Computational Paralinguistics: The Importance of Audio Analysis and Feature Extraction Methodologies

PhD Thesis

Mercedes Kiss-Vetráb

Supervisor:
Dr. Gábor Gosztolya

Doctoral School of Computer Science
Department of Computer Algorithms and Artificial Intelligence
Faculty of Science and Informatics
University of Szeged

Szeged
2025

Contents

List of Figures

List of Tables

Abbreviations

ASR Automatic Speech Recognition 16, 48, 50, 53, 54, 57, 58, 62, 63, 75, 97

AUC Area Under the ROC Curve 26, 65–67

BoAW Bag-of-Audio-Words i, 1, 7–9, 12, 15, 29–34, 36–42, 44–47, 77, 94–97

BoVW Bag-of-Visual-Words 30

BoW Bag-of-Words 30

DNN Deep Neural Network i, ii, 1, 7, 8, 12–16, 47–50, 52–55, 57–60, 62–64, 66–68, 70, 72–76, 78–80, 93, 94, 97, 98

F-bank Filter-bank 14, 20, 50, 58, 65

GMM Gaussian Mixture Model 47, 48

GRU Gated Recurrent Unit 48, 60, 65

HMM Hidden Markov Model i, 1, 7, 8, 13, 16, 47–50, 52, 54, 55, 78, 94, 97

LSTM Long-Short Term Memory 48, 60

MCI Mild Cognitive Impairment 19, 63–68, 98

MFCC Mel-Frequency Cepstral Coefficients 12, 14, 30, 58

PCA Principal Component Analysis 9, 43–46, 96, 97

Seq. Autoencoder Sequence-to-Sequence Autoencoder ii, 1, 7, 8, 12, 16, 57, 58, 60, 62–65, 67, 68, 75, 78, 94, 98

SFS Sequential Forward Selection 21, 52, 55, 72–74

SotA State-of-the-Art ii, 7, 8, 12, 13, 57, 58, 60, 62, 64, 66, 68, 70, 72, 74–76, 78, 80, 94, 99

SVM Support Vector Machine 10, 22, 23, 34, 39, 50, 51, 65, 69, 98

SVR Support Vector Regression Machine 23, 42

UAR Unweighted Average Recall 1, 25, 34, 39, 51, 70–73

Wav2Vec 2.0 Wav-to-Vec 2.0 Neural Network ii, 7, 8, 12, 16, 57, 58, 61–63, 68, 69, 71, 74–76, 78, 79, 93, 94, 99

X-Vector X-Vector Neural Network 1, 12, 16, 24, 51–53, 55, 58, 65

Chapter 1

Introduction

Communication extends far beyond the words we speak. In every human interaction, many non-verbal cues accompany our verbal expressions. For example, the tone of our voice, temporal dynamics, and the subtle intonations that broadcast emotions, intent, speaker state and meaning. While these acoustic cues have been fundamental to human communication throughout our evolutionary history, the computational analysis of paralinguistic information represents a relatively recent yet rapidly advancing field in speech technology. However, despite rapid advances in machine learning and speech processing, the field of computational paralinguistics lacks unified methodological frameworks that can reliably generalise across different tasks, languages, and datasets.

This current work addresses fundamental challenges by systematically investigating feature extraction methodologies and architectural design choices in computational paralinguistics. Through a comprehensive analysis of traditional, hybrid, and State-of-the-Art approaches, this thesis aims to establish global guidelines for specific algorithmic choices. The research encompasses three interconnected methodological streams: I. Traditional Feature Extraction Methods (Bag-of-Audio-Words), II. Hybrid Approaches (HMM/DNN Integration), and III. Deep Learning Techniques (Sequence-to-Sequence Autoencoder and Wav-to-Vec 2.0 Neural Network). These experiments collectively address the critical need for robust, generalisable solutions in computational paralinguistics while providing practical guidelines for researchers and developers working across diverse paralinguistic applications.

The thesis is organised into three main chapters, each representing a key thesis point. The datasets, methods, experiments, and results discussed in this thesis have been detailed in the earlier works of the authors. Table 1.1 highlights how these works relate to the specific thesis points.

The thesis is structured into four main chapters. Chapter 2 briefly outlines some of the most important fundamental concepts relevant to multiple parts of this thesis, including the technical challenges of working with small corpora and variable-

	[94]	[55]	[92]	[91]	[93]	[96]	[90]	[46]	[95]
I/1.	•	•							
I/2.			•	•					
I/3.					•				
II/1.						•			
III/1.							•	•	
III/2.									•

Table 1.1: *The relation between the theses and the corresponding publications*

length utterances, the databases employed throughout the research, and the evaluation methodologies used across various experiments.

The first part in Chapter 3 describes the results of our work in traditional feature extraction methodologies, focusing on the Bag-of-Audio-Words (BoAW) technique. This chapter provides an understanding of how traditional machine learning approaches handle paralinguistic tasks and why parameter optimisation is crucial for achieving robust performance. The chapter's main contributions include systematic parameter optimisation strategies, demonstration of corpus independence capabilities, and methods for handling the stochastic nature of clustering-based feature extraction.

The second part in Chapter 4 investigates hybrid methodologies that combine traditional statistical methods with modern deep learning approaches. It provides a detailed analysis of the HMM/DNN hybrid system. It explains why selecting the proper aggregation strategy is fundamental to achieving optimal performance. The chapter aims to provide a comprehensive understanding of how traditional Hidden Markov Models can be effectively combined with Deep Neural Networks for paralinguistic feature extraction, bridging the gap between traditional and State-of-the-Art approaches.

The third part in Chapter 5, dealing with State-of-the-Art Deep Neural Network methodologies. It describes how modern deep learning approaches, specifically the Sequence-to-Sequence Autoencoder and Wav-to-Vec 2.0 Neural Network models, can be optimised for paralinguistic tasks. The chapter includes experiments in audio preprocessing optimisation and advanced aggregation strategies. It establishes guidelines for different architectural choices and demonstrates how proper preprocessing and aggregation techniques can significantly improve classification performance across diverse paralinguistic tasks.

These findings address real-world constraints, including limited computational resources, small dataset sizes, and the need for cross-corpus generalisation. These challenges are particularly relevant for different paralinguistic applications, edge computing deployment, and rapid prototyping scenarios. The thesis concludes with a final

part that contains comprehensive summaries in both English and Hungarian. It is a detailed description of the thesis points and a comprehensive list of the author's contributions and publications related to this research.

