER based on speech input. Current speech emotion recognition systems face two major challenges: (i) mismatch of research experiments and practical applications; (ii) inconsistent emotion annotations resulting from ambiguous emotion expression and subjective emotion perception.

In response to the first challenge, Chapter 4 investigates AER with automatic segmentation. An integrated system is developed which integrates AER with speaker diarisation and automatic speech recognition (ASR) in a jointly-trained system. The system consists of a shared encoder and four distinct downstream blocks: voice activity detection, speaker classification, ASR, and AER. Taking the audio of a conversation as input, the integrated system finds all speech segments and transcribes the corresponding emotion classes, word sequences, and speaker identities. Two new metrics are proposed to evaluate AER performance with automatic segmentation based on time-weighted emotion and speaker classification errors. The proposed system surpasses separately optimised cascade systems in terms of both efficiency and performance. It also greatly reduces the recognition error of emotional speech.

In response to the second challenge, Chapter 5 explores various approaches to handling data with ambiguous emotions where human annotations diverge. Evidential deep learning (EDL) is adopted to quantify uncertainty in emotion classification which is the first work that detects utterances with ambiguous emotion as out-of-domain samples. Imposing a single ground truth through majority voting can lead to under-representation of minority views. Instead of a single emotion class, it is further proposed to represent emotion as a distribution over emotion classes which provides a more comprehensive representation of emotion content as well as a more inclusive representation of human opinions. A novel algorithm is then proposed that extends EDL to quantify uncertainty in emotion distribution estimation where the emotion labels are assumed to be drawn from an unknown likelihood distribution and the model is trained to learn an utterance-specific prior distribution over the likelihood distribution. The proposed method has been applied to both discrete emotion class labels (in Chapter 5) where a multinomial likelihood and a Dirichlet prior is used and continuous emotion attributes (in Chapter 6) where a Gaussian likelihood and a normal-inverse-gamma prior is used.

Beyond emotion annotation, inconsistent human opinions is a common challenge for general subjective tasks such as speech quality assessment and toxic speech detection since human perception and behaviour exhibit inherent variability due to diverse cognitive processes and subjective interpretation. Chapter 7 investigates human annotator simulation (HAS) which serves as a cost-effective substitute for human evaluation, which takes variability in human evaluation into account to better mimic the way people perceive and interact with the world. A novel meta-learning framework is introduced which treats HAS as a zero-shot density estimation problem. In this framework, two new model classes, conditional integer flows and conditional softmax flows are proposed which account for ordinal and categorical annotations respectively. The proposed method shows superior performance and efficiency for predicting the aggregated behaviour of human annotators, matching the

distribution of human annotations and simulating the inter-annotator disagreements. It is hoped that the proposed method could play a part in promoting inclusivity and fairness for ethical AI practices.

Emotion is naturally linked with mental wellbeing. Chapter 8 studies automatic detection of depression and AD via spontaneous speech. In order to tackle the data sparsity challenge of automatic diagnosis systems, foundation models pretrained on a large amount of speech data are used to transfer knowledge for speech-based automatic depression detection (SDD). It is found that finetuning pretrained speech foundation models for AER improves SDD performance, indicating that emotion information is helpful for depression detection. By integrating representations from multiple foundation models, state-of-the-art results are achieved without the need for oracle transcriptions. To improve the reliability of automatic diagnosis systems, confidence estimation methods are investigated for automatic detection of AD and depression. The EDL approach proposed in Chapter 5 is adapted for confidence estimation which learns a dynamic prior to model the probability of the predictive distribution. It is hoped that this method could help promote reliable and trustworthy automatic diagnostic systems.

9.2 Future Work

Apart from representing emotion as a distribution which has been studied in this thesis, there might be alternative approaches for describing complex emotions. The emergence and prevalence of large language models (LLMs), such as ChatGPT, have revolutionised human-computer interaction which integrate multiple functions in one powerful model and enable users to interact through natural language prompts. LLMs are large neural networks trained on vast amounts of text data, learning the statistical properties of language. They have demonstrated superior capabilities in contextual understanding and interpreting complex instructions. It might be interesting to investigate the direct description of emotions through natural language, which could better capture the nuances of complex emotions. Furthermore, echoing the statement at the beginning of introduction, equipping LLMs with emotional capability is a crucial step on its future development route towards a true form of artificial intelligence.

Variability in human opinions has also been studied in this thesis. Softmax flows and integer flows have been proposed to model such variability. Several extensions could be carried out. First, it can be interesting to incorporate demographic information (*e.g.*, gender, age, cultural background) when simulating human preferences. Second, it may be worthwhile

applying the proposed flow-based methods for other areas of application such as image segmentation for medical tasks.

Apart from depression and Alzheimer's disease that have been studied in the thesis, the detection of other types of mental illnesses such as autism, bipolar disorder, and suicidal risk, which can largely impact the quality of life, has also gathered rising interest. Instead of training separate models for each disease as binary tasks (*e.g.*, one model to detect depression and another to detect AD), developing a model that can simultaneously detect multiple mental health conditions is more advantageous. Such a system would function similarly to a mental health clinician who can diagnose whether an individual is healthy, depressed, autistic, or experiencing a combination of several disorders. This integrated approach mirrors the expertise of a mental health clinician and is more practically useful, providing a comprehensive assessment of an individual's mental health. The aforementioned LLMs with emotional capabilities can be useful for the development of such mental health expert system. Moreover, interpretability in automatic diagnosis systems is another important research direction with the potential to enhance both the reliability of these systems and our understanding of diseases. As AI models become increasingly complex and black-box, concerns about explainability grow. Gaining insights into how these models make decisions can help users better understand and, in turn, decide whether to trust the outcomes.

References

Abdel-Hamid, L. (2020). Egyptian arabic speech emotion recognition using prosodic, spectral and wavelet features. *Speech Communication*, 122:19–30.

Abdelwahab, M. and Busso, C. (2018). Domain adversarial for acoustic emotion recognition. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 26(12):2423–2435.

Abdullah, S. M. S. A., Ameen, S. Y. A., Sadeeq, M. A., and Zeebaree, S. (2021). Multimodal emotion recognition using deep learning. *Journal of Applied Science and Technology Trends*, 2(01):73–79.

Aftab, A., Morsali, A., Ghaemmaghami, S., and Champagne, B. (2022). Light-SERNet: A lightweight fully convolutional neural network for speech emotion recognition. In *Proc. ICASSP*, Singapore.

Al Hanai, T., Ghassemi, M. M., and Glass, J. R. (2018). Detecting depression with audio/text sequence modeling of interviews. In *Proc. Interspeech*, Hyderabad.

AlBadawy, E. A. and Kim, Y. (2018). Joint discrete and continuous emotion prediction using ensemble and end-to-end approaches. In *Proc. ICMI*, Boulder.

Alghowinem, S., Goecke, R., Wagner, M., Epps, J., Breakspear, M., and Parker, G. (2013a). Detecting depression: A comparison between spontaneous and read speech. In *Proc. ICASSP*, Vancouver.

Alghowinem, S., Goecke, R., Wagner, M., Epps, J., Breakspear, M., Parker, G., et al. (2012). From joyous to clinically depressed: Mood detection using spontaneous speech. In *Proc. FLAIRS*, Marco Island.

Alghowinem, S., Goecke, R., Wagner, M., Epps, J., Gedeon, T., Breakspear, M., and Parker, G. (2013b). A comparative study of different classifiers for detecting depression from spontaneous speech. In *Proc. ICASSP*, Vancouver.

Alm, C. O. (2011). Subjective natural language problems: Motivations, applications, characterizations, and implications. In *Proc. ACL*, Portland.

Amini, A., Schwarting, W., Soleimany, A., and Rus, D. (2020). Deep evidential regression. In *Proc. NeurIPS*, Vancouver.

Ando, A., Kobashikawa, S., Kamiyama, H., Masumura, R., Ijima, Y., and Aono, Y. (2018). Soft-target training with ambiguous emotional utterances for DNN-based speech emotion classification. In *Proc. ICASSP*, Brighton.

Asri, L. E., He, J., and Suleman, K. (2016). A sequence-to-sequence model for user simulation in spoken dialogue systems. In *Proc. Interspeech*, San Francisco.

Atcheson, M., Sethu, V., and Epps, J. (2019). Using Gaussian processes with LSTM neural networks to predict continuous-time, dimensional emotion in ambiguous speech. In *Proc. ACII*, Cambridge.

Atmaja, B. T. and Akagi, M. (2019). Speech emotion recognition based on speech segment using lstm with attention model. In *Proc. ICSigSys*, Piscataway.

Atmaja, B. T. and Akagi, M. (2020a). Improving valence prediction in dimensional speech emotion recognition using linguistic information. In *Proc. O-COCOSDA*, Yangon.

Atmaja, B. T. and Akagi, M. (2020b). Multitask learning and multistage fusion for dimensional audiovisual emotion recognition. In *Proc. ICASSP*, Conference held virtually.

Atmaja, B. T. and Akagi, M. (2021). Two-stage dimensional emotion recognition by fusing predictions of acoustic and text networks using SVM. *Speech Communication*, 126:9–21.

Aydoğan, E. and Akcayol, M. A. (2016). A comprehensive survey for sentiment analysis tasks using machine learning techniques. In *Proc. INISTA*, Sinaia.

Ba, J. L., Kiros, J. R., and Hinton, G. E. (2016). Layer normalization. *arXiv preprint arXiv:1607.06450*.

Badshah, A. M., Ahmad, J., Rahim, N., and Baik, S. W. (2017). Speech emotion recognition from spectrograms with deep convolutional neural network. In *Proc. PlatCon*, Jeju.

Baevski, A., Hsu, W.-N., Xu, Q., Babu, A., Gu, J., and Auli, M. (2022). Data2vec: A general framework for self-supervised learning in speech, vision and language. In *Proc. ICML*, Baltimore.

Baevski, A. and Mohamed, A. (2020). Effectiveness of self-supervised pre-training for ASR. In *Proc. ICASSP*, Conference held virtually.

Baevski, A., Zhou, Y., Mohamed, A., and Auli, M. (2020). Wav2vec 2.0: A framework for self-supervised learning of speech representations. In *Proc. NeurIPS*, Vancouver.

Bagher Zadeh, A., Liang, P. P., Poria, S., Cambria, E., and Morency, L.-P. (2018). Multimodal language analysis in the wild: CMU-MOSEI dataset and interpretable dynamic fusion graph. In *Proc. ACL*, Melbourne.

Bahreini, K., Nadolski, R., and Westera, W. (2016). Towards multimodal emotion recognition in e-learning environments. *Interactive Learning Environments*, 24(3):590–605.

Balagopalan, A., Eyre, B., Rudzicz, F., and Novikova, J. (2020). To BERT or not to BERT: Comparing speech and language-based approaches for Alzheimer's disease detection. In *Proc. Interspeech*, Shanghai.

Bänziger, T., Grandjean, D., and Scherer, K. R. (2009). Emotion recognition from expressions in face, voice, and body: the multimodal emotion recognition test (MERT). *Emotion*, 9(5):691.

Bayles, K. A. and Boone, D. R. (1982). The potential of language tasks for identifying senile dementia. *Journal of Speech and Hearing Disorders*, 47(2):210–217.

Beck, A. T., Steer, R. A., Ball, R., and Ranieri, W. F. (1996). Comparison of Beck Depression Inventories-IA and-II in psychiatric outpatients. *Journal of Personality Assessment*, 67(3):588–597.

Bengio, Y., Simard, P., and Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, 5(2):157–166.

Bhosale, S., Chakraborty, R., and Kopparapu, S. (2020). Deep encoded linguistic and acoustic cues for attention based end to end speech emotion recognition. In *Proc. ICASSP*, Barcelona.

Bishop, C. M. (1995). *Neural networks for pattern recognition*. Oxford university press.

Bishop, C. M. and Nasrabadi, N. M. (2006). *Pattern Recognition and Machine Learning*, volume 4. Springer.

Bitouk, D., Verma, R., and Nenkova, A. (2010). Class-level spectral features for emotion recognition. *Speech Communication*, 52(7-8):613–625.

Blundell, C., Cornebise, J., Kavukcuoglu, K., and Wierstra, D. (2015). Weight uncertainty in neural network. In *Proc. ICML*, Lille.

Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., et al. (2021). On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*.

Borade, J. G. and Netak, L. D. (2020). Automated grading of essays: A review. In *Proc. IHCI*, Daegu.

Brodaty, H., Pond, D., Kemp, N. M., Luscombe, G., Harding, L., Berman, K., and Huppert, F. A. (2002). The GPCOG: A new screening test for dementia designed for general practice. *Journal of the American Geriatrics Society*, 50(3):530–534.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. (2020). Language models are few-shot learners. In *Proc. NeurIPS*, Conference held virtually.

Bucks, R. S., Singh, S., Cuerden, J. M., and Wilcock, G. K. (2000). Analysis of spontaneous, conversational speech in dementia of Alzheimer type: Evaluation of an objective technique for analysing lexical performance. *Aphasiology*, 14(1):71–91.

Burkhardt, F., Paeschke, A., Rolfes, M., Sendlmeier, W., and Weiss, B. (2005). A database of German emotional speech. In *Proc. Eurospeech*, Lisbon.

Busso, C., Bulut, M., Lee, C.-C., Kazemzadeh, A., Provost, E., Kim, S., Chang, J., Lee, S., and Narayanan, S. (2008). IEMOCAP: Interactive emotional dyadic motion capture database. *Language Resources and Evaluation*, 42:335–359.

Busso, C., Lee, S., and Narayanan, S. (2007). Using neutral speech models for emotional speech analysis. In *Proc. Interspeech*, Antwerp.

Busso, C., Parthasarathy, S., Burmania, A., AbdelWahab, M., Sadoughi, N., and Provost, E. M. (2017). MSP-IMPROV: An acted corpus of dyadic interactions to study emotion perception. *IEEE Transactions on Affective Computing*, 8(1):67–80.

Cai, J., Xiao, R., Cui, W., Zhang, S., and Liu, G. (2021a). Application of electroencephalography-based machine learning in emotion recognition: A review. *Frontiers in Systems Neuroscience*, 15:729707.

Cai, X., Yuan, J., Zheng, R., Huang, L., and Church, K. (2021b). Speech emotion recognition with multi-task learning. In *Proc. Interspeech*, Brno.

Campbell, E. L., Mesía, R. Y., Docio-Fernandez, L., and García-Mateo, C. (2021). Paralinguistic and linguistic fluency features for Alzheimer's disease detection. *Computer Speech & Language*, 68:101198.

Cao, H., Cooper, D. G., Keutmann, M. K., Gur, R. C., Nenkova, A., and Verma, R. (2014). CREMA-D: Crowd-sourced emotional multimodal actors dataset. *IEEE Transactions on Affective Computing*, 5(4):377–390.

Carletta, J., Ashby, S., Bourban, S., Flynn, M., Guillemot, M., Hain, T., Kadlec, J., Karaiskos, V., Kraaij, W., Kronenthal, M., and Lathoud, G. (2005). The AMI meeting corpus: A pre-announcement. In *Proc. MLMI*, Edinburgh.

Cen, L., Wu, F., Yu, Z. L., and Hu, F. (2016). A real-time speech emotion recognition system and its application in online learning. In *Emotions, Technology, Design, and Learning*, pages 27–46. Academic Press.

Chen, M., Radford, A., Child, R., Wu, J., Jun, H., Luan, D., and Sutskever, I. (2020). Generative pretraining from pixels. In *Proc. ICML*, Conference held virtually.

Chen, M., Zhang, Y., Qiu, M., Guizani, N., and Hao, Y. (2018). SPHA: Smart personal health advisor based on deep analytics. *IEEE Communications Magazine*, 56(3):164–169.

Chen, M. and Zhao, X. (2020). A multi-scale fusion framework for bimodal speech emotion recognition. In *Proc. Interspeech*, Shanghai.

Chen, S., Wang, C., Chen, Z., Wu, Y., Liu, S., Chen, Z., Li, J., Kanda, N., Yoshioka, T., Xiao, X., et al. (2022). WavLM: Large-scale self-supervised pre-training for full stack speech processing. *IEEE Journal of Selected Topics in Signal Processing*, 16(6):1505–1518.