1.1 Contributions

The ideas, figures, tables and results included in this thesis were published in scientific papers (listed at the end of the thesis). The author has the following contributions presented in this chapter:

Chapter 3.:

I/1. The author implemented the Bag-of-Audio-Words feature-extraction pipeline for emotion recognition, including parameter optimisation for preprocessing, codebook generation, quantisation, and feature transformation. She has taken care of the experimental setup, ensuring speaker-independent evaluation and systematic testing of parameter ranges such as codebook size, neighbour count, clustering algorithms, and delta feature computation. The author performed data preparation, feature extraction using openSMILE and openXBOW, and integrated the output into SVM-based classification. She conducted multiple iterations of experiments to analyse the effect of each parameter on classification accuracy, performed statistical comparison of configurations, and identified optimal parameter settings, while also documenting the findings.

I/2. The author implemented an experimental framework for analysing corpus independence in Bag-of-Audio-Words feature extraction. She preprocessed the three different databases, constructed cross-corpus codebooks, and ran systematic tests with an emotion recognition task, while documenting all the results and conclusions.

I/3. The author implemented the experimental framework for the stochastic variability of Bag-of-Audio-Words feature extraction, while documenting all of the results. She developed an infrastructure for repeated feature extraction with multiple random seeds. The author implemented various aggregation strategies and addressed the data dimensionality issue using Principal Component Analysis.

Chapter 4.:

II/1. The author implemented an aggregation pipeline, which included various aggregations, feature transformations, and classification. She has taken care of this experimental setup, ensuring systematic testing in different databases. The author performed data preparation and integrated the output into SVM-based classification. She conducted multiple iterations of experiments to analyse the effect of layers and aggregations. She performed the comparison of configurations and identified optimal parameter settings while also documenting the findings.

Chapter 5.:

III/1. The author implemented a classification pipeline that included various prediction-level aggregations and feature transformations. Based on already calculated posterior values, she has taken care of this experimental setup, ensuring systematic testing. The author performed data preparation and evaluated the posteriors. She conducted multiple iterations of experiments to analyse the effect of noise reduction and aggregations. She performed a comparison of configurations, identifying optimal noise reduction settings and aggregation techniques, while also documenting her findings.

III/2. The author implemented the classification pipeline, which included various aggregations and feature transformations. Based on already calculated frame-level feature vectors, she has taken care of the experimental setup, ensuring systematic testing in different databases. The author performed data preparation and integrated the output into SVM-based classification. She conducted multiple iterations of experiments to analyse the effect of aggregation methodologies and the features given by different layers. She performed a comparison of configurations, identifying optimal layer settings and aggregation techniques, while also documenting her findings.

Chapter 2

Fundamentals

2.1 Paralinguistics

> "**Paralinguistic:** Connected with the ways in which people show what they mean other than by the words they use, for example by their tone of voice, or by making sounds with the breath." – Cambridge Dictionary

Computational Paraliguistics refers to the study of non-verbal communication cues that accompany speech, such as tone, pitch, volume, and speed. These elements can convey emotions, intentions, and meanings that words alone may not fully express. [72]

In the early stages of speech processing, initial research was primarily focused on creating systems for speech recognition, which aimed to transcribe spoken language into written text. However, in the 1990s, the introduction of paralinguistic concepts gained momentum as researchers started to recognise the significance of non-linguistic cues in human communication and the rich information embedded in the way we speak. They explored concepts such as emotion recognition, speaker identification, speaker verification, analysis of prosodic characteristics, health conditions, and inspection of other speaker traits. This marked the beginning of computational paralinguistics as a distinct field within speech processing. The interdisciplinary nature of computational paralinguistics has fostered collaborations between linguists, computer scientists, psychologists, and healthcare professionals. [72]

Nowadays, computational paralinguistics is an interdisciplinary field that focuses on analysing and understanding the non-linguistic aspects of human communication, such as

- speaker recognition and diarisation ("who's speaking when") [32],
- speech compression [51],
- cognitive load measurement [23, 89],
- detecting Parkinson's disease [42, 44, 98]

- detecting Alzheimer's disease [10, 64, 66],
- identifying Multiple Sclerosis symptoms [101],
- assessing the level of depression [15],
- recognising age, gender [68]
- recognising emotion [59, 105],
- identifying laughter events [22],
- estimating the degree of sleepiness [13]
- estimating conflict intensity [28],
- detecting whether the speaker is intoxicated [5]

Some of the trends focus on handling more tasks simultaneously to investigate task interdependencies, developing large and varied datasets, optimising features, and fusing linguistic and non-linguistic information. Overall, the ability to automatically process these non-verbal cues has opened up new possibilities for more natural and effective human-machine interaction, with applications ranging from healthcare support to security systems and personalised user experience. [77]

In this thesis, the primary focus is on feature extraction in the context of computational paralinguistics, which serves as a crucial step in paralinguistic analysis. Both traditional and non-traditional methodologies are commonly used for this purpose.

Traditional approaches often involve hand-crafted feature extraction, rule-based methods and early machine learning algorithms [68, 72, 85]. These methods, such as calculating Mel-Frequency Cepstral Coefficients (MFCC) and Bag-of-Audio-Words (BoAW), have been used for a long time to capture acoustic features. Traditional methodologies can outperform end-to-end DNNs in low-resource environments (such as in many paralinguistic use cases) [22, 82, 89]. When Deep Neural Networks are not an option, either due to data limitations, annotation scarcity, or infrastructure constraints, then traditional methods provide robust and practical solutions. This makes these methods a good choice for rapid prototyping, testing, and deployment in real-world and edge environments.

At the same time, State-of-the-Art (SotA) methodologies take advantage of deep learning techniques to automatically learn representations from data. Newest models, such as Sequence-to-Sequence Autoencoder (Seq. Autoencoder) [90] and X-Vector Neural Network (X-Vector) [14], or self-supervised models like Wav-to-Vec 2.0 Neural Network (Wav2Vec 2.0) [95], have revolutionised feature extraction by learning contextual embeddings directly from raw audio. These solutions may outperform traditional methods if sufficient data and computational resources are available. While traditional methods and deep learning approaches each have their strengths and weaknesses, there's a growing trend towards using DNNs for feature extraction, particularly with pre-trained models. However, the field currently lacks consensus, with solutions often being database and topic-dependent. Additionally, hybrid ap-

proaches, such as combining a traditional Hidden Markov Model (HMM) with Deep Neural Network (DNN), have demonstrated resource-efficient solutions [13, 50, 80]. Whether traditional or SotA solutions are best depends on the paralinguistic task at hand.

2.2 Technical Challenges

Computational paralinguistics involves several technical challenges. These challenges arise from the complexities involved in accurately capturing and interpreting the nuanced elements of human speech. In addition, the integration of diverse data sources and the necessity for robust algorithms to process and analyse this data highlight the development of effective paralinguistic models.

2.2.1 Small Corpora

One of the significant challenges in computational paralinguistics is dealing with small corpora or limited amounts of labelled data. As the analysis of data often requires large and diverse datasets, the lack of samples poses a major difficulty in training robust and accurate models. One reason for this is that each use case usually requires specific recording protocols and annotations. It means that usually there are just a few hundred (or at most, a few thousand) examples for a particular subtopic or class. There are different strategies, such as data augmentation, transfer learning, or domain adaptation, that can help mitigate this effect.