Chen, S., Xing, X., Zhang, W., Chen, W., and Xu, X. (2023a). DWFormer: Dynamic window Transformer for speech emotion recognition. In *Proc. ICASSP*, Rhodes.

Chen, W., Tripp, A., and Hernández-Lobato, J. M. (2023b). Meta-learning adaptive deep kernel gaussian processes for molecular property prediction. In *Proc. ICLR*, Kigali.

Chiu, C.-C., Qin, J., Zhang, Y., Yu, J., and Wu, Y. (2022). Self-supervised learning with random-projection quantizer for speech recognition. In *Proc. ICML*, Baltimore.

Chochlakis, G., Mahajan, G., Baruah, S., Burghardt, K., Lerman, K., and Narayanan, S. (2023). Leveraging label correlations in a multi-label setting: A case study in emotion. In *Proc. ICASSP*, Rhodes.

Chou, H.-C. and Lee, C.-C. (2019). Every rating matters: Joint learning of subjective labels and individual annotators for speech emotion classification. In *Proc. ICASSP*, Brighton.

Chowdary, M. K., Nguyen, T. N., and Hemanth, D. J. (2023). Deep learning-based facial emotion recognition for human–computer interaction applications. *Neural Computing and Applications*, 35(32):23311–23328.

Christiano, P. F., Leike, J., Brown, T., Martic, M., Legg, S., and Amodei, D. (2017). Deep reinforcement learning from human preferences. In *Proc. NeurIPS*, Long Beach.

Cohn, J. F., Kruez, T. S., Matthews, I., Yang, Y., Nguyen, M. H., Padilla, M. T., Zhou, F., and De la Torre, F. (2009). Detecting depression from facial actions and vocal prosody. In *Proc. ACII*, Amsterdam.

Cole, M., Dastoor, D., and Simpson, T. (2015). Hierarchic dementia scale–revised: Instruction manual. *Dementia Training Study Centre*.

Cowen, A. S. and Keltner, D. (2017). Self-report captures 27 distinct categories of emotion bridged by continuous gradients. *Proceedings of the National Academy of Sciences*, 114(38):E7900–E7909.

Cowen, A. S. and Keltner, D. (2021). Semantic space theory: A computational approach to emotion. *Trends in Cognitive Sciences*, 25(2):124–136.

Cui, Z., Wu, W., Zhang, W.-Q., Wu, J., and Zhang, C. (2023). Transferring speech-generic and depression-specific knowledge for Alzheimer's disease detection. In *Proc. ASRU*, Taipei.

Cummins, N., Epps, J., Breakspear, M., and Goecke, R. (2011). An investigation of depressed speech detection: Features and normalization. In *Proc. Interspeech*, Florence.

Cummins, N., Joshi, J., Dhall, A., Sethu, V., Goecke, R., and Epps, J. (2013). Diagnosis of depression by behavioural signals: A multimodal approach. In *Proc. ACM-MM*, Barcelona.

Cummins, N., Pan, Y., Ren, Z., Fritsch, J., Nallanthighal, V. S., Christensen, H., Blackburn, D., Schuller, B. W., Magimai-Doss, M., Strik, H., et al. (2020). A comparison of acoustic and linguistics methodologies for Alzheimer's dementia recognition. In *Proc. Interspeech*, Shanghai.

Cummins, N., Scherer, S., Krajewski, J., Schnieder, S., Epps, J., and Quatieri, T. F. (2015). A review of depression and suicide risk assessment using speech analysis. *Speech Communication*, 71:10–49.

Dadebayev, D., Goh, W. W., and Tan, E. X. (2022). EEG-based emotion recognition: Review of commercial EEG devices and machine learning techniques. *Journal of King Saud University-Computer and Information Sciences*, 34(7):4385–4401.

Dai, Z., Yang, Z., Yang, Y., Carbonell, J. G., Le, Q., and Salakhutdinov, R. (2019). Transformer-XL: Attentive language models beyond a fixed-length context. In *Proc. ACL*, Florence.

Daneshfar, F., Kabudian, S. J., and Neekabadi, A. (2020). Speech emotion recognition using hybrid spectral-prosodic features of speech signal/glottal waveform, metaheuristic-based dimensionality reduction, and gaussian elliptical basis function network classifier. *Applied Acoustics*, 166:107360.

Dang, T., Sethu, V., and Ambikairajah, E. (2018). Dynamic multi-rater Gaussian mixture regression incorporating temporal dependencies of emotion uncertainty using Kalman filters. In *Proc. ICASSP*, Calgary.

Dang, T., Sethu, V., Epps, J., and Ambikairajah, E. (2017). An investigation of emotion prediction uncertainty using gaussian mixture regression. In *Proc. Interspeech*, Stockholm.

Dauphin, Y. N., Fan, A., Auli, M., and Grangier, D. (2017). Language modeling with gated convolutional networks. In *Proc. ICML*, Sydney.

Davani, A. M., Díaz, M., and Prabhakaran, V. (2022). Dealing with disagreements: Looking beyond the majority vote in subjective annotations. *Transactions of the Association for Computational Linguistics*, 10:92–110.

de la Fuente Garcia, S., Ritchie, C. W., and Luz, S. (2020). Artificial intelligence, speech, and language processing approaches to monitoring Alzheimer's disease: A systematic review. *Journal of Alzheimer's Disease*, 78(4):1547–1574.

Dempster, A. P. (1968). A generalization of Bayesian inference. *Journal of the Royal Statistical Society: Series B (Methodological)*, 30(2):205–232.

Deng, J., Han, W., and Schuller, B. (2012). Confidence measures for speech emotion recognition: A start. In *Speech Communication; 10. ITG Symposium*.

Deng, J., Xu, X., Zhang, Z., Frühholz, S., and Schuller, B. (2017). Universum autoencoder-based domain adaptation for speech emotion recognition. *IEEE Signal Processing Letters*, 24(4):500–504.

Der Kiureghian, A. and Ditlevsen, O. (2009). Aleatory or epistemic? Does it matter? *Structural Safety*, 31(2):105–112.

DeVault, D., Artstein, R., Benn, G., Dey, T., Fast, E., Gainer, A., Georgila, K., Gratch, J., Hartholt, A., Lhommet, M., Lucas, G., Marsella, S., Morbini, F., Nazarian, A., Scherer, S., Stratou, G., Suri, A., Traum, D., Wood, R., Xu, Y., Rizzo, A., and Morency, L. P. (2014). SimSensei Kiosk: A virtual human interviewer for healthcare decision support. In *Proc. AAMAS*, Paris.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proc. NAACL*, Minneapolis.

Dimitrova-Grekow, T. and Konopko, P. (2019). New parameters for improving emotion recognition in human voice. In *Proc. SMC*, Bari.

Dinh, L., Sohl-Dickstein, J., and Bengio, S. (2017). Density estimation using real NVP. In *Proc. ICLR*, Toulon.

Dinkel, H., Wu, M., and Yu, K. (2019). Text-based depression detection on sparse data. *arXiv preprint arXiv:1904.05154*.

Dissanayake, V., Seneviratne, S., Suriyaarachchi, H., Wen, E., and Nanayakkara, S. (2022). Self-supervised representation fusion for speech and wearable based emotion recognition. In *Proc. Interspeech*, Incheon.

Dixon, L., Li, J., Sorensen, J., Thain, N., and Vasserman, L. (2018). Measuring and mitigating unintended bias in text classification. In *Proc. AAAI*, New Orleans.

Djuric, N., Zhou, J., Morris, R., Grbovic, M., Radosavljevic, V., and Bhamidipati, N. (2015). Hate speech detection with comment embeddings. In *Proc. WWW*, Florence.

Du, G., Long, S., and Yuan, H. (2020). Non-contact emotion recognition combining heart rate and facial expression for interactive gaming environments. *IEEE Access*, 8:11896–11906.

Duan, J., Yu, S., Tan, H. L., Zhu, H., and Tan, C. (2022). A survey of embodied AI: From simulators to research tasks. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 6(2):230–244.

Duchi, J., Hazan, E., and Singer, Y. (2011). Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(7).

Edition, F. et al. (2013). Diagnostic and statistical manual of mental disorders. *Am Psychiatric Assoc*, 21(21):591–643.

Ekman, P. (1971). Universals and cultural differences in facial expressions of emotion. In *Nebraska symposium on motivation*. University of Nebraska Press.

Ekman, P. et al. (1999). Basic emotions. *Handbook of cognition and emotion*, 98(45-60):16.

El Ayadi, M., Kamel, M. S., and Karray, F. (2011). Survey on speech emotion recognition: Features, classification schemes, and databases. *Pattern Recognition*, 44(3):572–587.

Elman, J. L. (1990). Finding structure in time. *Cognitive science*, 14(2):179–211.

Evermann, G. and Woodland, P. C. (2000). Large vocabulary decoding and confidence estimation using word posterior probabilities. In *Proc. ICASSP*, Istanbul.

Fayek, H., Lech, M., and Cavedon, L. (2016). Modeling subjectiveness in emotion recognition with deep neural networks: Ensembles vs soft labels. In *Proc. IJCNN*, Vancouver.

Feng, H., Ueno, S., and Kawahara, T. (2020). End-to-end speech emotion recognition combined with acoustic-to-word ASR model. In *Proc. Interspeech*, Shanghai.

Fernandez, R. (2004). *A computational model for the automatic recognition of affect in speech*. PhD thesis, Massachusetts Institute of Technology.

Fleiss, J. L. (1971). Measuring nominal scale agreement among many raters. *Psychological Bulletin*, 76(5):378.

Folstein, M. F., Folstein, S. E., and McHugh, P. R. (1975). "Mini-mental state": A practical method for grading the cognitive state of patients for the clinician. *Journal of Psychiatric Research*, 12(3):189–198.

Forbes-McKay, K. E. and Venneri, A. (2005). Detecting subtle spontaneous language decline in early Alzheimer's disease with a picture description task. *Neurological sciences*, 26(4):243–254.

Fraser, K. C., Meltzer, J. A., and Rudzicz, F. (2016). Linguistic features identify Alzheimer's disease in narrative speech. *Journal of Alzheimer's Disease*, 49(2):407–422.

Gal, Y. and Ghahramani, Z. (2016). Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In *Proc. ICML*, New York.

Gao, Y., Li, B., Wang, N., and Zhu, T. (2017). Speech emotion recognition using local and global features. In *Proc. BI*, Beijing. Springer.

Gatt, J., Nemeroff, C., Dobson-Stone, C., Paul, R., Bryant, R., Schofield, P., Gordon, E., Kemp, A., and Williams, L. (2009). Interactions between BDNF Val66Met polymorphism and early life stress predict brain and arousal pathways to syndromal depression and anxiety. *Molecular Psychiatry*, 14(7):681–695.

Gehring, J., Auli, M., Grangier, D., Yarats, D., and Dauphin, Y. N. (2017). Convolutional sequence to sequence learning. In *Proc. ICML*, Sydney.

Ghosh, S., Laksana, E., Morency, L.-P., and Scherer, S. (2016). Representation learning for speech emotion recognition. In *Proc. Interspeech*, San Francisco.

Ghriss, A., Yang, B., Rozgic, V., Shriberg, E., and Wang, C. (2022). Sentiment-aware automatic speech recognition pre-training for enhanced speech emotion recognition. In *Proc. ICASSP*, Singapore.

Gideon, J., McInnis, M. G., and Provost, E. M. (2019). Improving cross-corpus speech emotion recognition with adversarial discriminative domain generalization (ADDoG). *IEEE Transactions on Affective Computing*, 12(4):1055–1068.

Gill, P. E., Murray, W., and Wright, M. H. (1981). *Practical Optimization*. SIAM.

Glowinski, D., Camurri, A., Volpe, G., Dael, N., and Scherer, K. (2008). Technique for automatic emotion recognition by body gesture analysis. In *Proc. CVPR workshop*, Anchorage.

Goncalves, L. and Busso, C. (2022). AuxFormer: Robust approach to audiovisual emotion recognition. In *Proc. ICASSP*, Singapore.

Gong, Y. and Poellabauer, C. (2017). Topic modeling based multi-modal depression detection. In *Proc. ACM-MM*, Mountain View.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT press.

Grandjean, D., Sander, D., and Scherer, K. R. (2008). Conscious emotional experience emerges as a function of multilevel, appraisal-driven response synchronization. *Consciousness and Cognition*, 17(2):484–495.

Gratch, J., Artstein, R., Lucas, G., Stratou, G., Scherer, S., Nazarian, A., Wood, R., Boberg, J., DeVault, D., Marsella, S., Traum, D., Rizzo, S., and Morency, L.-P. (2014). The distress analysis interview corpus of human and computer interviews. In *Proc. LREC*, Reykjavik.

Graves, A., Fernández, S., Gomez, F., and Schmidhuber, J. (2006). Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proc. ICML*, Pittsburgh.

Grimm, M. and Kroschel, K. (2005). Evaluation of natural emotions using self assessment manikins. In *Proc. ASRU*, Cancun.

Guerrero-Cristancho, J. S., Vásquez-Correa, J. C., and Orozco-Arroyave, J. R. (2020). Word-embeddings and grammar features to detect language disorders in Alzheimer's disease patients. *TecnoLógicas*, 23(47):63–75.

Guidi, A., Gentili, C., Scilingo, E. P., and Vanello, N. (2019). Analysis of speech features and personality traits. *Biomedical signal processing and control*, 51:1–7.

Gulati, A., Qin, J., Chiu, C.-C., Parmar, N., Zhang, Y., Yu, J., Han, W., Wang, S., Zhang, Z., Wu, Y., and Pang, R. (2020). Conformer: Convolution-augmented Transformer for speech recognition. In *Proc. Interspeech*, Shanghai.

Gunes, H. and Schuller, B. (2013). Categorical and dimensional affect analysis in continuous input: Current trends and future directions. *Image and Vision Computing*, 31(2):120–136.

Gunes, H., Schuller, B., Pantic, M., and Cowie, R. (2011). Emotion representation, analysis and synthesis in continuous space: A survey. In *Proc. FG*, Santa Barbara.

Guo, C., Pleiss, G., Sun, Y., and Weinberger, K. Q. (2017). On calibration of modern neural networks. In *Proc. ICML*, Sydney.

Gür, I., Hakkani-Tür, D., Tür, G., and Shah, P. (2018). User modeling for task oriented dialogues. In *Proc. SLT*, Athens.

Haider, F., De La Fuente, S., and Luz, S. (2019). An assessment of paralinguistic acoustic features for detection of Alzheimer's dementia in spontaneous speech. *IEEE Journal of Selected Topics in Signal Processing*, 14(2):272–281.

Hall, J. A., Harrigan, J. A., and Rosenthal, R. (1995). Nonverbal behavior in clinician-patient interaction. *Applied and Preventive Psychology*, 4(1):21–37.

Hamilton, M. (1986). The Hamilton rating scale for depression. In *Assessment of depression*, pages 143–152. Springer.

Han, J., Zhang, Z., Ren, Z., and Schuller, B. (2021). Exploring perception uncertainty for emotion recognition in dyadic conversation and music listening. *Cognitive Computation*, 13(2):231–240.

Han, J., Zhang, Z., Schmitt, M., Pantic, M., and Schuller, B. (2017). From hard to soft: Towards more human-like emotion recognition by modelling the perception uncertainty. In *Proc. ACM-MM*, Mountain View.

Haque, A., Guo, M., Miner, A. S., and Fei-Fei, L. (2018). Measuring depression symptom severity from spoken language and 3D facial expressions. *arXiv preprint arXiv:1811.08592*.

Harrison, P., Marjieh, R., Adolfi, F., van Rijn, P., Anglada-Tort, M., Tchernichovski, O., Larrouy-Maestri, P., and Jacoby, N. (2020). Gibbs sampling with people. In *Proc. NeurIPS*, Vancouver.

Haselton, M. G., Nettle, D., and Andrews, P. W. (2015). The evolution of cognitive bias. *The Handbook of Evolutionary Psychology*, pages 724–746.

Hastings, W. K. (1970). Monte Carlo sampling methods using Markov chains and their applications.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proc. CVPR*, Las Vegas.

He, L., Chan, J. C.-W., and Wang, Z. (2021). Automatic depression recognition using CNN with attention mechanism from videos. *Neurocomputing*, 422:165–175.

Hinton, G. E. and Zemel, R. (1993). Autoencoders, minimum description length and helmholtz free energy. In *Proc. NeurIPS*, Denver.

Hirschberg, J., Liscombe, J., and Venditti, J. (2003). Experiments in emotional speech. In *Proc. SSPR*, Tokyo.

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

Hoogeboom, E., Nielsen, D., Jaini, P., Forré, P., and Welling, M. (2021). Argmax flows and multinomial diffusion: Learning categorical distributions. In *Proc. NeurIPS*, Conference held virtually.

Hsu, W.-N., Bolte, B., Tsai, Y.-H. H., Lakhotia, K., Salakhutdinov, R., and Mohamed, A. (2021). HuBERT: Self-supervised speech representation learning by masked prediction of hidden units. *IEEE Transactions on Audio, Speech, and Language Processing*, 29:3451–3460.

Hu, D., Hu, X., and Xu, X. (2022). Multiple enhancements to LSTM for learning emotion-salient features in speech emotion recognition. In *Proc. Interspeech*, Incheon.

Huang, Y., Wen, H., Qing, L., Jin, R., and Xiao, L. (2021). Emotion recognition based on body and context fusion in the wild. In *Proc. ICCV*, Montreal.

Hutchinson, B., Prabhakaran, V., Denton, E., Webster, K., Zhong, Y., and Denuyl, S. (2020). Social biases in NLP models as barriers for persons with disabilities. In *Proc. ACL*, Conference held virtually.

Iandola, F., Moskewicz, M., Karayev, S., Girshick, R., Darrell, T., and Keutzer, K. (2014). Densenet: Implementing efficient convnet descriptor pyramids. *arXiv preprint arXiv:1404.1869*.

Iliou, T. and Anagnostopoulos, C.-N. (2009). Comparison of different classifiers for emotion recognition. In *Panhellenic Conference on Informatics*, Corfu.

Ito, K. and Johnson, L. (2017). The LJ speech dataset. https://keithito.com/LJ-Speech-Dataset/.

Ivanov, A. V., Jalalvand, S., Gretter, R., and Falavigna, D. (2013). Phonetic and anthropometric conditioning of msa-kst cognitive impairment characterization system. In *Proc. ASRU*, Olomouc.

Jaitly, N. and Hinton, G. E. (2013). Vocal tract length perturbation (VTLP) improves speech recognition. In *Proc. ICML Workshop on Deep Learning for Audio, Speech and Language*, Atlanta.

Jiang, H. (2005). Confidence measures for speech recognition: A survey. *Speech communication*, 45(4):455–470.

Jordan, M. I. (1997). Serial order: A parallel distributed processing approach. In *Advances in Psychology*, volume 121, pages 471–495. Elsevier.

Joshi, J., Goecke, R., Alghowinem, S., Dhall, A., Wagner, M., Epps, J., Parker, G., and Breakspear, M. (2013). Multimodal assistive technologies for depression diagnosis and monitoring. *Journal on Multimodal User Interfaces*, 7(3):217–228.

Jsang, A. (2018). *Subjective Logic: A formalism for reasoning under uncertainty*. Springer Publishing Company, Incorporated.

Ju, X., Zhang, D., Li, J., and Zhou, G. (2020). Transformer-based label set generation for multi-modal multi-label emotion detection. In *Proc. ACM-MM*, Seattle.

Kavé, G. and Levy, Y. (2003). Morphology in picture descriptions provided by persons with Alzheimer's disease.

Kearns, J. (2014). LibriVox: Free public domain audiobooks. *Reference Reviews*, 28(1):7–8.

Kendall, A. and Gal, Y. (2017). What uncertainties do we need in Bayesian deep learning for computer vision? In *Proc. NeurIPS*, Long Beach.

Kerkeni, L., Serrestou, Y., Raoof, K., Mbarki, M., Mahjoub, M. A., and Cleder, C. (2019). Automatic speech emotion recognition using an optimal combination of features based on EMD-TKEO. *Speech communication*, 114:22–35.

Kim, J., An, Y., and Kim, J. (2022). Improving speech emotion recognition through focus and calibration attention mechanisms. In *Proc. Interspeech*, Incheon.

Kim, Y. and Kim, J. (2018). Human-like emotion recognition: Multi-label learning from noisy labeled audio-visual expressive speech. In *Proc. ICASSP*, Calgary.

Kim, Y., Lee, H., and Provost, E. M. (2013). Deep learning for robust feature generation in audiovisual emotion recognition. In *Proc. ICASSP*, Vancouver.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kingma, D. P. and Welling, M. (2014). Auto-encoding variational Bayes. In *Proc. ICLR*, Banff.

Ko, T., Peddinti, V., Povey, D., and Khudanpur, S. (2015). Audio augmentation for speech recognition. In *Proc. Interspeech*, Dresden.

Kolesnikov, A., Dosovitskiy, A., Weissenborn, D., Heigold, G., Uszkoreit, J., Beyer, L., Minderer, M., Dehghani, M., Houlsby, N., Gelly, S., Unterthiner, T., and Zhai, X. (2021). An image is worth 16x16 words: Transformers for image recognition at scale. In *Proc. ICLR*, Vienna.

Koolagudi, S. G. and Rao, K. S. (2012). Emotion recognition from speech: A review. *International Journal of Speech Technology*, 15:99–117.

Kossaifi, J., Walecki, R., Panagakis, Y., Shen, J., Schmitt, M., Ringeval, F., Han, J., Pandit, V., Toisoul, A., Schuller, B., et al. (2019). SEWA DB: A rich database for audio-visual emotion and sentiment research in the wild. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(3):1022–1040.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. In *Proc. NeurIPS*, Lake Tahoe.

Kroenke, K., Spitzer, R. L., and Williams, J. B. (2001). The PHQ-9: Validity of a brief depression severity measure. *Journal of General Internal Medicine*, 16(9):606–613.

Lakshminarayanan, B., Pritzel, A., and Blundell, C. (2017). Simple and scalable predictive uncertainty estimation using deep ensembles. In *Proc. NeurIPS*, Long Beach.

Le, D., Aldeneh, Z., and Provost, E. M. (2017). Discretized continuous speech emotion recognition with multi-task deep recurrent neural network. In *Proc. Interspeech*, Stockholm.

LeCun, Y., Bengio, Y., et al. (1995). Convolutional networks for images, speech, and time series. *The Handbook of Brain Theory and Neural Networks*, 3361(10):1995.

Lee, C. M. and Narayanan, S. S. (2005). Toward detecting emotions in spoken dialogs. *IEEE Transactions on Speech and Audio Processing*, 13(2):293–303.

Lee, J. and Tashev, I. (2015). High-level feature representation using recurrent neural network for speech emotion recognition. In *Proc. Interspeech*, Dresden.

Leem, S.-G., Fulford, D., Onnela, J.-P., Gard, D., and Busso, C. (2022). Not all features are equal: Selection of robust features for speech emotion recognition in noisy environments. In *Proc. ICASSP*, Singapore.

Leng, Y., Tan, X., Zhao, S., Soong, F., Li, X.-Y., and Qin, T. (2021). MBNet: MOS prediction for synthesized speech with mean-bias network. In *Proc. ICASSP*, Toronto.

Li, H., Wang, N., Yu, Y., Yang, X., and Gao, X. (2021a). LBAN-IL: A novel method of high discriminative representation for facial expression recognition. *Neurocomputing*, 432:159–169.

Li, J., Yu, J., Ye, Z., Wong, S., Mak, M., Mak, B., Liu, X., and Meng, H. (2021b). A comparative study of acoustic and linguistic features classification for Alzheimer's disease detection. In *Proc. ICASSP*, Toronto.

Li, Q., Qiu, D., Zhang, Y., Li, B., He, Y., Woodland, P. C., Cao, L., and Strohman, T. (2021c). Confidence estimation for attention-based sequence-to-sequence models for speech recognition. In *Proc. ICASSP*, Toronto.

Li, S. and Deng, W. (2020). Deep facial expression recognition: A survey. *IEEE transactions on affective computing*, 13(3):1195–1215.

Li, S., Shi, F., Pu, F., Li, X., Jiang, T., Xie, S., and Wang, Y. (2007). Hippocampal shape analysis of Alzheimer disease based on machine learning methods. *American Journal of Neuroradiology*, 28(7):1339–1345.

Li, Y., Bell, P., and Lai, C. (2022). Fusing ASR outputs in joint training for speech emotion recognition. In *Proc. ICASSP*, Singapore.

Li, Y., Mohamied, Y., Bell, P., and Lai, C. (2023). Exploration of a self-supervised speech model: A study on emotional corpora. In *Proc. SLT*, Doha.

Li, Y., Tao, J., Schuller, B., Shan, S., Jiang, D., and Jia, J. (2018). MEC 2017: Multimodal emotion recognition challenge. In *Proc. ACII Asia*, Beijing.

Liang, B., Su, H., Gui, L., Cambria, E., and Xu, R. (2022). Aspect-based sentiment analysis via affective knowledge enhanced graph convolutional networks. *Knowledge-Based Systems*, 235:107643.

Likitha, M., Gupta, S. R. R., Hasitha, K., and Raju, A. U. (2017). Speech based human emotion recognition using MFCC. In *Proc. WiSPNET*, Chennai.

Lin, H.-C., Lubis, N., Hu, S., van Niekerk, C., Geishauser, C., Heck, M., Feng, S., and Gasic, M. (2021). Domain-independent user simulation with transformers for task-oriented dialogue systems. In *Proc. SIGDIAL*, Singapore.

Lin, K., Xia, F., Wang, W., Tian, D., and Song, J. (2016). System design for big data application in emotion-aware healthcare. *IEEE Access*, 4:6901–6909.

Liu, P., Li, K., and Meng, H. (2020). Group gated fusion on attention-based bidirectional alignment for multimodal emotion recognition. In *Proc. Interspeech*, Shanghai.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. (2019). RoBERTa: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Lo, C.-C., Fu, S.-W., Huang, W.-C., Wang, X., Yamagishi, J., Tsao, Y., and Wang, H.-M. (2019). MOSNet: Deep learning-based objective assessment for voice conversion. In *Proc. Interspeech*, Graz.

Lotfian, R. and Busso, C. (2019). Building naturalistic emotionally balanced speech corpus by retrieving emotional speech from existing podcast recordings. *IEEE Transactions on Affective Computing*, 10(4):471–483.

Low, L.-S. A., Maddage, N. C., Lech, M., Sheeber, L. B., and Allen, N. B. (2010). Detection of clinical depression in adolescents' speech during family interactions. *IEEE Transactions on Biomedical Engineering*, 58(3):574–586.

Lu, D., Popuri, K., Ding, G. W., Balachandar, R., and Beg, M. F. (2018). Multimodal and multiscale deep neural networks for the early diagnosis of Alzheimer's disease using structural MR and FDG-PET images. *Scientific Reports*, 8(1):1–13.

Lu, Z., Cao, L., Zhang, Y., Chiu, C., and Fan, J. (2020). Speech sentiment analysis via pre-trained features from end-to-end ASR models. In *Proc. ICASSP*, Barcelona.

Lugger, M. and Yang, B. (2007). The relevance of voice quality features in speaker independent emotion recognition. In *Proc. ICASSP*, Honolulu.

Luscher, B., Shen, Q., and Sahir, N. (2011). The GABAergic deficit hypothesis of major depressive disorder. *Molecular Psychiatry*, 16(4):383–406.

Luz, S. (2017). Longitudinal monitoring and detection of Alzheimer's type dementia from spontaneous speech data. In *Proc. CBMS*, Thessaloniki.

Luz, S., de la Fuente, S., and Albert, P. (2018). A method for analysis of patient speech in dialogue for dementia detection. *arXiv preprint arXiv:1811.09919*.

Luz, S., Haider, F., de la Fuente, S., Fromm, D., and MacWhinney, B. (2020). Alzheimer's dementia recognition through spontaneous speech: The ADReSS challenge. In *Proc. Interspeech*, Shanghai.

Ma, K., Zeng, K., and Wang, Z. (2015). Perceptual quality assessment for multi-exposure image fusion. *IEEE Transactions on Image Processing*, 24(11):3345–3356.

Ma, X., Yang, H., Chen, Q., Huang, D., and Wang, Y. (2016). DepAudioNet: An efficient deep model for audio based depression classification. In *Proc. ACM-MM*, Amsterdam.

Majumder, N., Hazarika, D., Gelbukh, A., Cambria, E., and Poria, S. (2018). Multimodal sentiment analysis using hierarchical fusion with context modeling. *Knowledge-Based Systems*, 161:124–133.

Makiuchi, M. R., Uto, K., and Shinoda, K. (2021). Multimodal emotion recognition with high-level speech and text features. In *Proc. ASRU*, Cartagena.

Malinin, A. and Gales, M. (2018). Predictive uncertainty estimation via prior networks. In *Proc. NeurIPS*, Montreal.

Maniati, G., Vioni, A., Ellinas, N., Nikitaras, K., Klapsas, K., Sung, J. S., Jho, G., Chalamandaris, A., and Tsiakoulis, P. (2022). SOMOS: The Samsung open MOS dataset for the evaluation of neural text-to-speech synthesis. In *Proc. Interspeech*, Incheon.

Mao, Q., Dong, M., Huang, Z., and Zhan, Y. (2014). Learning salient features for speech emotion recognition using convolutional neural networks. *IEEE Transactions on Multimedia*, 16(8):2203–2213.

Marín-Morales, J., Llinares, C., Guixeres, J., and Alcañiz, M. (2020). Emotion recognition in immersive virtual reality: From statistics to affective computing. *Sensors*, 20(18):5163.

Marsella, S. and Gratch, J. (2014). Computationally modeling human emotion. *Communications of the ACM*, 57(12):56–67.

Mathew, B., Saha, P., Yimam, S. M., Biemann, C., Goyal, P., and Mukherjee, A. (2021). HateXplain: A benchmark dataset for explainable hate speech detection. In *Proc. AAAI*, Vancouver.

Matthies, H. G. (2007). Quantifying uncertainty: Modern computational representation of probability and applications. In *Extreme Man-made and Natural Hazards in Dynamics of Structures*, pages 105–135. Springer.

Mattson, M. P. (2004). Pathways towards and away from Alzheimer's disease. *Nature*, 430(7000):631–639.

McKeown, G., Valstar, M., Cowie, R., Pantic, M., and Schröder, M. (2012). The SEMAINE database: Annotated multimodal records of emotionally colored conversations between a person and a limited agent. *IEEE Transactions on Affective Computing*, 3(1):5–17.

Meghanani, A., Anoop, C., and Ramakrishnan, A. (2021). An exploration of log-mel spectrogram and MFCC features for Alzheimer's dementia recognition from spontaneous speech. In *Proc. SLT*, Conference held virtually.

Mehrabian, A. (1980). Basic dimensions for a general psychological theory: Implications for personality, social, environmental, and developmental studies.

Mehrabian, A. (1996). Pleasure-arousal-dominance: A general framework for describing and measuring individual differences in temperament. *Current Psychology*, 14:261–292.

Ménard, M., Richard, P., Hamdi, H., Daucé, B., and Yamaguchi, T. (2015). Emotion recognition based on heart rate and skin conductance. In *PhyCS*, pages 26–32.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. (1953). Equation of state calculations by fast computing machines. *The journal of chemical physics*, 21(6):1087–1092.

Mihalcea, R. and Liu, H. (2006). A corpus-based approach to finding happiness. In *Proc. AAAI Spring Symposium*, Stanford.

Mirheidari, B., Blackburn, D., Walker, T., Venneri, A., Reuber, M., and Christensen, H. (2018). Detecting signs of dementia using word vector representations. In *Proc. Interspeech*, Hyderabad.

Mirsamadi, S., Barsoum, E., and Zhang, C. (2017). Automatic speech emotion recognition using recurrent neural networks with local attention. In *Proc. ICASSP*, New Orleans.

Mirzaei, G., Adeli, A., and Adeli, H. (2016). Imaging and machine learning techniques for diagnosis of Alzheimer's disease. *Reviews in the Neurosciences*, 27(8):857–870.

Mitra, V., Chien, H.-Y. S., Kowtha, V., Cheng, J. Y., and Azemi, E. (2022). Speech emotion: Investigating model representations, multi-task learning and knowledge distillation. In *Proc. Interspeech*, Incheon.