Furthermore, careful experimental design and validation procedures are essential in each paralinguistic task. Commonly, the optimal hyperparameters of a model depend on the current database and the paralinguistic topic being studied. The problem is that each database has different characteristics, such as variations in speaking styles, recording quality, and noise levels. Hence, the optimal hyperparameters for different databases and use cases are mostly not the same, and task interdependencies always require further research. The choice of hyperparameters can significantly impact the model's accuracy, robustness, ability to generalise to unseen data and performance on different paralinguistic tasks. This means we are highly dependent on the quality of the research databases, and it is challenging to create a solution that performs as well in real-life scenarios as it does in the experimental setup.

2.2.2 Variable-Length Utterances

Another crucial technical challenge in computational paralinguistics is getting a fixed-sized feature vector from variable-length speech segments. In computational paralinguistics, we aim to assign a single label output (classification or regression) to

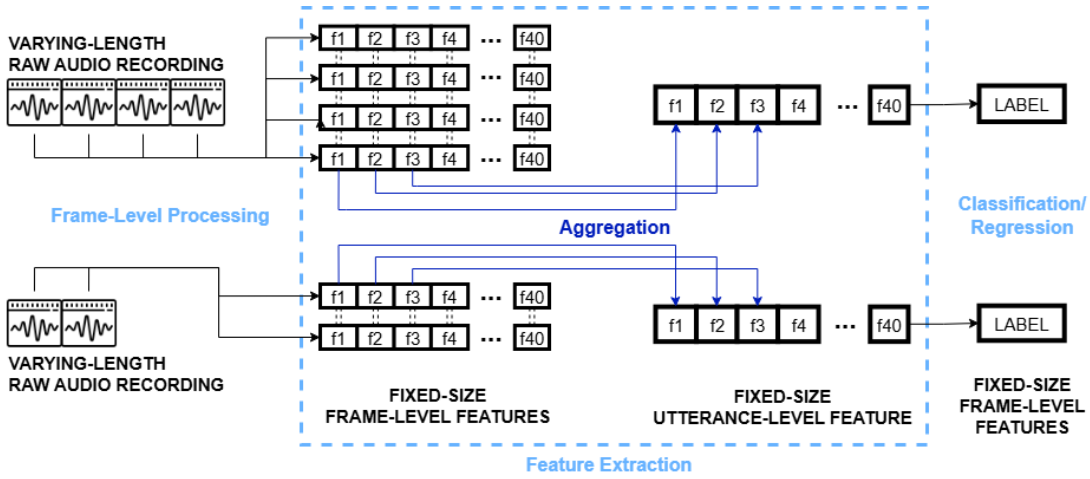

Figure 2.1: *Creating fixed-sized feature vectors from varying-length utterances.*

an audio recording (utterance) of varying length. For instance, we need to extract features from recordings and then determine whether the speaker's emotion is angry or not. Figure 2.1 illustrates a computational paralinguistic pipeline that processes audio recordings to extract non-verbal information from speech for classification or regression tasks. The workflow demonstrates how raw audio files with different durations are converted into fixed-size features suitable for machine learning models:

1. The pipeline begins with varying-length raw audio recordings represented as waveforms. These recordings can have different durations. The raw audio is divided into short-duration frames (segments). The length of each segment depends on the type of feature extraction technique that will be used. There are several common windowing types and sizes, such as Hamming or Gaussian windows, with sizes ranging from 20ms to 40ms [73, 75, 76].

2. The next step is frame-level processing. Fixed-sized feature vectors are extracted from each frame. The size and the characteristics of the vector depend on the processing methodology. It is labelled as f1, f2, f3, f4... f40 in our figure. In case of Traditional Machine Learning models, we commonly calculate hand-crafted features that are manually designed acoustic features, such as MFCC and F-bank, as well as spectral or prosodic features, among others. Each frame-level feature vector contains the same number of feature dimensions, but the total number of vectors varies with audio length. This produces sequences of fixed-size frame-level features. In case of most DNN models, the network itself learns hierarchical feature representations directly from raw frames. It produces dense feature vectors directly, which can replace hand-crafted features [73, 74, 78].

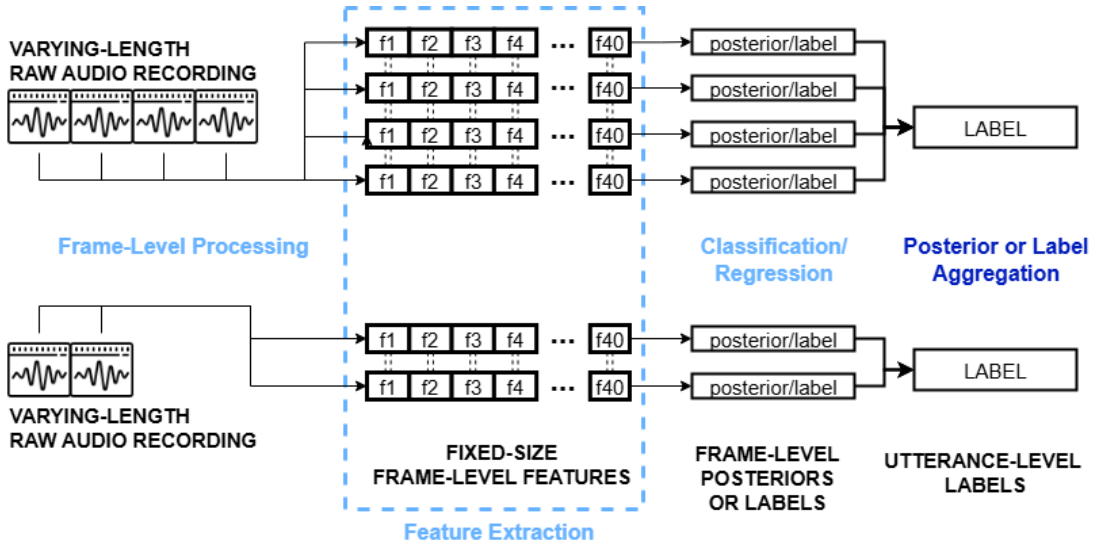

Figure 2.2: *Creating fixed-sized feature vectors from varying-length utterances.*

3. The next step is the Aggregation: To handle the varying number of frames per utterance, the pipeline employs an aggregation step that converts the variable-length sequence of frame-level features into a single fixed-size utterance-level feature vector. In the case of Traditional Machine Learning models, we commonly use statistical techniques to calculate feature-wise values. Various aggregation techniques compute summary statistics (such as mean, standard deviation, percentiles, BoAW features, and so on). In case of most DNN models, the aggregation is an integral part of the network and is performed by specific layers.

4. The resulting fixed-sized utterance-level feature vector serves as an input to a classification or regression model that outputs labels for various paralinguistic tasks such as emotion recognition or speaker age detection, and so on.

In addition to aggregating features at the frame level, there is a less commonly used but potentially valuable approach. In this method, we do not calculate utterance-level features. Instead, we conduct classification/regression individually for each frame, and the resulting frame-level posterior distributions or labels are combined to produce the final outcome. The process is shown in Figure 2.1

2.2.3 Using Deep Neural Networks for Computational Paralinguistic

The challenges mentioned earlier often make DNNs unsuitable to use as classifiers, and these methods are still in their early stages of development in this field [59, 88].

However, there is a growing trend among scientists to use DNNs for feature extraction. When discussing these networks, the result of feature extraction is often referred to as an embedding. DNN embeddings can be effectively represented in a low-dimensional feature space while retaining crucial information. These embeddings have been shown to capture complex relationships in data and can outperform traditional feature extraction methods.

Due to the limited size of paralinguistic datasets, training a feature extractor DNN from scratch is challenging, so researchers often use standard Automatic Speech Recognition (ASR) corpora for pretraining. Examples of such pretraining methods include HMM/DNN acoustic models [24], X-Vector [84], ECAPA-TDNN [83], Sequence-to-Sequence Autoencoder [90] and Wav2Vec 2.0 [50]. These pretrained models have demonstrated significant improvements in various paralinguistic tasks. The use of transfer learning from these models allows researchers to leverage large-scale ASR datasets and apply the learned representations to smaller, task-specific paralinguistic datasets. Despite these advancements, it is always a challenge to adapt these models to the specific requirements of different paralinguistic tasks.

2.3 Data and Methods

2.3.1 Databases

AIBO

The FAU AIBO Emotion Corpus [86] contains speech taken from 51 native German children. The children were selected from two schools. The database contains 9959 recordings from the Ohm school and 8257 recordings from the Mont school. The total duration is approximately 9 hours. The subjects had to play with a pet robot called AIBO. They were told that AIBO responds to their commands, but it was remotely controlled by a human. The Ohm school recordings are commonly used for training. The Mont school recordings were used for the test set. Because of the size of the training set, we were able to define a development set. We kept recordings of 20 children in the training set (7578 utterances) and used recordings of 6 children in the development set (2381 utterances). The original 11 emotional classes were merged to form a 5-class problem. The new classes were derived from the originals: Anger (angry, irritated, reprimanding), Emphatic, Neutral, Positive (motherese and joyful), and the Rest (helpless, surprised, bored, and non-neutral but not belonging to the other categories). This database was also employed in the INTERSPEECH 2009 Emotion Challenge [73].

URTIC

The Upper Respiratory Tract Infection Corpus (URTIC) [47] was provided by the Institute of Safety Technology, University of Wuppertal in Germany. It contains native German speech from 630 subjects (248 female, 382 male). The total duration is approximately 45 hours. The recordings have a sampling rate of $44.1\,\mathrm{kHz}$ downsampled to $16\,\mathrm{kHz}$. They were split into $28\,652$ chunks of 3 to 10 seconds. The participants were required to complete various tasks. They had to read short stories (e.g, a well-known story in the field of phonetics "The North Wind and the Sun"), had to produce voice commands (such as state numbers from 1 to 40), and they also had to narrate spontaneous speech (e.g, tell something about their best vacation). The number of tasks varied for each speaker. The database was split speaker-independently into training, development and test sets, where each one contained 210 speakers. The training and development sets contained 37 infected participants and 173 participants with no cold. There are 2 classes, namely cold and no cold. The purpose of the classification was to decide whether the speaker had a cold. This database was also employed in the INTERSPEECH 2017 Computational Paralinguistics Challenge [74].

iHEARu-EAT

The iHEARu-EAT corpus [33] was provided by the Munich University of Technology. It contains close-to-native German speech taken from 30 subjects (15 female, 15 male). It was recorded in a quiet, slightly echoing office room, and the recordings have a sampling rate of $16\,\mathrm{kHz}$. It contains approximately 2.9 hours of speech (sampled at 16 kHz). They were segmented into roughly equal parts. The participants had to perform practice trials to familiarise themselves with the procedure. The speakers had to complete different tasks, such as reading the German version of "The North Wind and the Sun" story, and they had to give a spontaneous narrative about their favourite activity or place. The number of recordings varied for each speaker because not everyone was willing to eat every type of food offered. The database was split speaker-independently into a training set (20 speakers) (sometimes a dev set from 6 train speakers) and a test set (10 speakers). There are 7 classes determined by the consistency: apple, nectarine, banana, crisp, biscuit, gummy bear and without any food. The classification aimed to recognise what the subject was eating while speaking. These types of foods typically allowed the participants to eat while talking. This database was also employed in the INTERSPEECH 2015 Computational Paralinguistics Challenge [75].

Sleepiness Database

The public German Sleepy Language Corpus (i.e, Sleepiness database) [39] was introduced in the Interspeech 2011 Speaker State Challenge [78]. It contains 16463 recordings. The recordings came from native German speakers (aged between 20-52 years). One part of it came from a story reading task, another part came from giving verbal commands to the GPS navigator, another from traffic controller communication statements, another from picture descriptions and another from giving a presentation. The dataset was divided into three speaker-disjunct sets as training, development and test sets. The training set contained 20 female and 16 male speakers, $5\,564$ recordings in total. The development (i.e dev) set contained 17 female and 13 male speakers, and $5\,328$ recordings in total. The test set contained 19 female and 14 male speakers, $5\,571$ recordings in total. The labels were defined by a subjective questionnaire that was filled in by the subject and three other assistants. The labels are integers and lie in the range from 1 to 10: extremely alert (1), very alert (2), alert (3), rather alert (4), neither alert nor sleepy (5), some signs of sleepiness (6), sleepy, without any effort to stay awake (7), sleepy, some effort to stay awake (8), very sleepy, great effort to stay awake, struggling against sleep (9), extremely sleepy, cannot stay awake (10).

HUN Emotion Dataset

The Hungarian Emotion corpus [87] was provided by the Department of Telecommunication and Media Informatics of the Budapest University of Technology and Economics. It contains utterances from 97 native Hungarian speakers. The voice samples came from recorded television shows. The vast majority of segments were recorded from an emotion-rich, continuous, spontaneous programme with actors. In this case, due to the acting performance, the samples are vivid, and the emotions are more clearly represented. The other part came from an improvisation entertainment show. The samples from this case are closer to real-life emotions due to the improvisation. The training set contains approximately 20 minutes of recordings, and the test set contains approximately 7 minutes of recordings. The sampling frequency of the samples is $16\,\mathrm{kHz}$. The database contains 1111 sentences, which were separated into an 831 sample training set and a 280 sample test set. In this thesis, we used samples from 4 classes: neutral, joy, anger, and sadness. The distribution of the emotions was not uniform. The training set sample distribution was: $\approx 57\%$ neutral, $\approx 6\%$ sad, $\approx 9\%$ joy and $\approx 27\%$ anger. The test set sample distribution was: $\approx 62\%$ neutral, $\approx 4\%$ sad $\approx 7\%$ joy and $\approx 27\%$ anger.