Moore II, E., Clements, M. A., Peifer, J. W., and Weisser, L. (2007). Critical analysis of the impact of glottal features in the classification of clinical depression in speech. *IEEE Transactions on Biomedical Engineering*, 55(1):96–107.

Moradi, E., Pepe, A., Gaser, C., Huttunen, H., Tohka, J., Alzheimer's Disease Neuroimaging Initiative, et al. (2015). Machine learning framework for early MRI-based Alzheimer's conversion prediction in MCI subjects. *Neuroimage*, 104:398–412.

Morais, E., Hoory, R., Zhu, W., Gat, I., Damasceno, M., and Aronowitz, H. (2022). Speech emotion recognition using self-supervised features. In *Proc. ICASSP*, Singapore.

Morris, J. C. (1991). The clinical dementia rating (CDR): Current version and scoring rules. *Young*, 41:1588–1592.

Mousa, A. and Schuller, B. (2017). Contextual bidirectional long short-term memory recurrent neural network language models: A generative approach to sentiment analysis. In *Proc. EACL*, Valencia.

Mower, E., Matarić, M. J., and Narayanan, S. (2010). A framework for automatic human emotion classification using emotion profiles. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(5):1057–1070.

Mower, E., Metallinou, A., chun Lee, C., Kazemzadeh, A., Busso, C., Lee, S., and Narayanan, S. (2009). Interpreting ambiguous emotional expressions. In *Proc. ACII*, Amsterdam.

Murphy, K. P. (2012). *Machine learning: A probabilistic perspective*. MIT press.

Naeini, M. P., Cooper, G., and Hauskrecht, M. (2015). Obtaining well calibrated probabilities using Bayesian binning. In *Proc. AAAI*, Austin.

Nagrani, A., Chung, J. S., and Zisserman, A. (2017). VoxCeleb: A large-scale speaker identification dataset. *arXiv preprint arXiv:1706.08612*.

Nasreddine, Z. S., Phillips, N. A., Bédirian, V., Charbonneau, S., Whitehead, V., Collin, I., Cummings, J. L., and Chertkow, H. (2005). The Montreal cognitive assessment, MoCA: A brief screening tool for mild cognitive impairment. *Journal of the American Geriatrics Society*, 53(4):695–699.

Nediyanchath, A., Paramasivam, P., and Yenigalla, P. (2020). Multi-head attention for speech emotion recognition with auxiliary learning of gender recognition. In *Proc. ICASSP*, Barcelona.

Neiberg, D., Elenius, K., and Laskowski, K. (2006). Emotion recognition in spontaneous speech using GMMs. In *Proc. ICSLP*, Pittsburgh.

Neumann, M. and Vu, N. T. (2019). Improving speech emotion recognition with unsupervised representation learning on unlabeled speech. In *Proc. ICASSP*, Brighton.

Nguyen, T. Q. and Salazar, J. (2019). Transformers without tears: Improving the normalization of self-attention. In *Proc. IWSLT*, Hong Kong.

Ooi, K. E. B., Lech, M., and Allen, N. B. (2012). Multichannel weighted speech classification system for prediction of major depression in adolescents. *IEEE Transactions on Biomedical Engineering*, 60(2):497–506.

Oord, A. v. d., Li, Y., and Vinyals, O. (2018). Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.

Ortiz, A., Lozano, F., Gorriz, J. M., Ramirez, J., Martinez Murcia, F. J., Alzheimer's Disease Neuroimaging Initiative, et al. (2018). Discriminative sparse features for Alzheimer's disease diagnosis using multimodal image data. *Current Alzheimer Research*, 15(1):67–79.

Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., et al. (2022). Training language models to follow instructions with human feedback. In *Proc. NeurIPS*, New Orleans.

Pampouchidou, A., Simos, P. G., Marias, K., Meriaudeau, F., Yang, F., Pediaditis, M., and Tsiknakis, M. (2017). Automatic assessment of depression based on visual cues: A systematic review. *IEEE Transactions on Affective Computing*, 10(4):445–470.

Pan, Y., Mirheidari, B., Harris, J. M., Thompson, J. C., Jones, M., Snowden, J. S., Blackburn, D., and Christensen, H. (2021). Using the outputs of different automatic speech recognition paradigms for acoustic-and bert-based Alzheimer's dementia detection through spontaneous speech. In *Proc. Interspeech*, Brno.

Panayotov, V., Chen, G., Povey, D., and Khudanpur, S. (2015). LibriSpeech: An ASR corpus based on public domain audio books. In *Proc. ICASSP*, South Brisbane.

Pappagari, R., Wang, T., Villalba, J., Chen, N., and Dehak, N. (2020). X-vectors meet emotions: A study on dependencies between emotion and speaker recognition. In *Proc. ICASSP*, Barcelona.

Park, D. S., Chan, W., Zhang, Y., Chiu, C.-C., Zoph, B., Cubuk, E. D., and Le, Q. V. (2019). SpecAugment: A simple data augmentation method for automatic speech recognition. *arXiv preprint arXiv:1904.08779*.

Parmar, N., Vaswani, A., Uszkoreit, J., Kaiser, L., Shazeer, N., Ku, A., and Tran, D. (2018). Image transformer. In *Proc. ICML*, Stockholm.

Parry, J., Palaz, D., Clarke, G., Lecomte, P., Mead, R., Berger, M., and Hofer, G. (2019). Analysis of deep learning architectures for cross-corpus speech emotion recognition. In *Proc. Interspeech*, Graz.

Pasad, A., Chou, J.-C., and Livescu, K. (2021). Layer-wise analysis of a self-supervised speech representation model. In *Proc. ASRU*, Cartagena.

Pasad, A., Shi, B., and Livescu, K. (2023). Comparative layer-wise analysis of self-supervised speech models. In *Proc. ICASSP*, Rhodes.

Pastoriza-Domínguez, P., Torre, I. G., Diéguez-Vide, F., Gómez-Ruiz, I., Geladó, S., Bello-López, J., Ávila-Rivera, A., Matías-Guiu, J. A., Pytel, V., and Hernández-Fernández, A. (2022). Speech pause distribution as an early marker for Alzheimer's disease. *Speech Communication*, 136:107–117.

Patton, B., Agiomyrgiannakis, Y., Terry, M., Wilson, K., Saurous, R. A., and Sculley, D. (2016). AutoMOS: Learning a non-intrusive assessor of naturalness-of-speech. In *Proc, NeurIPS Workshop*, Barcelona.

Pellegrini, E., Ballerini, L., Hernandez, M. d. C. V., M, C., González-Castro, V., Anblagan, D., Danso, S., Muñoz-Maniega, S., Job, D., Pernet, C., et al. (2018). Machine learning of neuroimaging for assisted diagnosis of cognitive impairment and dementia: A systematic review. *Alzheimer's & Dementia: Diagnosis, Assessment & Disease Monitoring*, 10:519–535.

Picard, R. W. (2000). *Affective computing*. MIT press.

Pineda, F. (1987). Generalization of back propagation to recurrent and higher order neural networks. *Neural Information Processing Systems*.

Plank, B., Hovy, D., and Søgaard, A. (2014). Learning part-of-speech taggers with inter-annotator agreement loss. In *Proc. EACL*, Gothenburg.

Plutchik, R. (2001). The nature of emotions: Human emotions have deep evolutionary roots, a fact that may explain their complexity and provide tools for clinical practice. *American Scientist*, 89(4):344–350.

Plutchik, R. (2003). *Emotions and life: Perspectives from psychology, biology, and evolution.* American Psychological Association.

Poria, S., Cambria, E., Bajpai, R., and Hussain, A. (2017). A review of affective computing: From unimodal analysis to multimodal fusion. *Information Fusion*, 37:98–125.

Poria, S., Hazarika, D., Majumder, N., Naik, G., Cambria, E., and Mihalcea, R. (2019). MELD: A multimodal multi-party dataset for emotion recognition in conversations. In *Proc. ACL*, Florence.

Poria, S., Majumder, N., Hazarika, D., Cambria, E., Gelbukh, A., and Hussain, A. (2018). Multimodal sentiment analysis: Addressing key issues and setting up the baselines. *IEEE Intelligent Systems*, 33(6):17–25.

Povolny, F., Matejka, P., Hradis, M., Popková, A., Otrusina, L., Smrz, P., Wood, I., Robin, C., and Lamel, L. (2016). Multimodal emotion recognition for AVEC 2016 challenge. In *Proc. ACM-MM*, Amsterdam.

Prabhakaran, V., Bloodgood, M., Diab, M., Dorr, B., Levin, L., Piatko, C. D., Rambow, O., and Van Durme, B. (2012). Statistical modality tagging from rule-based annotations and crowdsourcing. In *Proc. ExProM Workshop*, Jeju.

Pulido, M. L. B., Hernández, J. B. A., Ballester, M. Á. F., González, C. M. T., Mekyska, J., and Smékal, Z. (2020). Alzheimer's disease and automatic speech analysis: A review. *Expert systems with applications*, 150:113213.

Pushpa, C., Priya, M., et al. (2016). A review on deep learning algorithms for speech and facial emotion recognition. *International Journal of Control Theory & Applications*, 9(24):183–204.

Qian, Y., Zhang, Y., Ma, X., Yu, H., and Peng, L. (2019). EARS: Emotion-aware recommender system based on hybrid information fusion. *Information Fusion*, 46:141–146.

Qiao, Y., Yin, X., Wiechmann, D., and Kerz, E. (2021). Alzheimer's disease detection from spontaneous speech through combining linguistic complexity and (dis) fluency features with pretrained language models. In *Proc. Interspeech*, Brno.

Qin, Y., Liu, W., Peng, Z., Ng, S.-I., Li, J., Hu, H., and Lee, T. (2021). Exploiting pre-trained ASR models for Alzheimer's disease recognition through spontaneous speech. In *Proc. NCMMSC*, Xuzhou.

Radford, A., Kim, J. W., Xu, T., Brockman, G., McLeavey, C., and Sutskever, I. (2023). Robust speech recognition via large-scale weak supervision. In *Proc. ICML*, Hawaii.

Rafailov, R., Sharma, A., Mitchell, E., Manning, C. D., Ermon, S., and Finn, C. (2023). Direct preference optimization: Your language model is secretly a reward model. In *Proc. NeurIPS*, New Oleans.

Rajasekhar, A. and Hota, M. K. (2018). A study of speech, speaker and emotion recognition using mel frequency cepstrum coefficients and support vector machines. In *Proc. ICCSP*, Chennai.

Ramachandran, P., Zoph, B., and Le, Q. V. (2017). Searching for activation functions. In *Proc. ICLR*, Vancouver.

Ramesh, D. and Sanampudi, S. K. (2022). An automated essay scoring systems: A systematic literature review. *Artificial Intelligence Review*, 55(3):2495–2527.

Ratner, A. J., De Sa, C. M., Wu, S., Selsam, D., and Ré, C. (2016). Data programming: Creating large training sets, quickly. In *Proc. NeurIPS*, Barcelona.

Ravi, V., Wang, J., Flint, J., and Alwan, A. (2022). A Step Towards Preserving Speakers' Identity While Detecting Depression Via Speaker Disentanglement. In *Proc. Interspeech*, Incheon.

Ray, A., Kumar, S., Reddy, R., Mukherjee, P., and Garg, R. (2019). Multi-level attention network using text, audio and video for depression prediction. In *Proc. ACM-MM*, Nice.

Revina, I. M. and Emmanuel, W. S. (2021). A survey on human face expression recognition techniques. *Journal of King Saud University-Computer and Information Sciences*, 33(6):619–628.

Ringeval, F., Schuller, B., Valstar, M., Cowie, R., and Pantic, M. (2015). AVEC 2015: The 5th international audio/visual emotion challenge and workshop. In *Proc. ACM-MM*, Brisbane.

Ringeval, F., Schuller, B., Valstar, M., Gratch, J., Cowie, R., Scherer, S., Mozgai, S., Cummins, N., Schmitt, M., and Pantic, M. (2017). AVEC 2017: Real-life depression, and affect recognition workshop and challenge. In *Proc. ACM-MM*, Mountain View.

Ringeval, F., Sonderegger, A., Sauer, J., and Lalanne, D. (2013). Introducing the RECOLA multimodal corpus of remote collaborative and affective interactions. In *Proc. FG*, Shanghai.

Ristea, N.-C. and Ionescu, R. T. (2021). Self-paced ensemble learning for speech and audio classification. In *Proc. Interspeech*, Brno.

Ritchie, K., Carriere, I., Su, L., O'Brien, J. T., Lovestone, S., Wells, K., and Ritchie, C. W. (2017). The midlife cognitive profiles of adults at high risk of late-onset Alzheimer's disease: The prevent study. *Alzheimer's & Dementia*, 13(10):1089–1097.

Robbins, H. and Monro, S. (1951). A stochastic approximation method. *The Annals of Mathematical Statistics*, pages 400–407.

Rohanian, M., Hough, J., and Purver, M. (2021a). Alzheimer's dementia recognition using acoustic, lexical, disfluency and speech pause features robust to noisy inputs. *arXiv preprint arXiv:2106.15684*.

Rohanian, M., Hough, J., and Purver, M. (2021b). Multi-modal fusion with gating using audio, lexical and disfluency features for alzheimer's dementia recognition from spontaneous speech. *arXiv preprint arXiv:2106.09668*.

Rong, J., Chen, Y.-P. P., Chowdhury, M., and Li, G. (2007). Acoustic features extraction for emotion recognition. In *Proc. ICIS*, Montreal.

Ross, G. W., Cummings, J. L., and Benson, D. F. (1990). Speech and language alterations in dementia syndromes: Characteristics and treatment. *Aphasiology*, 4(4):339–352.

Ruiz, N., Schulter, S., and Chandraker, M. (2019). Learning to simulate. In *Proc. ICLR*, New Orleans.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, 323(6088):533–536.

Russell, J. A. (1979). Affective space is bipolar. *Journal of personality and social psychology*, 37(3):345.

Russell, J. A. (1980). A circumplex model of affect. *Journal of Personality and Social Psychology*, 39(6):1161.

Russell, J. A. and Mehrabian, A. (1977). Evidence for a three-factor theory of emotions. *Journal of Research in Personality*, 11(3):273–294.

Saha, S., Datta, S., Konar, A., and Janarthanan, R. (2014). A study on emotion recognition from body gestures using kinect sensor. In *Proc. ICICSP*, Taichung.

Sahu, S., Mitra, V., Seneviratne, N., and Espy-Wilson, C. Y. (2019). Multi-modal learning for speech emotion recognition: An analysis and comparison of ASR outputs with ground truth transcription. In *Proc. Interspeech*, Graz.

Sanborn, A. and Griffiths, T. (2007). Markov chain Monte Carlo with people. In *Proc. NeurIPS*, Vancouver.

Sapiński, T., Kamińska, D., Pelikant, A., and Anbarjafari, G. (2019). Emotion recognition from skeletal movements. *Entropy*, 21(7):646.

Sarawgi, U., Zulfikar, W., Soliman, N., and Maes, P. (2020). Multimodal inductive transfer learning for detection of Alzheimer's dementia and its severity. *arXiv preprint arXiv:2009.00700*.

Sarraf, S. and Tofighi, G. (2016). Deep learning-based pipeline to recognize Alzheimer's disease using fMRI data. In *Future technologies conference*, San Francisco.

Schatzmann, J., Thomson, B., Weilhammer, K., Ye, H., and Young, S. (2007). Agenda-based user simulation for bootstrapping a POMDP dialogue system. In *Proc. NAACL*, Vancouver.

Scherer, S., Stratou, G., Mahmoud, M., Boberg, J., Gratch, J., Rizzo, A., and Morency, L.-P. (2013). Automatic behavior descriptors for psychological disorder analysis. In *Proc. FG*, Shanghai.

Schneider, S., Baevski, A., Collobert, R., and Auli, M. (2019). Wav2Vec: Unsupervised pre-training for speech recognition. In *Proc. Interspeech*, Graz.

Schuster, M. and Paliwal, K. (1997). Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45:2673 – 2681.

Sebe, N., Cohen, I., and Huang, T. S. (2005). Multimodal emotion recognition. In *Handbook of pattern recognition and computer vision*, pages 387–409. World Scientific.

Seehapoch, T. and Wongthanavasu, S. (2013). Speech emotion recognition using support vector machines. In *Proc. KST*, Chonburi.