EmoDB

The German Emotion Speech Database (EmoDB) [6] was provided by the Technical University of Berlin. This database contains speech from 10 native German actors. The recordings were made with actors aged between 25 and 35. Each participant produced 10 German speeches (5 short and 5 longer sentences), all of them with a different emotion. The database contains a total of 535 utterances. The recordings were taken in an anechoic chamber with high-quality recording equipment. Recordings were taken with a sampling frequency of $48\,\mathrm{kHz}$ and later downsampled to $16\,\mathrm{kHz}$. The whole database contains approximately 25 minutes of recordings. The actors were standing in front of the microphone, allowing them to use body language if desired, only hindered by the cable of the laryngograph. They were speaking in the direction of the microphone at a distance of about 30 cm. There are 7 classes: neutral (79 samples), anger (127 samples), boredom (81 samples), disgust (46 samples), fear (69 samples), happiness (71 samples) and sadness (62 samples).

MCI Dataset

Our utterances were recorded at the Memory Clinic at the Department of Psychiatry of the University of Szeged, Hungary. The Mild Cognitive Impairment (MCI) data were recorded using a digital voice recorder and a tie-clip microphone. Recordings have a sampling rate of $44.1\,\mathrm{kHz}$ in stereo. Later, the recordings were converted to $16\,\mathrm{kHz}$ mono with a 16-bit resolution. A total of 50 subjects, selected from a larger pool of test participants, were used in the current thesis: 25 MCI patients and 25 healthy controls. Subjects were chosen to ensure that the two study groups did not differ significantly from each other in terms of gender ($p = 0.734$), age ($p = 0.150$), and years of education ($p = 0.214$). All the subjects were right-handed and native speakers of Hungarian. The exclusion criteria were drug or alcohol consumption, being under pharmacological treatment affecting cognitive functions, depression, a medical history of head injuries or psychosis, and visual or auditory deficits.

MCI patients were selected based on a medical diagnosis supported by neuropsychological tests and CT or MRI scans. Patients indicating any signs of dementia were not enrolled in this study. The following clinical tests were applied to assess the cognitive state of the subjects: Mini-Mental State Examination (MMSE), Clock Drawing Test (CDT) and ADAS-Cog. All techniques and procedures were performed per the Declaration of Helsinki, with approval from the University of Szeged Ethical Committee and the Regional Human Investigation Review Board, and written informed consent was obtained from all participants. They focused on spontaneous speech: in their protocol, the subjects were asked to talk about their previous day. The duration of the responses lay in the range of $25 \ldots 325$ seconds, with a mean duration of 89.8 seconds.

BEA

In this thesis, a subset of the BEA Hungarian corpus [60] is also used to pretrain acoustic models. This was not a specific paralinguistics corpus like the others mentioned above, but it is also a speech corpus. It contains only spontaneous speech, and it is suitable for generalising a neural network for speech processing. This subset includes the speech of 165 subjects ($\approx$ 60 hours). This subset contained only spontaneous speech, including special events such as filled pauses, breathing sounds, laughter, gasps, and other similar sounds. It has a transcription, where the phonetics set and the special events were also marked.

Hand-Crafted Features

ComParE

Hand-crafted features are manually designed acoustic or prosodic descriptors extracted from raw audio based on human knowledge of speech signal characteristics. In this thesis, the feature set of the INTERSPEECH 2013 Paralinguistic Challenge (so-called ComParE) is used many times [76]. It contains 65 frame-level features: 55 spectral, 6 voicing-related low-level descriptors and 4 energy-related. Mostly 60 ms frame (Gaussian window function) and a sigma value of 0.4 was used for the speech-related features; and a 25 ms frame (Hamming window function with a step size of 10 ms) for the others. Not only were basic features used, but their derivatives were also utilised. With the Δvalues, we aimed to obtain information about the dynamics of the speech samples over time.

F-bank

The human ear does not perceive all frequencies equally, and letting a speech system mimic this uneven sensitivity can improve recognition. Rather than using a straight-line analysis of the sound, a Filter-bank (F-bank) breaks the spectrum into overlapping bands that align with our hearing. The filters used are triangular, and they are equally spaced along the mel-scale. The triangular filters are spread across the entire frequency range, from zero. After applying the filters, the resulting energy in each filter band is captured, and these values are then transformed into a simple Fourier transform based F-bank designed to provide approximately equal resolution on a mel-scale. In the thesis points of this work, a 40 Mel-frequency filter bank was used with a standard window size of 25ms and a frame step of 10ms. Then 1 additional feature was added, the energy of the signal. Then, the first and second order derivatives were calculated (Δvalues and $\Delta\Delta$values). The final number of features in a frame-level vector was $41 * 3 = 123$. F-bank features were calculated using the HTK tool. [102]

2.3.2 Employed Research Methods

Aggregation Techniques

Aggregation could be done straightforwardly by calculating statistical values along the time axis of feature vectors. For example, we have an input recording with a number of N windows, each containing a number of y features. We can calculate statistical values from all the N vectors by taking a summary of each feature along the time axis. The used aggregations have the following mathematical formula, if we have N frame-level embeddings in the form $x_1, x_2, \ldots, x_i, \ldots, x_N$:

Arithmetic mean: $\overline{x} = \frac{1}{N}\sum_{i=1}^{n} x_i$

Standard deviation: $\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \overline{x})^2}$

Kurtosis $= \frac{1}{N}\sum_{i=1}^{N}\frac{(x_i-\overline{x})^4}{\sigma^4}$

Skewness $= \frac{1}{N}\sum_{i=1}^{N}\frac{(x_i-\overline{x})^3}{\sigma^3}$

Zero ratio $= \frac{1}{N}\sum_{i=1}^{N} y_i$, where $y_i = \begin{cases} 1 & \text{if } x_i > 0, \\ 0 & otherwise \end{cases}$

Percentile: The p-th percentile of N values $(x_{(1)} \leq x_{(2)} \leq \cdots \leq x_{(N)})$, is the value below which $p\%$ of the observations fall.

Sequential Forward Selection (SFS)

The basic idea behind SFS is to initialise a subset with only the best method, and then iteratively add one more aggregation to the subset, based on which combination provides the greatest improvement in performance.

For example, to select the best combination of a set of aggregations, we can measure the efficiency of each aggregation and then select the best one. In the next iteration, we can measure the efficiency of the best aggregation combined with one more aggregation (in every possible combination) and choose the best combination. With this iterative method, we can reduce the risk of overfitting, as the selected subset is more likely to be the most relevant and informative for the given prediction task.

K-fold Cross-Validation

In this thesis, k-fold cross-validation was employed multiple times as a method for parameter optimisation. The process can be outlined as follows:

1. Data Segmentation: The training dataset was divided into k approximately equal-sized segments, known as folds. Each speaker's data was allocated to only one fold, ensuring that the data in each fold was entirely speaker-independent.

2. Model Training: For k iteration, we trained a model. Each model used a different $9/10$ part of the folds for training and the remaining $1/10$ for testing. It was done for every possible combination of folds.

3. Prediction Generation: By designating each fold as the test set during all of the k evaluations, we were able to generate predictions for all the samples in the training dataset. Upon completing all evaluations, we obtained a single prediction for each sample.

4. Performance Assessment: Following the k evaluations, all of the predicted scores were collected. Finally, any evaluation metrics can be computed across all samples.

2.3.3 Machine Learning Models

Support Vector Machine (SVM)

SVMs were initially designed for two-class learning, but it was later extended to multi-class classification. In the case of two classes, we assume that in a multidimensional space, the samples are arranged in such a way that we can divide the space into two parts with a hyperplane, so that only samples belonging to a given class fall on one side of the hyperplane. So there should be a hyperplane with only samples from the same class on one side. This hyperplane can be a line in 2D space or a plane in an n-dimensional space, where n is the number of features for each observation in the dataset. If our dataset is truly linearly separable, then the number of such hyperplanes is infinite. In this case, we have to decide which of the possible planes will have the best generalisation ability [54].

To decide this question, we will use a so-called decision margin. We are measuring the distance to the first nearest sample in both directions from the decision boundary. Our new goal will be to place the hyperplane in space so that the size of the corresponding margin can be maximised, while maintaining an equal distance from the decision boundary on both sides. In this case, we will obtain the best generalisation ability. Figure 2.3 represents this process. The margin is the maximal width of the slab parallel to the hyperplane that has no interior data points. The data

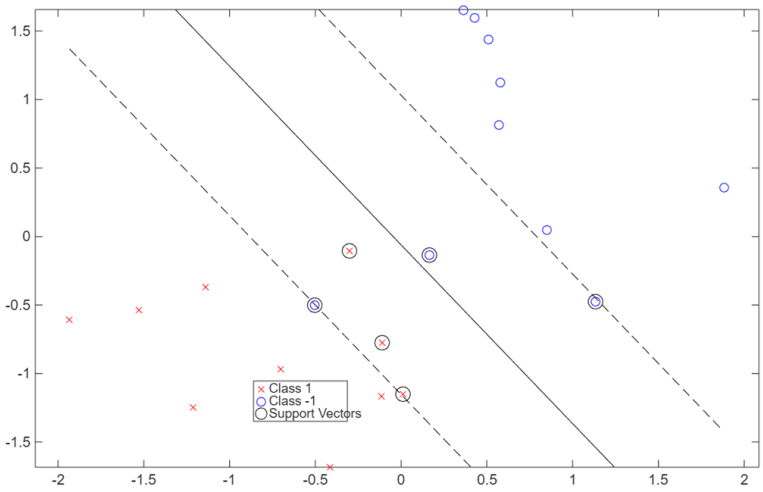

Figure 2.3: *Maximising the margin of the decision plane of the SVM algorithm [54].*

points that mark the boundary of this parallel slab and are closest to the separating hyperplane are the support vectors. Support vectors refer to a subset of the training observations that identify the location of the separating hyperplane [54].

Of course, in real life, samples are unlikely to be separable with a linear hyperplane. For this reason, we introduce an error threshold, which allows us to control the number of patterns that are allowed to fall within the decision margin. Margin violation penalties are controlled by the hyperparameter C. For nonlinearly separable data, nonlinear support vector machines use kernel functions to transform the features. The number of support vectors determines the number of transformed features [54].

Kernel functions map the data to a different, often higher-dimensional space. This transformation can make the classes easier to separate by simplifying the complex nonlinear decision boundary to a linear boundary in the higher-dimensional, mapped feature space. In this process, commonly known as the kernel trick, the data does not have to be explicitly transformed, which would be computationally expensive [54].

Support Vector Regression Machine (SVR)

SVMs are primarily used for classification tasks, but they can also be adapted for regression. Support Vector Regression Machine (SVR) relies on kernel functions. The working principle of SVR is the same as that of support vector machine classifiers, except that SVR aims to predict continuous values instead of discrete classes. In this case, the margin will define a region around the regression hyperplane where predictions are considered "good enough." If a data point falls within this tube, the model doesn't penalise it. Using the kernel trick, we can perform nonlinear regression by mapping data to a high-dimensional space.

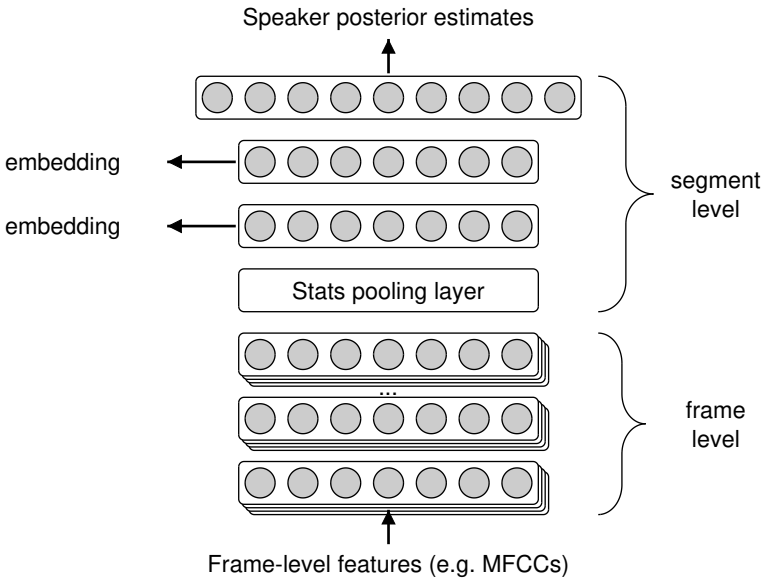

Figure 2.4: *Structure of X-Vector Neural Network [84].*

X-Vector Neural Network (X-Vector)

X-Vector was initially developed for speaker recognition [84]. Nowadays, X-Vectors have become a strong baseline in computational paralinguistics [13, 15, 38, 42]. Figure 2.4 visualises the structure of the neural network. It begins with lower frame-level layers using a time-delayed architecture. There are three time-delayed layers in total. The first layer takes a 5-slice speech frame with a context of [t-2, t+2] as input and produces an output size of 512. The second layer takes a 3-slice output from the previous layer with a context of (t-2, t, t+2) and also produces an output size of 512. The third layer takes a 3-slice output from the previous layer with a context of (t-3, t, t+3), again resulting in an output size of 512. This temporal context builds upon the earlier layers, allowing the third layer to consider a total context of 15 frames [84].