Sensoy, M., Kaplan, L., and Kandemir, M. (2018). Evidential deep learning to quantify classification uncertainty. In *Proc. NeurIPS*, Montréal.

Seshadrinathan, K., Soundararajan, R., Bovik, A. C., and Cormack, L. K. (2010). Study of subjective and objective quality assessment of video. *IEEE Transactions on Image Processing*, 19(6):1427–1441.

Shen, P., Changjun, Z., and Chen, X. (2011). Automatic speech emotion recognition using support vector machine. In *Proc. ICMEE*, Hefei.

Shen, Y., Yang, H., and Lin, L. (2022). Automatic depression detection: An emotional audio-textual corpus and a GRU/BiLSTM-based model. In *Proc. ICASSP*, Singapore.

Shi, W., Qian, K., Wang, X., and Yu, Z. (2019). How to build user simulators to train RL-based dialog systems. In *Proc. EMNLP-IJCNLP*, Hong Kong.

Shirian, A. and Guha, T. (2021). Compact graph architecture for speech emotion recognition. In *Proc. ICASSP*, Toronto.

Shrout, P. E. and Fleiss, J. L. (1979). Intraclass correlations: Uses in assessing rater reliability. *Psychological bulletin*, 86(2):420.

Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Siu, M.-h., Gish, H., and Richardson, F. (1997). Improved estimation, evaluation and applications of confidence measures for speech recognition. In *Proc. Eurospeech*, Rhodes.

Snell, J., Swersky, K., and Zemel, R. (2017). Prototypical networks for few-shot learning. In *Proc. NeurIPS*, Long Beach.

Snyder, D., Garcia-Romero, D., Sell, G., Povey, D., and Khudanpur, S. (2018). X-vectors: Robust DNN embeddings for speaker recognition. In *Proc. ICASSP*, Hyderabad.

Sobin, C. and Sackeim, H. A. (1997). Psychomotor symptoms of depression. *American Journal of Psychiatry*, 154(1):4–17.

Spezialetti, M., Placidi, G., and Rossi, S. (2020). Emotion recognition for human-robot interaction: Recent advances and future perspectives. *Frontiers in Robotics and AI*, 7:532279.

Sridhar, K. and Busso, C. (2020a). Ensemble of students taught by probabilistic teachers to improve speech emotion recognition. In *Proc. Interspeech*, Shanghai.

Sridhar, K. and Busso, C. (2020b). Modeling uncertainty in predicting emotional attributes from spontaneous speech. In *Proc. ICASSP*, Barcelona.

Sridhar, K., Lin, W.-C., and Busso, C. (2021). Generative approach using soft-labels to learn uncertainty in predicting emotional attributes. In *Proc. ACII*, Chicago.

Srinivasan, S., Huang, Z., and Kirchhoff, K. (2022). Representation learning through cross-modal conditional teacher-student training for speech emotion recognition. In *Proc. ICASSP*, Singapore.

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

Sun, B., Zhang, Y., He, J., Yu, L., Xu, Q., Li, D., and Wang, Z. (2017). A random forest regression method with selected-text feature for depression assessment. In *Proc. ACM-MM*, Mountain View.

Sun, G., Zhang, C., and Woodland, P. C. (2021). Combination of deep speaker embeddings for diarisation. *Neural Networks*, 141:372–384.

Sun, Y., Wen, G., and Wang, J. (2015). Weighted spectral features based on local hu moments for speech emotion recognition. *Biomedical signal processing and control*, 18:80–90.

Syed, M. S. S., Syed, Z. S., Lech, M., and Pirogova, E. (2020). Automated screening for Alzheimer's dementia through spontaneous speech. In *Proc. Interspeech*, Shanghai.

Syed, Z. S., Syed, M. S. S., Lech, M., and Pirogova, E. (2021). Automated recognition of Alzheimer's dementia using bag-of-deep-features and model ensembling. *IEEE Access*, 9:88377–88390.

Szatloczki, G., Hoffmann, I., Vincze, V., Kalman, J., and Pakaski, M. (2015). Speaking in Alzheimer's disease, is that an early sign? Importance of changes in language abilities in Alzheimer's disease. *Frontiers in Aging Neuroscience*, 7:195.

Taboada, M., Brooke, J., Tofiloski, M., Voll, K., and Stede, M. (2011). Lexicon-based methods for sentiment analysis. *Computational linguistics*, 37(2):267–307.

Talebi, H. and Milanfar, P. (2018). NIMA: Neural image assessment. *IEEE Transactions on Image Processing*, 27(8):3998–4011.

Tang, T. B., Chong, J. S., Kiguchi, M., Funane, T., and Lu, C.-K. (2021). Detection of emotional sensitivity using fNIRS based dynamic functional connectivity. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 29:894–904.

Termenon, M., Grana, M., Besga, A., Echeveste, J., and Gonzalez-Pinto, A. (2013). Lattice independent component analysis feature selection on diffusion weighted imaging for Alzheimer's disease classification. *Neurocomputing*, 114:132–141.

Tracy, J. L. and Randles, D. (2011). Four models of basic emotions: A review of ekman and cordaro, izard, levenson, and panksepp and watt. *Emotion review*, 3(4):397–405.

Triantafyllopoulos, A., Wagner, J., Wierstorf, H., Schmitt, M., Reichel, U., Eyben, F., Burkhardt, F., and Schuller, B. W. (2022). Probing speech emotion recognition transformers for linguistic knowledge. In *Proc. Interspeech*, Incheon.

Trigeorgis, G., Ringeval, F., Brueckner, R., Marchi, E., Nicolaou, M. A., Schuller, B., and Zafeiriou, S. (2016). Adieu features? End-to-end speech emotion recognition using a deep convolutional recurrent network. In *Proc. ICASSP*, Shanghai.

Tripathi, S., Tripathi, S., and Beigi, H. (2018). Multi-modal emotion recognition on IEMOCAP dataset using deep learning. *arXiv preprint 1804.05788*.

Tur, G., Hakkani-Tür, D., and Schapire, R. E. (2005). Combining active and semi-supervised learning for spoken language understanding. *Speech Communication*, 45(2):171–186.

Tzirakis, P., Trigeorgis, G., Nicolaou, M., Schuller, B., and Zafeiriou, S. (2017). End-to-end multimodal emotion recognition using deep neural networks. *IEEE Journal of Selected Topics in Signal Processing*, 11(8):1301–1309.

Tzirakis, P., Zhang, J., and Schuller, B. W. (2018). End-to-end speech emotion recognition using deep neural networks. In *Proc. ICASSP*, Calgary.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). Attention is all you need. In *Proc. NeurIPS*, Long Beach.

Velayudhan, L., Ryu, S.-H., Raczek, M., Philpot, M., Lindesay, J., Critchfield, M., and Livingston, G. (2014). Review of brief cognitive tests for patients with suspected dementia. *International Psychogeriatrics*, 26(8):1247–1262.

Vinyals, O., Blundell, C., Lillicrap, T., Wierstra, D., et al. (2016). Matching networks for one shot learning. In *Proc. NeurIPS*, Barcelona.

Wainwright, M. J., Jordan, M. I., et al. (2008). Graphical models, exponential families, and variational inference. *Foundations and Trends in Machine Learning*, 1(1–2):1–305.

Wang, J., Xue, M., Culhane, R., Diao, E., Ding, J., and Tarokh, V. (2020). Speech emotion recognition with dual-sequence LSTM architecture. In *Proc. ICASSP*, Barcelona.

Wang, Q., Li, B., Xiao, T., Zhu, J., Li, C., Wong, D. F., and Chao, L. S. (2019). Learning deep transformer models for machine translation. In *Proc. ACL*, Florence.

Wang, X., Wang, M., Qi, W., Su, W., Wang, X., and Zhou, H. (2021). A novel end-to-end speech emotion recognition network with stacked transformer layers. In *Proc. ICASSP*, Toronto.

Watanabe, S., Mandel, M., Barker, J., Vincent, E., Arora, A., Chang, X., Khudanpur, S., Manohar, V., Povey, D., Raj, D., and Snyder, D. (2020). CHiME-6 challenge: Tackling multispeaker speech recognition for unsegmented recordings. In *Proc. CHiME*, San Francisco.

Weiner, J., Frankenberg, C., Schröder, J., and Schultz, T. (2019). Speech reveals future risk of developing dementia: Predictive dementia screening from biographic interviews. In *Proc. ASRU*, Sentosa.

Wessel, F., Schluter, R., Macherey, K., and Ney, H. (2001). Confidence measures for large vocabulary continuous speech recognition. *IEEE Transactions on Speech and Audio Processing*, 9(3):288–298.

Wiebe, J., Wilson, T., Bruce, R., Bell, M., and Martin, M. (2004). Learning subjective language. *Computational linguistics*, 30(3):277–308.

Williams, C. K. and Rasmussen, C. E. (2006). *Gaussian processes for machine learning*. MIT press.

Williamson, J. R., Godoy, E., Cha, M., Schwarzentruber, A., Khorrami, P., Gwon, Y., Kung, H.-T., Dagli, C., and Quatieri, T. F. (2016). Detecting depression using vocal, facial and semantic communication cues. In *Proc. ACM-MM*, Amsterdam.

Williamson, J. R., Quatieri, T. F., Helfer, B. S., Horwitz, R., Yu, B., and Mehta, D. D. (2013). Vocal biomarkers of depression based on motor incoordination. In *Proc. ACM-MM*, Barcelona.

Wittenberg, R., Hu, B., Barraza-Araiza, L., and Rehill, A. (2019). Projections of older people with dementia and costs of dementia care in the United Kingdom, 2019–2040. *London School of Economics*.

World Health Organisation (Accessed: March 12, 2024). Depression. *https://www.who.int/news-room/fact-sheets/detail/depression*.

Wu, J., Dang, T., Sethu, V., and Ambikairajah, E. (2022a). A novel sequential Monte Carlo framework for predicting ambiguous emotion states. In *Proc. ICASSP*, Singapore.

Wu, R., Chen, S.-E., Zhang, J., and Chu, X. (2022b). Learning hyper label model for programmatic weak supervision. In *Proc. ICLR*, Conference held virtually.

Wu, W. (2020). Multimodal emotion recognition. MPhil thesis, University of Cambridge.

Wu, W., Chen, W., Zhang, C., and Woodland, P. (2024a). Modelling variability in human annotator simulation. In *Proc. ACL*, Bangkok.

Wu, W., Li, B., Zhang, C., Chiu, C.-C., Li, Q., Bai, J., Sainath, T. N., and Woodland, P. C. (2024b). Handling ambiguity in emotion: From out-of-domain detection to distribution estimation. In *Proc. ACL*, Bangkok.

Wu, W., Wu, M., and Yu, K. (2022c). Climate and weather: Inspecting depression detection via emotion recognition. In *Proc. ICASSP*, Singapore.

Wu, W., Zhang, C., and Woodland, P. C. (2021). Emotion recognition by fusing time synchronous and time asynchronous representations. In *Proc. ICASSP*, Toronto.

Wu, W., Zhang, C., and Woodland, P. C. (2022d). Distribution-based emotion recognition in conversation. In *Proc. SLT*, Doha.

Wu, W., Zhang, C., and Woodland, P. C. (2023a). Estimating the uncertainty in emotion attributes using deep evidential regression. In *Proc. ACL*, Toronto.

Wu, W., Zhang, C., and Woodland, P. C. (2023b). Integrating emotion recognition with speech recognition and speaker diarisation for conversations. In *Proc. Interspeech*, Dublin.

Wu, W., Zhang, C., and Woodland, P. C. (2023c). Self-supervised representations in speech-based depression detection. In *Proc. ICASSP*, Rhodes.

Wu, W., Zhang, C., and Woodland, P. C. (2024c). Confidence estimation for automatic detection of depression and Alzheimer's disease based on clinical interviews. In *Proc. Interspeech*, Kos.

Wu, W., Zhang, C., Wu, X., and Woodland, P. (2022e). Estimating the uncertainty in emotion class labels with utterance-specific Dirichlet priors. *IEEE Transactions on Affective Computing*, 14(4):2810–2822.

Wu, X., Liu, S., Cao, Y., Li, X., Yu, J., Dai, D., Ma, X., Hu, S., Wu, Z., Liu, X., et al. (2019). Speech emotion recognition using capsule networks. In *Proc, ICASSP*, Brighton.

Xu, Y., Xu, H., and Zou, J. (2020). HGFM: A hierarchical grained and feature model for acoustic emotion recognition. In *Proc. ICASSP*, Barcelona.

Yang, L., Jiang, D., He, L., Pei, E., Oveneke, M. C., and Sahli, H. (2016). Decision tree based depression classification from audio video and language information. In *Proc. ACM-MM*, Amsterdam.

Yang, L., Jiang, D., Xia, X., Pei, E., Oveneke, M. C., and Sahli, H. (2017). Multimodal measurement of depression using deep learning models. In *Proc. ACM-MM*, Mountain View.

Yang, S.-W., Chi, P.-H., Chuang, Y.-S., Lai, C.-I. J., Lakhotia, K., Lin, Y. Y., Liu, A. T., Shi, J., Chang, X., Lin, G.-T., Huang, T.-H., Tseng, W.-C., tik Lee, K., Liu, D.-R., Huang, Z., Dong, S., Li, S.-W., Watanabe, S., Mohamed, A., and yi Lee, H. (2021). SUPERB: Speech processing universal performance benchmark. In *Proc. Interspeech*, Brno.

Ye, Z., Hu, S., Li, J., Xie, X., Geng, M., Yu, J., Xu, J., Xue, B., Liu, S., Liu, X., et al. (2021). Development of the CUHK elderly speech recognition system for neurocognitive disorder detection using the DementiaBank corpus. In *Proc. ICASSP*, Toronto.

Yoon, W.-J., Cho, Y.-H., and Park, K.-S. (2007). A study of speech emotion recognition and its application to mobile services. In *Proc. UIC*, Hong Kong.

Yu, B., Quatieri, T. F., Williamson, J. R., and Mundt, J. C. (2015). Cognitive impairment prediction in the elderly based on vocal biomarkers. In *Proc. Interspeech*, Dresden.

Yu, D., Li, J., and Deng, L. (2011). Calibration of confidence measures in speech recognition. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(8):2461–2473.

Yuan, J., Bian, Y., Cai, X., Huang, J., Ye, Z., and Church, K. (2020). Disfluencies and fine-tuning pre-trained language models for detection of Alzheimer's disease. In *Proc. Interspeech*, Shanghai.

Zadeh, A. B., Liang, P. P., Poria, S., Cambria, E., and Morency, L.-P. (2018). Multimodal language analysis in the wild: CMU-MOSEI dataset and interpretable dynamic fusion graph. In *Proc. ACL*, Melbourne.

Zeiler, M. D. (2012). AdaDelta: An adaptive learning rate method. *arXiv preprint arXiv:1212.5701*.

Zeisel, J., Bennett, K., and Fleming, R. (2020). World Alzheimer report 2020: Design, dignity, dementia: Dementia-related design and the built environment.

Zen, M. and Vanderdonckt, J. (2016). Assessing user interface aesthetics based on the inter-subjectivity of judgment. In *Proc. BCS HCI*, Poole.

Zhang, D., Ju, X., Li, J., Li, S., Zhu, Q., and Zhou, G. (2020). Multi-modal multi-label emotion detection with modality and label dependence. In *Proc. EMNLP*, Conference held virtually.

Zhang, H., Mimura, M., Kawahara, T., and Ishizuka, K. (2022a). Selective multi-task learning for speech emotion recognition using corpora of different styles. In *Proc. ICASSP*, Singapore.

Zhang, Q., An, N., Wang, K., Ren, F., and Li, L. (2013). Speech emotion recognition using combination of features. In *Proc. ICICIP*, Beijing.

Zhang, S., Lei, B., Chen, A., Chen, C., and Chen, Y. (2010). Kisomap-based feature extraction for spoken emotion recognition. In *Proc. ICSP*, Beijing.

Zhang, S., Zhang, S., Huang, T., and Gao, W. (2017). Speech emotion recognition using deep convolutional neural network and discriminant temporal pyramid matching. *IEEE Transactions on Multimedia*, 20(6):1576–1590.