Following these frame-level layers, there are fully connected layers with input and output sizes of 512. Subsequently, there is another fully connected layer with an input size of 512 and an output size of 1500. After the last frame-level layer, the network establishes a connection to the next utterance-level (segment-level) block using a special ”statistics pooling” layer. This middle layer aggregates the output segments from the last frame-level layer and calculates the mean and standard deviation of the activations [84].

Following the middle layer, there are three more fully connected layers with input-output sizes of 3000-512, 512-512, and 512-N. This network design enables us to train for speaker identification, as the output of the last softmax layer has N neurons, corresponding to the number of speakers in the training set. It predicts labels over frame-level features while effectively handling variable-length utterances [84].

2.3.4 Evaluation Metrics

Unweighted Average Recall (UAR)

Unweighted Average Recall measures the average recall across all classes without considering class imbalance. To calculate UAR, you compute the recall for each class and then take the average across all classes. Recall, also known as sensitivity or true positive rate. It is the proportion of true positive instances (correctly identified instances) out of all actual positive instances. UAR is called ”unweighted” because it treats each class equally, regardless of class size or prevalence. This makes it suitable for datasets with imbalanced class distributions, where some classes may have significantly fewer instances than others. It provides a balanced view of the overall performance of a classification system, taking into account the performance across all classes equally [56].

For example, in an emotion recognition task with an imbalanced dataset (Happy: 500, Sad: 300, Neutral: 2000), accuracy can be misleading. If a classifier only predicts the majority class (Neutral) for all instances, then it would have high accuracy ($2000/2800 \approx 71.4\%$). However, UAR provides a more accurate evaluation. In this case, it would indicate a worse performance (UAR: $(0 + 0 + 1)/3 \approx 0.333$) as the classifier fails to identify instances of the minority classes (Happy and Sad) while performing well on the majority class [56].

The Unweighted Average Recall is calculated by taking the average of the recall values across all classes, treating each class equally regardless of its frequency: UAR $= \frac{1}{K} \sum_{k=1}^{K} \frac{TP_k}{TP_k + FN_k}$, where K is the total number of classes, TP_k is the true positives for a class, and FN_k is the false negatives for a class [56].

Pearson and Spearman Correlations

The Pearson correlation coefficient measures the strength and direction of the linear relationship between two continuous variables, X and Y. The range of the result is between -1 (perfect negative linear association) and +1 (perfect positive linear association), with 0 indicating no linear relationship at all. A strong assumption underlying the metric is that both variables are measured on an interval scale, are approximately normally distributed, and that their relationship is linear.

The Spearman correlation coefficient assesses the strength and direction of a monotonic relationship between two variables, X and Y. The range of the result is between -1 and +1, where the edge values indicate a perfect monotonic relationship (increasing (+1) or decreasing (-1)), and 0 means no monotonic relationship at all. In this case, there is no assumption about normality or linearity, only that the relationship is monotonic.

Area Under the ROC Curve (AUC)

Area Under the ROC Curve is a performance measurement for the classification problems at various threshold settings. In binary classification, the Receiver Operating Characteristic (ROC) curve plots the True Positive Rate against the False Positive Rate at different decision thresholds. The Area Under the ROC Curve (AUC) quantifies the overall ability of the classifier to discriminate between two classes. The range of the result is between 0 (perfectly swapping the classes, like predicting 0s as 1s and 1s as 0s) and 1 (perfect predictions). The value 0.5 is the worst, as it indicates that the model performs no better than random guessing. The higher the (AUC), the better the model is at predicting 0 classes as 0 and 1 classes as 1.

2.4 Motivation

Computational paralinguistics faces significant challenges, including small corpora, variable-length speech segments, and the need for task-independent embeddings. Despite these obstacles, the field continues to advance, driven by its potential for wide-ranging applications in human-machine interaction, healthcare, security, and personalised user experiences. Understanding the non-verbal aspects of human communication plays a crucial role in numerous applications. The ability to automatically infer emotions, speaker identities, and other paralinguistic attributes from speech signals has the potential to support a wide range of domains, such as healthcare, customer service, education, and entertainment.

It can be used in human-computer interfaces, such as monitoring human communication and detecting the emotional state of the speaker, or their level of confidence [41]. We can also utilise paralinguistics in dialogue systems to detect problematic dialogue phrases or adapt the dialogue to assist the speaker [7]. Moreover, we can utilise emotion detection in call centres [97]. For instance, if the client becomes angry, we can automatically notify a supervisor. Besides this, it may be helpful in healthcare systems to monitor the patient's mental state and may be useful for assessing patients. [37, 97] Furthermore, we can also utilise it for therapist support diagnostics and create more empathetic healthcare robots. In the future, we will be able to create more human-oriented systems. For example, we can develop intelligent tutorial systems that adapt to the student's mental state and provide more constructive advice. Additionally, we can utilise it for lie detection to enhance law enforcement. In computer games, we can use it to set the game's difficulty based on the user's emotions [43]. Human-computer interfaces and user adaptation systems could be used to recognise the age and gender of the speaker from their voice. For instance, we can use these human features in an automatic dialogue system to adapt to the speaker by speaking slower and louder for an older user or use a different corpus

for younger customers. An interactive voice response system can select background music based on guessing the age and the gender of the user. A smart home system can adapt to the needs of older customers with more automation, while adapting to the needs of younger customers with a more collaborative system. Last, but not least, a police call analysis system can identify the age and the gender of a suspect from a telephone call [68].

By comprehending the motivations driving research in this field, we gain a deeper appreciation for the significance and real-world impact of computational paralinguistics. These motivations directly highlight the development of task-independent solutions, which aim to capture general speech characteristics across various paralinguistic tasks. It is a crucial step towards addressing the technical challenges in computational paralinguistics while simultaneously advancing its real-world applications. Nowadays, there are no consents about feature-extraction methodologies in the field. Published solutions are mainly topic- and database-dependent. Lastly, achieving comparability among research findings is quite challenging. The range of datasets is quite wide (e.g, numerous initial studies presenting results from their unique and self-collected datasets), and the used evaluation metrics are diverse. Based on these aspects, this thesis focuses on finding global best practices. My goal is to construct guidelines that can help identify common directions for different paralinguistic tasks and methodologies.