Zhang, T., Liu, M., Yuan, T., and Al-Nabhan, N. (2021). Emotion-aware and intelligent internet of medical things toward emotion recognition during covid-19 pandemic. *IEEE Internet of Things Journal*, 8(21):16002–16013.

Zhang, Y., Dong, Z., Phillips, P., Wang, S., Ji, G., Yang, J., and Yuan, T.-F. (2015). Detection of subjects and brain regions related to Alzheimer's disease using 3D MRI scans based on eigenbrain and machine learning. *Frontiers in Computational Neuroscience*, 9:66.

Zhang, Y., Han, W., Qin, J., Wang, Y., Bapna, A., Chen, Z., Chen, N., Li, B., Axelrod, V., Wang, G., et al. (2023). Google USM: Scaling automatic speech recognition beyond 100 languages. *arXiv preprint arXiv:2303.01037*.

Zhang, Y., Park, D. S., Han, W., Qin, J., Gulati, A., Shor, J., Jansen, A., Xu, Y., Huang, Y., Wang, S., et al. (2022b). BigSSL: Exploring the frontier of large-scale semi-supervised learning for automatic speech recognition. *IEEE Journal of Selected Topics in Signal Processing*, 16(6):1519–1532.

Zhao, Z., Wang, Y., and Wang, Y. (2022). Multi-level Fusion of Wav2vec 2.0 and BERT for Multimodal Emotion Recognition. In *Proc. Interspeech*, Incheon.

Zheng, X., Zhang, C., and Woodland, P. (2022). Tandem multitask training of speaker diarisation and speech recognition for meeting transcription. In *Proc. Interspeech*, Incheon.

Zhou and Chellappa (1988). Computation of optical flow using a neural network. In *Proc. ICNN*, San Diego.

Zhu, Y., Kiros, R., Zemel, R., Salakhutdinov, R., Urtasun, R., Torralba, A., and Fidler, S. (2015). Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proc. ICCV*, Santiago.

Zhu, Y., Obyat, A., Liang, X., Batsis, J. A., and Roth, R. M. (2021). WavBERT: Exploiting semantic and non-semantic speech using Wav2vec and BERT for dementia detection. In *Proc. Interspeech*, Brno.

Zhu-Zhou, F., Gil-Pita, R., García-Gómez, J., and Rosa-Zurera, M. (2022). Robust multi-scenario speech-based emotion recognition system. *Sensors*, 22(6):2343.

Zong, Y., Zheng, W., Zhang, T., and Huang, X. (2016). Cross-corpus speech emotion recognition based on domain-adaptive least-squares regression. *IEEE signal processing letters*, 23(5):585–589.

Appendix A

Published Papers Related to the Thesis

This chapter provides detail of published papers related to the thesis, which includes papers directly covered by the thesis (Appendix A.1) and papers related to but not included in the thesis (Appendix A.2)[1].

A.1 Papers Included in the Thesis

This section describes the papers published during the PhD study which have been covered by the thesis. The list of published papers directly included in the thesis is listed in Table A.1 along with the corresponding chapters related to these publications.

Wu et al. (2023b) **Wu, W.**, Zhang, C., and Woodland, P. C. (2023). Integrating emotion recognition with speech recognition and speaker diarisation for conversations. In *Proceedings of Interspeech 2023*.

This paper proposes a system that integrates emotion recognition with speech recognition and speaker diarisation in a jointly-trained model. The system investigates emotion recognition with automatic segmentation to address the issue of lacking manual segmentation in practical applications. The system also improves recognition performance on emotional speech with a 12% reduction in relative word error rate with automatic segmentation. The time-weighted emotion error rate and the speaker-attributed time-weighted emotion error rate were proposed to evaluate emotion classification performance when segmentation is non-oracle.

[1] Only (co-)first author publications are considered.

Table A.1 List of published papers included in the thesis.

Paper Title	Reference	Related Chapter
Integrating emotion recognition with speech recognition and speaker diarisation for conversations	Wu et al. (2023b)	Chapter 4
Handling ambiguity in emotion: from out-of-domain detection to distribution estimation	Wu et al. (2024b)	Chapter 5
Estimating the uncertainty in emotion attributes using deep evidential regression	Wu et al. (2023a)	Chapter 6
Modelling variability in human annotator simulation	Wu et al. (2024a)	Chapter 7
Self-supervised representations in speech-based depression detection	Wu et al. (2023c)	Chapter 8
Confidence estimation for automatic detection of depression and Alzheimer's disease based on clinical interviews	Wu et al. (2024c)	Chapter 8

Wu et al. (2024b) **Wu, W.**, Li, B., Zhang, C., Chiu, C.-C., Li, Q., Bai, J., Sainath, T. N., and Woodland, P. C. (2024). Handling ambiguity in emotion: From out-of-domain detection to distribution estimation. In *Proceedings of the 62st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (ACL 2024).

This work re-examines the emotion classification problem, starting with an exploration of ways to handle data with ambiguous emotions. It is first shown that incorporating ambiguous emotions as an extra class reduces the classification performance of the original emotion classes. Then, evidence theory is adopted to quantify uncertainty in emotion classification which allows the classifier to output "I don't know" when it encounters utterances with ambiguous emotion. The model is trained to predict the hyperparameters of a Dirichlet distribution, which models the second-order probability of the probability assignment over emotion classes. This is the first work that treats ambiguous emotion as OOD and detects it by uncertainty estimation, and is also the first work that applies EDL to quantify uncertainty in emotion classification. Furthermore, to capture finer-grained emotion differences, the emotion classification problem is transformed into an emotion distribution estimation problem. All annotations are taken into account rather than only the majority opinion. A novel approach is proposed which extends standard EDL to quantify uncertainty in emotion distribution estimation. Experimental results show that given an utterance with ambiguous emotion the

proposed approach is able to provide a comprehensive representation of its emotion content as a distribution with a reliable uncertainty measure.

Wu et al. (2023a) **Wu, W.**, Zhang, C., and Woodland, P. C. (2023). Estimating the uncertainty in emotion attributes using deep evidential regression. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (ACL 2023).

This paper proposes deep evidential emotion regression (DEER) for estimating aleatoric and epistemic uncertainty in emotion attributes. Treating observed attribute-based annotations as samples drawn from a Gaussian distribution, DEER places a normal-inverse gamma (NIG) prior over the Gaussian likelihood. A novel training loss is proposed which combines a per-observation-based negative log likelihood loss with a regulariser on both the mean and the variance of the Gaussian likelihood. Experiments on both the MSP-Podcast and IEMOCAP datasets show that DEER can produce SOTA results in estimating both the mean value and the distribution of emotion attributes. The use of NIG, the conjugate prior to the Gaussian distribution, leads to tractable analytic computation of the marginal likelihood as well as aleatoric and epistemic uncertainty associated with attribute prediction. Uncertainty estimation is analysed by visualisation and a reject option.

Wu et al. (2024a) **Wu, W.**, Chen, W., Zhang, C., and Woodland, P. (2024). Modelling variability in human annotator simulation. In *Findings of the Association for Computational Linguistics: ACL 2024*.

This paper studies human annotator simulation (HAS), a cost-effective alternative to generating human-like annotations for automatic data labelling and model evaluation. A novel framework is proposed to incorporate the variability of human evaluations into HAS. This framework leverages diverse annotations to estimate the distribution of categorical human annotations by meta-learning a conditional softmax flow (S-CNF) on large crowd-sourced datasets. This overcomes the drawbacks of prior work and enables efficient generation of annotations that exhibit human-like variability for unlabelled test inputs. The proposed method clearly and consistently outperformed a wide range of methods on emotion class labelling and toxic speech detection, achieving the best performance for matching human annotation distributions and inter-annotator disagreement simulation. Apart from the S-CNF proposed in Wu et al. (2024a), conditional integer flow (I-SNF) is introduced in Chapter 7 to model ordinal annotations.

Wu et al. (2023c) **Wu, W.**, Zhang, C., and Woodland, P. C. (2023). Self-supervised representations in speech-based depression detection. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing* (ICASSP 2023).

This paper studies the use of SSL representations in speech-based depression detection (SDD). Block-wise analysis of the foundation models implies that word meaning information is helpful in SDD. Finetuning pretrained speech foundation models for AER improves SDD performance, indicating that some indicators are shared between AER and SDD. SDD performance when using ASR transcriptions matches that of using reference transcriptions when combined with the hidden representations derived from an ASR-fine-tuned foundation model. The ensemble of speech and text foundation models produced the SOTA F1 score of 0.89 on DAIC-WOZ dataset without using the reference transcriptions.

Wu et al. (2024c) **Wu, W.**, Zhang, C., and Woodland, P. C. (2024). Confidence estimation for automatic detection of depression and Alzheimer's disease based on clinical interviews. In *Proceedings of Interspeech 2024*.

This paper investigates confidence estimation of automatic detection of Alzheimer's disease and depression. A novel Bayesian approach is proposed which places a dynamic Dirichlet prior over the categorical likelihood to model the second-order uncertainty of model prediction. A range of metrics are adopted to evaluate the performance for detection accuracy and confidence estimation. Experiments were conducted on the AD dataset ADReSS and depression dataset DAIC-WOZ. Results show that the proposed method clearly and consistently outperforms a range of baselines in terms of both classification accuracy and confidence estimation for both Alzheimer's disease detection and depression detection.

A.2 Papers Related to the Thesis

This section presents several additional published papers, the background of which are related to the thesis. This includes earlier published papers on AER[2], works initiated prior to the commencement of the PhD programme, and a paper on AD detection where W. W. is the co-first author[3]. The details are listed in Table A.2.

[2]The methods of these earlier works are now out-dated and have been replaced by improved approaches in later publications.

[3]This work would be included in the other co-first author's thesis.

Table A.2 List of additional published papers related to the thesis.

Paper Title	Reference
Emotion recognition by fusing time synchronous and time asynchronous representations	Wu et al. (2021)
Estimating the uncertainty in emotion class labels with utterance-specific Dirichlet priors	Wu et al. (2022e)
Distribution-based emotion recognition in conversation	Wu et al. (2022d)
Climate and weather: Inspecting depression detection via emotion recognition	Wu et al. (2022c)
Transferring speech-generic and depression-specific knowledge for Alzheimer's disease detection	Cui et al. (2023)

Wu et al. (2021) **Wu, W.**, Zhang, C., and Woodland, P. C. (2021). Emotion recognition by fusing time synchronous and time asynchronous representations. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing* (ICASSP 2021).

This paper proposes a novel structure for AER which consists of a time synchronous branch that aligns MFB and word embeddings to capture the correlations between each word and its acoustic realisations at each time step, and a time asynchronous branch integrates the BERT sentence embeddings from context utterances. The novel two-branch structure achieved SOTA on IEMOCAP dataset[4] and the use of automatic transcriptions has also been investigated. This paper is primarily based on work conducted during W. W.'s MPhil study and has already been included in W. W.'s MPhil thesis (Wu, 2020).

Wu et al. (2022e) **Wu, W.**, Zhang, C., Wu, X., and Woodland, P. (2022). Estimating the uncertainty in emotion class labels with utterance-specific Dirichlet priors. in *IEEE Transactions on Affective Computing*, vol. 14, no. 4, pp. 2810-2822, 1 Oct.-Dec. 2023 (Early access since Nov. 2022).

This work builds on the structure of Wu et al. (2021) and starts to address the disagreement in emotion class annotations by a Dirichlet prior network (DPN)[5]. The EDL* method introduced in Chapter 5 is an improved version of this work. Both methods involve a Dirichlet distribution, but the improvement lies in the problem formulation.

[4]It achieved SOTA on the IEMOCAP dataset at the time of publication but has now been surpassed by foundation models. The method is now out-dated, replaced by foundation-model-based structures.

[5]The EDL* method introduced in Chapter 5 and the approach proposed in Wu et al. (2022e) both learn a Dirichlet prior. In a broad sense, both can be considered as Dirichlet prior networks. However, for the sake of distinction here, the approach proposed in Wu et al. (2022e) is called DPN and the method introduced in Chapter 5 is called EDL*.

In DPN, individual emotion class labels provided by human annotators are treated as one-hot categorical distributions ($\{\boldsymbol{\pi}_m\}_{m=1}^M$) and the model is trained by maximising the likelihood of sampling those one-hot categorical distributions given the Dirichlet prior: $\mathcal{L}_{\text{DPN}}^{\text{NLL}} = \log \text{p}(\{\boldsymbol{\pi}_m\}_{m=1}^M|\boldsymbol{\alpha})$. However, the EDL* method preserves the target emotion labels as discrete class labels ($\{\boldsymbol{y}_m\}_{m=1}^M$) which are drawn from an unknown categorical likelihood ($\boldsymbol{\pi}$). The model is then trained by maximising the likelihood of sampling discrete labels given the Dirichlet prior by marginalising out all possible categorical distributions: $\mathcal{L}_{\text{EDL*}}^{\text{NLL}} = \log \text{P}(\{\boldsymbol{y}_m\}_{m=1}^M|\boldsymbol{\alpha}) = \log \int \text{P}(\{\boldsymbol{y}_m\}_{m=1}^M|\boldsymbol{\pi})\text{p}(\boldsymbol{\pi}|\boldsymbol{\alpha})\text{d}\boldsymbol{\pi}$. EDL* improves over DPN in two aspects: (i) training stability and (ii) preserving or even boosting classification accuracy, whereas DPN greatly reduces it.

Wu et al. (2022d) **Wu, W.**, Zhang, C., and Woodland, P. C. (2022d). Distribution-based emotion recognition in conversation. *In Proceedings of IEEE Spoken Language Technology Workshop* (SLT 2022).

This paper is a follow-up work for Wu et al. (2022e) which applies DPN to emotion recognition in conversation where the emotion state of an utterance is represented by a categorical distribution which depends on the context information and emotion states of the previous utterances in the dialogue. Both the reference transcriptions and the reference segmentation were used in this work.

Wu et al. (2022c) **Wu, W.**, Wu, M., and Yu, K. (2022). Climate and weather: Inspecting depression detection via emotion recognition. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing* (ICASSP 2022).

This paper also builds on the structure of Wu et al. (2021). It investigates the knowledge transfer from emotion recognition to depression detection. A robust depression detection method is derived from utilising emotion features extracted from the pretrained emotion model, which achieved an F1 score of 0.87 on DAIC-WOZ development set while the reference transcriptions were used. It also reveals that depression data shows diverse emotional content and emotion expressed through audio and text modalities are sometimes inconsistent, which provides clues for understanding the relationship between emotion and depression, in particular how healthy/depressed subjects express their emotions. As stated in footnote 4, the backbone method is now out-dated, replaced by foundation-model-based approaches (*i.e.*, Wu et al. (2023c)). This work is instead compared as a baseline in Table 8.7.

Cui et al. (2023) Cui, Z.*, **Wu, W.***[6], Zhang, W.-Q., Wu, J., and Zhang, C. (2023). Transferring speech-generic and depression-specific knowledge for Alzheimer's disease detection. In *Proceedings of IEEE Automatic Speech Recognition and Understanding Workshop* (ASRU 2023).

This paper investigates speech-generic and depression-specific knowledge transfer for Alzheimer's disease detection. It applies the similar approach in Section 8.4 to investigate the use of speech-generic knowledge based on a block-wise analysis with several speech foundation models, which founds the importance of phonetic information and word-level information in AD diagnosis. An end-to-end system structure is then proposed for simultaneous detection of both AD and depression. The system contains a shared encoder which captures the cross-domain information between the AD and depression detection tasks and separate processing streams for each task to retain their domain-specific knowledge. Improvements are observed for both tasks that imply the connection between AD and depression.

[6]W. W. is a co-first author of this work.

Appendix B

Datasets Used in the Thesis

The datasets used in this thesis is listed in Table B.1. The datasets that have not been described in detail in the main text are introduced below.

Table B.1 List of datasets used in the thesis.

Dataset	Reference	Cross Reference	Related Chapter
IEMOCAP	Busso et al. (2008)	Section 3.3.2	Chapter 4, 5
CREMA-D	Cao et al. (2014)	Section 3.3.2	Chapter 5
MSP-Podcast	Lotfian and Busso (2019)	Section 3.3.2	Chapter 6, 7
HateXplain	Mathew et al. (2021)	Section 7.3.2	Chapter 7
SOMOS	Maniati et al. (2022)	Section 7.3.3	Chapter 7
DAIC-WOZ	DeVault et al. (2014)	Section 8.2.1	Chapter 8
ADReSS	Luz et al. (2020)	Section 8.2.2	Chapter 8
LibriSpeech	Panayotov et al. (2015)	Appendix B.1	Chapter 4
Voxceleb 1	Nagrani et al. (2017)	Appendix B.2	Chapter 4
AMI Meeting	Carletta et al. (2005)	Appendix B.3	Chapter 4
LJ Speech	Ito and Johnson (2017)	Appendix B.4	Chapter 7

B.1 LibriSpeech

The LibriSpeech corpus (Panayotov et al., 2015) is a collection of approximately 1,000 hours of English audiobooks recordings from around 2.5k speakers. It is a widely used dataset for ASR. The corpus is split into train, development (dev), and test sets. The full training set is split into 3 partitions of "train-clean-100", "train-clean-360", and "train-other-500" subsets where he number in each set indicates the number of hours of recordings. The set named "clean" contains only selected low-error speech that can be assumed to be clean while sets

with “other” may contain more challenging audio samples. The ‘train-clean-100’ training set is often used on its own to either investigate low-data scenarios, or for quick experiments to determine model-related hyper-parameters before running experiments at a larger scale. The dev and test sets are also split into “clean” and “other” categories, resulting in four subsets “dev-clean”, “dev-other”, “test-clean”, and “test-other”. Each of the dev and test sets is around 5 hours of audio length. The corpus contains around 290k utterances and is designed to be roughly balanced in terms of gender and per-speaker duration. The average duration of an utterance is 12 s.

B.2 Voxceleb 1

The VoxCeleb 1 dataset (Nagrani et al., 2017) contains 100k+ utterances for 1.2k celebrities, extracted from videos uploaded to YouTube. It is widely used for speaker identification tasks. The dataset is roughly gender balanced, with 55% of the speakers male. The speakers span a wide range of different ethnicities, accents, professions and ages. The videos are also shot in various acoustic environments such as on the red carpet, outdoor stadiums, quiet studio interviews, and speeches given to large audiences. The videos are degraded with real world noise, consisting of background chatter, laughter, overlapping speech, room acoustics, and there is a range in the quality of recording equipment and channel noise. The corpus contains about 150k samples in total. The average duration of an utterance is 8.2 s.

B.3 AMI Meeting Corpus

The augmented multi-party interaction (AMI) meeting corpus (Carletta et al., 2005) consists of 100 hours of meeting recordings from 100+ speakers. In each recording, several people discussed technical projects in English. The AMI corpus is widely used for ASR and speaker diarisation. It is split into train, dev and eval sets, where the train set contains 80 hours of audio and the other two contain 10 hours each. Each meeting contains between 10 and 60 minutes of audio data and the average duration of each meeting is 35 minutes.

B.4 LJ Speech

The LJ Speech dataset (Ito and Johnson, 2017) is a public domain speech dataset consisting of 13,100 short audio clips of a single speaker reading passages from 7 non-fiction books. A transcription is provided for each clip. Clips vary in length from 1 to 10 seconds and have

a total length of approximately 24 hours. The mean clip duration is 6.6 s. The texts were published between 1884 and 1964, and are in the public domain. The average number of words per clip is 17.2. The LJ Speech dataset has been used for generating the SOMOS dataset (see Section 7.3.3).

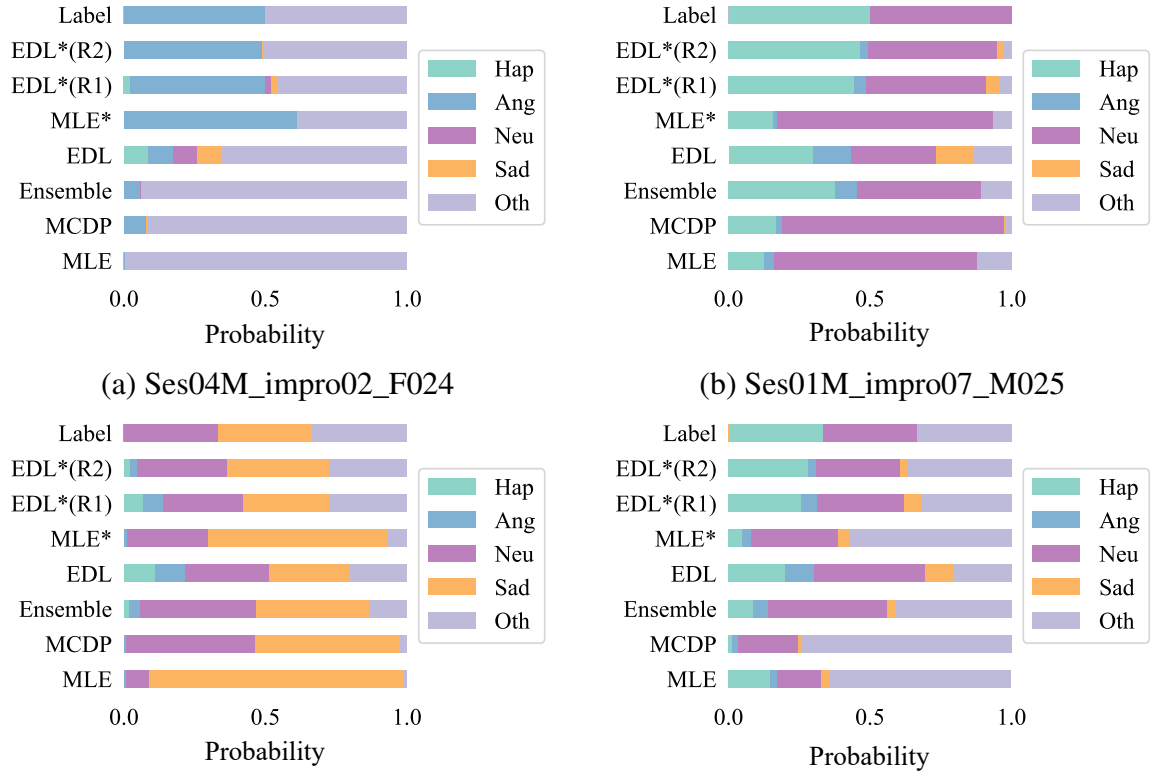
Appendix C

Further Visualised Examples for Section 5.4

This section shows more examples for emotion distribution estimation via evidential deep learning on IEMOCAP (Fig. C.1) and CREMA-D (Fig. C.2). The abbreviation of compared methods are summarised below:

- Label: the "soft label", which corresponds to the relative frequency of occurrence;
- MLE: a deterministic classification network with softmax activation trained by the cross-entropy loss;
- MCDP: a Monte-Carlo dropout model;
- Ensemble: an ensemble of 10 MLE models;
- EDL: the EDL method proposed in Section 5.3;
- MLE*: the "soft label" approach;
- EDL*(R1): the EDL* systems proposed using regularisation terms defined in Eqn. 5.17;
- EDL*(R2): the EDL* systems proposed using regularisation terms defined in Eqn. 5.18.

Aligned with the findings in Section 5.4.5, EDL* methods can better approximate the distribution of emotional content of an utterance.



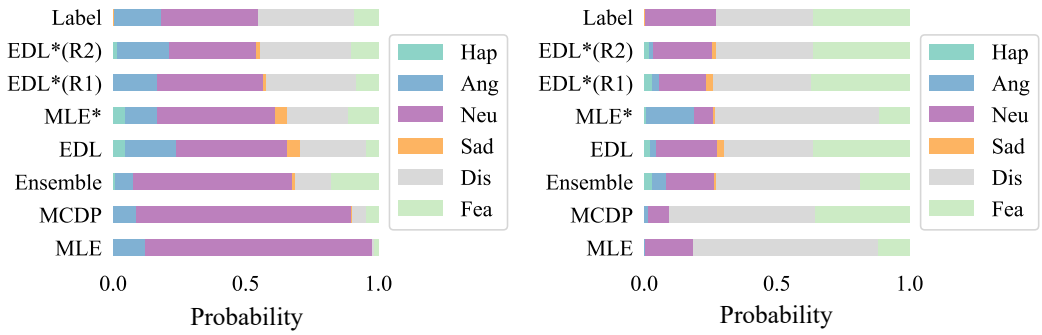
(a) Ses04M_impro02_F024

(b) Ses01M_impro07_M025

(c) Ses04F_script01_1_M033

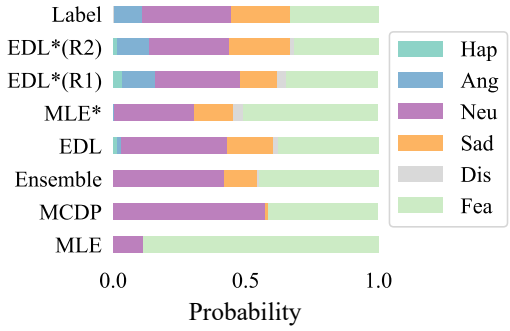
(d) Ses04M_script01_3_M013

Fig. C.1 More visualised examples for emotion distribution estimation via evidential deep learning on IEMOCAP.

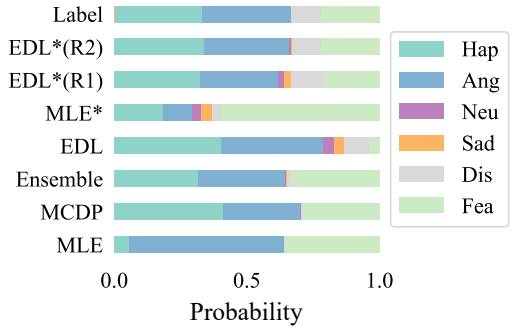


(a) 1084_TSI_ANG_XX

(a) 1033_IWW_DIS_XX



(c) 1068_ITH_SAD_XX



(d) 1009_IWL_FEA_XX

Fig. C.2 More visualised examples for emotion distribution estimation via evidential deep learning on CREMA-D.

Appendix D

Derivations in Chapter 7

Detailed derivations for the training objectives on a single event $\boldsymbol{d}_i = \{\boldsymbol{x}_i, \mathcal{D}_i\}$ where $\mathcal{D}_i = \{\boldsymbol{\eta}_i^{(1)}, \cdots, \boldsymbol{\eta}_i^{(M)}\}$ are presented in this section. For the simplicity of notation, the subscription i in the derivations will be omitted without ambiguity where possible. The meta-learning objectives presented in the paper are obtained by averaging such single-task objectives across tasks.

D.1 Objective Function for the Base CNF and I-CNF

Denote the empirical human annotation distribution as $\mathrm{p}_m(\boldsymbol{y}|\boldsymbol{x}) = \delta\left(\boldsymbol{y} - \boldsymbol{\eta}^{(m)}\right)$, $m = 1, \cdots, M$ and model output distribution as $\mathrm{p}_{\boldsymbol{\theta}}(\boldsymbol{y}|\boldsymbol{x})$. The average KL divergence between them over all M human annotations for this input $\boldsymbol{x}$ is given by:

$$\begin{aligned}
\mathcal{L}(\boldsymbol{\theta}; \boldsymbol{d}) &= \frac{1}{M}\sum_{m=1}^{M} \mathcal{KL}\left(\mathrm{p}_m(\boldsymbol{y}|\boldsymbol{x}) \parallel \mathrm{p}_{\boldsymbol{\theta}}(\boldsymbol{y}|\boldsymbol{x})\right) \\
&= \frac{1}{M}\sum_{m=1}^{M} \int \mathrm{p}_m(\boldsymbol{y}|\boldsymbol{x}) \log \frac{\mathrm{p}_m(\boldsymbol{y}|\boldsymbol{x})}{\mathrm{p}_{\boldsymbol{\theta}}(\boldsymbol{y}|\boldsymbol{x})} \mathrm{d}\boldsymbol{y} \\
&= -\frac{1}{M}\sum_{m=1}^{M} \int \mathrm{p}_m(\boldsymbol{y}|\boldsymbol{x}) \log \mathrm{p}_{\boldsymbol{\theta}}(\boldsymbol{y}|\boldsymbol{x}) \mathrm{d}\boldsymbol{y} + \text{const} \\
&= -\frac{1}{M}\sum_{m=1}^{M} \log \mathrm{p}_{\boldsymbol{\theta}}(\boldsymbol{\eta}^{(m)}|\boldsymbol{x}) + \text{const}
\end{aligned} \tag{D.1}$$

Minimising this KL objective is equivalent to maximising the average log likelihood $\log \mathrm{p}_{\boldsymbol{\theta}}(\boldsymbol{\eta}^{(m)}|\boldsymbol{x})$ over all human annotations as presented in the paper. With numerical approximation, the training objective for I-CNF shares the same formula as that for the base CNF.

D.2 Objective Function of S-CNF

For categorical annotations, each label $\boldsymbol{\eta}^{(m)}$ represents the probabilities of all categories in the categorical human annotation distribution: $\boldsymbol{\eta}^{(m)} = [\boldsymbol{\eta}_1^{(m)}, \cdots, \boldsymbol{\eta}_K^{(m)}]$, where $\boldsymbol{\eta}_k^{(m)} = \mathrm{P}_m(c = k|\boldsymbol{x})$. Denote the model output distribution as $\mathrm{P}_{\boldsymbol{\theta}}(c|\boldsymbol{x})$. The average KL divergence between them over all M human annotations for this input $\boldsymbol{x}$ is given by:

$$\begin{aligned}
\mathcal{L}^{\text{exact}}(\boldsymbol{\theta}; \boldsymbol{d}) &= \frac{1}{M}\sum_{m=1}^{M} \mathcal{KL}\left(\mathrm{P}_m(c|\boldsymbol{x}) \parallel \mathrm{P}_{\boldsymbol{\theta}}(c|\boldsymbol{x})\right) \\
&= \frac{1}{M}\sum_{m=1}^{M}\sum_{k=1}^{K} \mathrm{P}_m(c = k|\boldsymbol{x}) \log \frac{\mathrm{P}_m(c = k|\boldsymbol{x})}{\mathrm{P}_{\boldsymbol{\theta}}(c = k|\boldsymbol{x})} \\
&= -\frac{1}{M}\sum_{m=1}^{M}\sum_{k=1}^{K} \mathrm{P}_m(c = k|\boldsymbol{x}) \log \mathrm{P}_{\boldsymbol{\theta}}(c = k|\boldsymbol{x}) + \text{const} \\
&= -\frac{1}{M}\sum_{m=1}^{M}\sum_{k=1}^{K} \boldsymbol{\eta}_k^{(m)} \log \mathrm{P}_{\boldsymbol{\theta}}(c = k|\boldsymbol{x}) + \text{const},
\end{aligned} \tag{D.2}$$

where the marginal likelihood is lower bounded using variational inference:

$$\begin{aligned}
\log \mathrm{P}_{\boldsymbol{\theta}}(c = k|\boldsymbol{x}) &= \log \int \mathrm{P}(c = k|\boldsymbol{v})\mathrm{p}_{\boldsymbol{\theta}}(\boldsymbol{v}|\boldsymbol{x})\mathrm{d}\boldsymbol{v} \\
&= \log \int \boldsymbol{q}_{\boldsymbol{\Omega}}(\boldsymbol{v}|\boldsymbol{\eta}) \frac{\mathrm{P}(c = k|\boldsymbol{v})\mathrm{p}_{\boldsymbol{\theta}}(\boldsymbol{v}|\boldsymbol{x})}{\boldsymbol{q}_{\boldsymbol{\Omega}}(\boldsymbol{v}|\boldsymbol{\eta})}\mathrm{d}\boldsymbol{v} \\
&\geq \int \boldsymbol{q}_{\boldsymbol{\Omega}}(\boldsymbol{v}|\boldsymbol{\eta}) \log \frac{\mathrm{P}(c = k|\boldsymbol{v})\mathrm{p}_{\boldsymbol{\theta}}(\boldsymbol{v}|\boldsymbol{x})}{\boldsymbol{q}_{\boldsymbol{\Omega}}(\boldsymbol{v}|\boldsymbol{\eta})}\mathrm{d}\boldsymbol{v} \\
&= \mathbb{E}_{\boldsymbol{q}_{\boldsymbol{\Omega}}(\boldsymbol{v}|\boldsymbol{\eta})}\left[\log \mathrm{P}(c = k|\boldsymbol{v}) + \log \mathrm{p}_{\boldsymbol{\theta}}(\boldsymbol{v}|\boldsymbol{x}) - \log \boldsymbol{q}_{\boldsymbol{\Omega}}(\boldsymbol{v}|\boldsymbol{\eta})\right].
\end{aligned} \tag{D.3}$$

Therefore, the final negative ELBO objective is obtained by

$$\begin{aligned}
\mathcal{L}^{\text{exact}} &= -\frac{1}{M}\sum_{m=1}^{M}\sum_{k=1}^{K} \boldsymbol{\eta}_k^{(m)} \log \mathrm{P}_{\boldsymbol{\theta}}(c = k|\boldsymbol{x}) \\
&\leq -\frac{1}{M}\sum_{m=1}^{M}\sum_{k=1}^{K} \boldsymbol{\eta}_k^{(m)} \mathbb{E}_{\boldsymbol{q}_{\boldsymbol{\Omega}}(\boldsymbol{v}|\boldsymbol{\eta}^{(m)})}\left[\log \mathrm{P}(c = k|\boldsymbol{v}) + \log \mathrm{p}_{\boldsymbol{\theta}}(\boldsymbol{v}|\boldsymbol{x}) - \log \boldsymbol{q}_{\boldsymbol{\Omega}}(\boldsymbol{v}|\boldsymbol{\eta}^{(m)})\right] \\
&= -\frac{1}{M}\sum_{m=1}^{M} \mathbb{E}_{\boldsymbol{q}_{\boldsymbol{\Omega}}(\boldsymbol{v}|\boldsymbol{\eta}^{(m)})}\left[\sum_{k=1}^{K} \boldsymbol{\eta}_k^{(m)} \log \mathrm{P}(c = k|\boldsymbol{v}) + \log \mathrm{p}_{\boldsymbol{\theta}}(\boldsymbol{v}|\boldsymbol{x}) - \log \boldsymbol{q}_{\boldsymbol{\Omega}}(\boldsymbol{v}|\boldsymbol{\eta}^{(m)})\right] \\
&= \mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\Omega}; \boldsymbol{d}),
\end{aligned} \tag{D.4}$$

where

$$\log \mathrm{P}(c = k|\boldsymbol{v}) = \mathrm{logsoftmax}(\boldsymbol{v})_k, \tag{D.5}$$

$$\log \mathrm{p}_{\boldsymbol{\theta}}(\boldsymbol{v}|\boldsymbol{x}) = \mathrm{p}_{\boldsymbol{\Lambda}}\left(\boldsymbol{f}_{\boldsymbol{\theta}}^{-1}(\boldsymbol{v})|\boldsymbol{x}\right)\left|\det\left(\frac{\partial \boldsymbol{f}_{\boldsymbol{\theta}}^{-1}(\boldsymbol{v})}{\partial \boldsymbol{v}}\right)\right|, \tag{D.6}$$

$$\log \boldsymbol{q}_{\boldsymbol{\Omega}}(\boldsymbol{v}|\boldsymbol{\eta}^{(m)}) = \mathcal{N}(\boldsymbol{v}|\boldsymbol{\mu}_{\boldsymbol{\Omega}}(\boldsymbol{\eta}^{(m)}), \mathrm{diag}(\boldsymbol{\sigma}^2_{\boldsymbol{\Omega}}(\boldsymbol{\eta}^{(m)}))). \tag{D.7}$$

D.3 The Negative Log Likelihood ($\mathrm{NLL}^{\mathrm{all}}$) for Categorical Annotations

The marginal likelihood of S-CNF is intractable, but can be approximated using Monte Carlo simulation:

$$\begin{aligned}
\mathrm{P}_{\boldsymbol{\theta}}(c = k|\boldsymbol{x}) &= \int \mathrm{P}(c = k|\boldsymbol{v})\mathrm{p}_{\boldsymbol{\theta}}(\boldsymbol{v}|\boldsymbol{x})\mathrm{d}\boldsymbol{v} \\
&= \mathbb{E}_{\mathrm{p}_{\boldsymbol{\theta}}(\boldsymbol{v}|\boldsymbol{x})}\left[\mathrm{P}(c = k|\boldsymbol{v})\right] \\
&\approx \frac{1}{Q}\sum_{j=1}^{Q} \mathrm{P}(c = k|\boldsymbol{v}_j), \quad \{\boldsymbol{v}_j\}_{j=1}^{Q} \sim_{\mathrm{iid}} \mathrm{p}_{\boldsymbol{\theta}}(\boldsymbol{v}|\boldsymbol{x}) \\
&= \frac{1}{Q}\sum_{j=1}^{Q} \mathrm{softmax}(\boldsymbol{v}_j)_k, \quad \{\boldsymbol{v}_j\}_{j=1}^{Q} \sim_{\mathrm{iid}} \mathrm{p}_{\boldsymbol{\theta}}(\boldsymbol{v}|\boldsymbol{x}) \\
&= \bar{\boldsymbol{y}}_k,
\end{aligned} \tag{D.8}$$

where $\bar{\boldsymbol{y}} = \frac{1}{Q}\sum_{j=1}^{Q} \mathrm{softmax}(\boldsymbol{v}_j) = \frac{1}{Q}\sum_{j=1}^{Q} \boldsymbol{y}_j$ is the average of the simulated categorical distributions. Let $\bar{\boldsymbol{\eta}} = \frac{1}{M}\sum_{m=1}^{M} \boldsymbol{\eta}^{(m)}$ be the average label. Then, the $\mathrm{NLL}_i^{\mathrm{all}}$ for a single input $\boldsymbol{x}_i$ is given by

$$\begin{aligned}
\mathrm{NLL}_i^{\mathrm{all}} &= -\frac{1}{M}\sum_{m=1}^{M}\sum_{k=1}^{K} \boldsymbol{\eta}_{i,k}^{(m)} \log \mathrm{P}_{\boldsymbol{\theta}}(c = k|\boldsymbol{x}_i) \\
&\approx -\frac{1}{M}\sum_{m=1}^{M}\sum_{k=1}^{K} \boldsymbol{\eta}_{i,k}^{(m)} \log \bar{\boldsymbol{y}}_{i,k} \\
&= -\sum_{k=1}^{K} \bar{\boldsymbol{\eta}}_{i,k} \log \bar{\boldsymbol{y}}_{i,k},
\end{aligned} \tag{D.9}$$

which is the cross-entropy between the averaged label and averaged sample.

Appendix E

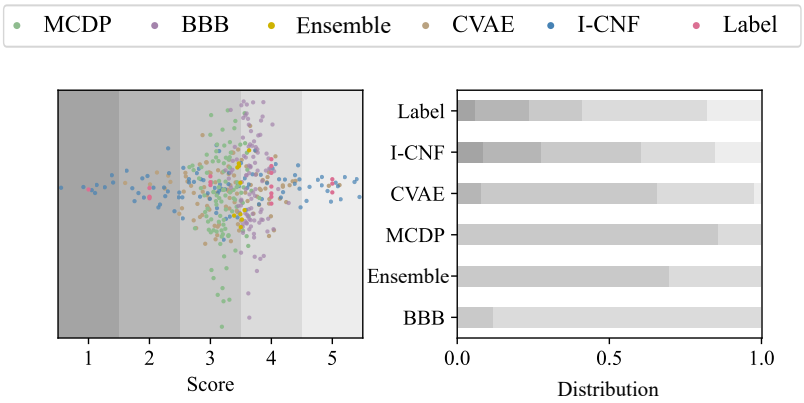
Further Visualised Examples for Chapter 7

E.1 Further Visualised Examples for I-CNF

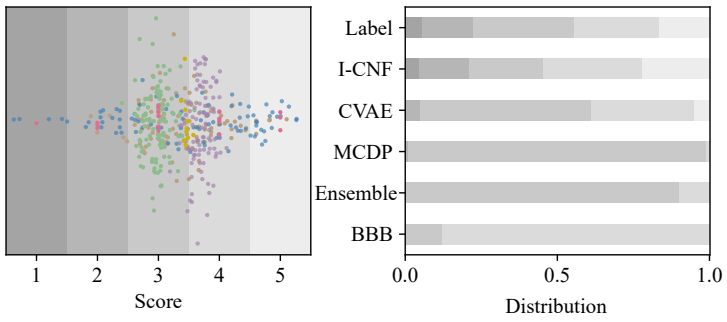
This section presents several additional visualised cases for speech quality evaluation. The abbreviation of compared methods are summarised below:

- Label: human annotations provided in the dataset;
- MCDP: Monte-Carlo dropout;
- BBB: Bayes-by-backprop with a standard Gaussian prior;
- Ensemble: deep ensemble of 10 systems;
- CVAE: conditional variational autoencoder.

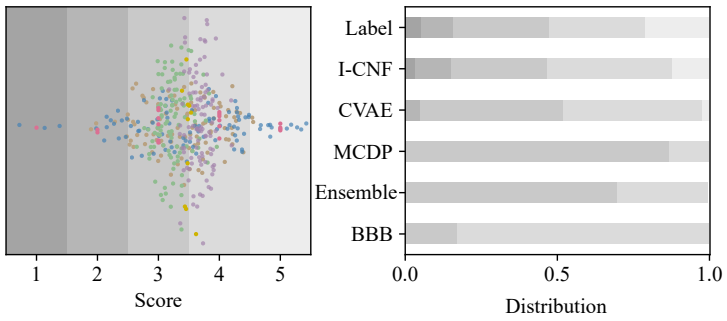
Generated samples (before rounding) are plotted in the sub-figures on the left. For clearer visualisation, the samples are spread along y axis according to density to avoid overlapping. As can be seen, samples generated by the proposed I-CNF method (in blue) can better simulate the diversity of human annotations (in pink).



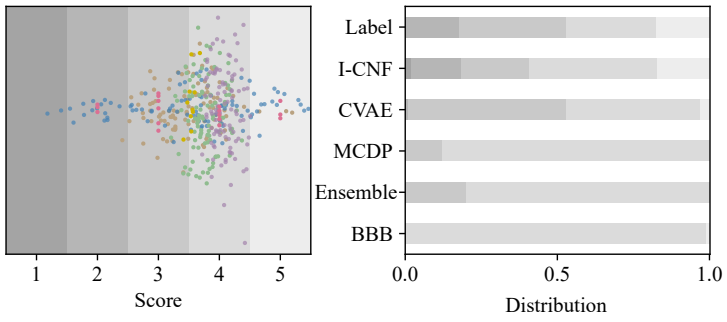
(a) Utterance "reportorial_2011_0154_145"



(b) Utterance "reportorial_2011_0096_186"



(c) Utterance "wiki_0033_055"



(d) Utterance "novel_2011_0011_110"

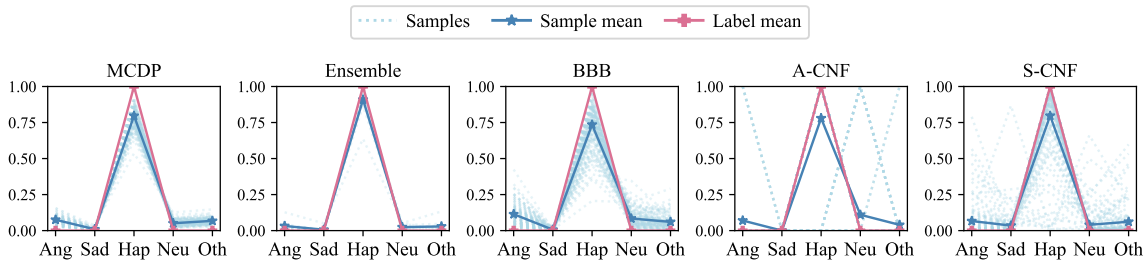
Fig. E.1 Additional visualisation of simulated annotations on the speech quality assessment task for case study. For the visualisation purpose, the points that have same x values are spread along y-axis according to density to avoid overlapping.

E.2 Further Visualised Examples for S-CNF

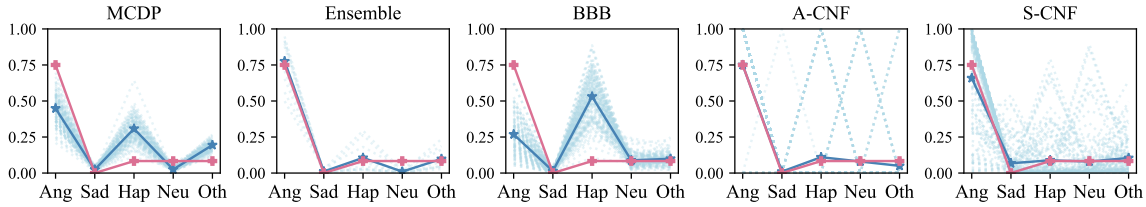
This section shows additional visualised examples for emotion class labelling when human annotators reach a consensus (Fig. E.2 (a)(b)), diverge (Fig. E.2 (c)(d)), and give distinct labels (Fig. E.2 (e)). The abbreviation of compared methods are summarised below:

- MCDP: Monte-Carlo dropout;
- Ensemble: deep ensemble of 10 systems;
- BBB: Bayes-by-backprop with a standard Gaussian prior;
- A-CNF: conditional argmax flow.

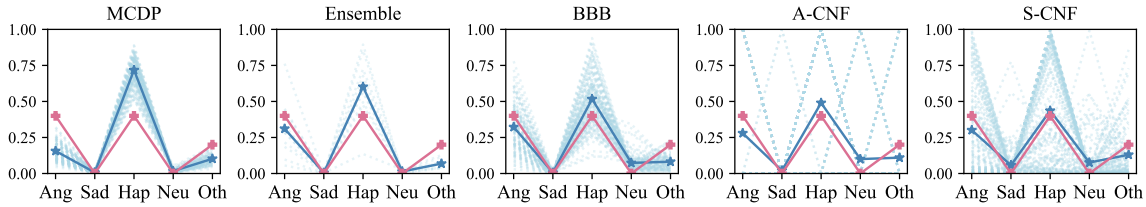
As can be seen, the proposed S-CNF can better simulate the aggregated behaviour as well as the variability of human annotations in all cases.



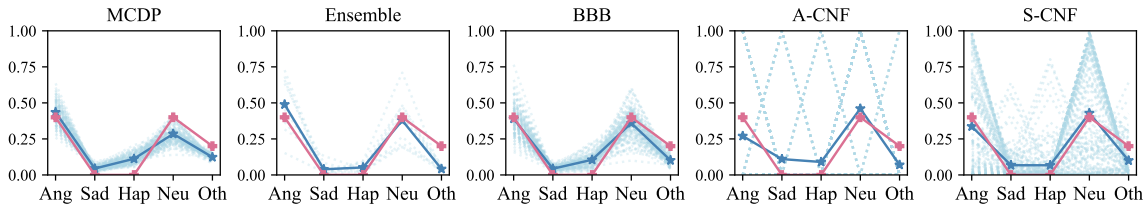
(a) Utterance "MSP-PODCAST_1216_0067.wav"



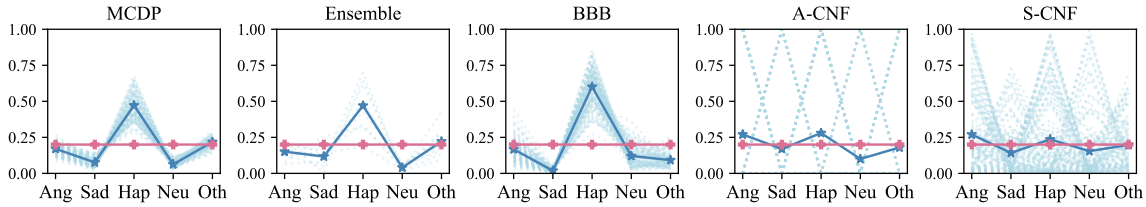
(b) Utterance "MSP-PODCAST_0566_0220"



(c) Utterance "MSP-PODCAST_0584_0145.wav"



(d) Utterance "MSP-PODCAST_0876_0069.wav"



(e) Utterance "MSP-PODCAST_0587_0073.wav"

Fig. E.2 Additional visualised examples for emotion class labelling. The y-axis corresponds to the probability mass. Each sample is a categorical distribution. The probability mass values of different categories in each categorical distribution are connected for the purpose of better visualisation. CVAE is omitted because it collapses to one category for all inputs.