Chapter 3

Bag-of-Audio-Words as a Traditional Feature Extraction Method

3.1 Chapter Overview

Traditional machine learning models have long served as the backbone of computational paralinguistics. These conventional techniques are especially vital in scenarios characterised by limited available data or restricted computational resources, as in many paralinguistic research and clinical applications. This makes these methods a good choice for rapid prototyping, testing, and deployment in real-world and edge environments. Additionally, traditional methods provide reliable benchmarks for evaluating new techniques, ensuring that innovations advance the field rather than merely improve performance. In this chapter, the focus is on the Bag-of-Audio-Words (BoAW) traditional method. There are three main works with different core thesis points.

Section 3.2 summarises the related works and briefly presents previous research results with the Bag-of-Audio-Words method. Section 3.3 introduce the Bag-of-Audio-Words (BoAW) method. Thesis Work I/1 is covered in Section 3.4.1. Thesis Work I/2 is covered in Section 3.4.2. Thesis Work I/3 is covered in Section 3.4.3. Results and final thoughts are summarised in Section 3.5.

3.2 Related Works

Since the beginning of research in computational paralinguistics, various feature extraction and classification techniques have been used along with different datasets to achieve the best results. One of the most challenging problems in speech emotion recognition and other paralinguistic areas is feature extraction, because our recordings vary in length, but classification/regression techniques require fixed-sized fea-

ture vectors. This particular feature extraction approach shares similarities with the Bag-of-Visual-Words (BoVW) image analysis technique and the Bag-of-Words (BoW) text preprocessing technique.

Several methods have already been developed to address the problem of varying lengths and to standardise the features extracted from the recordings to a uniform length. For example, the hand-crafted features (like ComParE, intonation, volume contours, jitter and shimmer values, etc.) with statistical aggregation [53, 73], the GMM supervectors [8], i-vectors [12, 29, 89, 104] and Fisher vectors [21].

BoAW representations have been used in various paralinguistic tasks as well. For example, it has consistently demonstrated competitive performance across Interspeech challenges, often serving as one of the primary baseline methods in case of several aspects, like recognise COVID-19 infection, classify the level of escalation in human dialogues, differentiate four species of primates versus background noise [80], recognise the emotion of elderly people, predict breathing patterns provide medical insights, tell apart whether a speaker wears a surgical mask or not [81], indicate speech dialects, detect the level of sleepiness, recognise five types of baby cries, differentiate between orca sounds to understand their communication [82], classifying emotions in disabled speech, detecting severities of heartbeats [79], classifying child-directed versus adult-directed speech, recognising speech affected by illness (cold), identifying snore types by origin [74].

3.3 The Bag-of-Audio-Words Technique

As an overview, Bag-of-Audio-Words (BoAW) first performs an analysis on the entire audio database and then, based on the results obtained, generates statistical info for each file separately. This info represents the relationship of the audio to the entire database, and it will be the fixed-sized feature vector. By employing this feature representation, the issue of variable length, as previously discussed, can be effectively addressed. The BoAW technique provides a compact and informative representation of audio signals, ensuring efficient analysis of audio data using machine learning algorithms.

Figure 3.1 provides a more detailed, step-by-step introduction of the BoAW technique. It have some differences wether we are talking about a training or a test utterance. This method has various steps:

1. The initial step involves frame-level feature extraction for each recording. This step leads to varying numbers of feature vectors for each recording, where the quantity is determined by the original length of the audio and the windowing size. Generally, we use the same hand-crafted features for both train and test sets, like MFCCs.

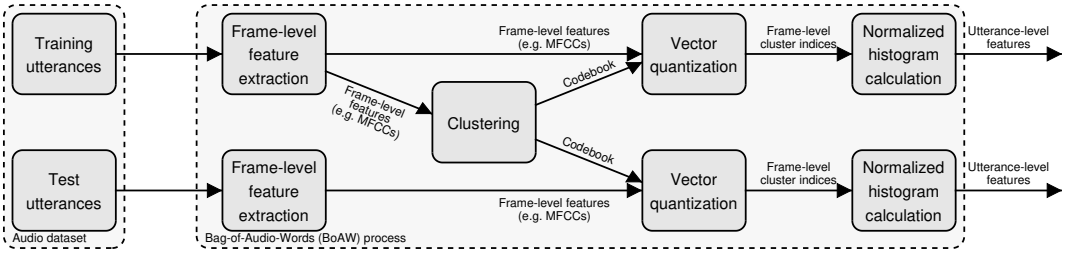

Figure 3.1: *Workflow for the Bag-of-Audio-Words technique.*

2. When we have all the frame-level feature vectors from all the recordings, the next step is clustering. The feature vectors belonging to the train set are collected into a unified ”bag” and a clustering algorithm (e.g: k-means) is applied on them. This step creates a specific number of groups from the vectors. Each cluster is represented by its centroid vector (i,e: ”codeword”). Then, a ”codebook” is formed using the collection of ”codewords”. This step gives a reliable description of the acoustic patterns found in the training audio data. The purpose is to organise the frame-level vectors into subsets where vectors within the same group have greater similarity to each other than to vectors in different groups. After we have this ”codebook”, we will use it for further analysis.

3. The next step is the vector quantisation, where we assign the original frame-level vectors to their nearest ”codewords” based on a distance metric such as Euclidean distance. We are using the same ”codebook” for quantising booth train and test samples.

4. To construct the final BoAW representation for an audio, we create a histogram vector, where the length of this vector equals the number of ”codewords”. We go through the quantised vectors of the original recording, and for each vector, we check the index of the ”codeword” and increase our new vector by one at the same index. The distribution of codewords across the audio forms a histogram-like representation (shown in Figure 3.2), known as the utterance-level BoAW feature vector.

5. In the last step, this histogram is normalised by dividing each counter value by the original number of frames in the recording. This normalisation process ensures that the resulting histogram accurately represents the relative frequencies within the audio segment.

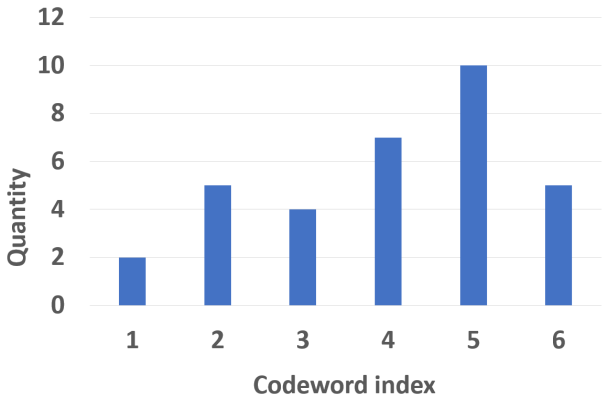

Figure 3.2: *The Bag-of-Audio-Words histogram of an audio recording.*

3.4 Experiments

The effectiveness of the Bag-of-Audio-Words methodology depends on the careful optimisation of its hyperparameters and the architectural decisions made during its implementation. While it provides an elegant solution to the variable-length utterance problem inherent in computational paralinguistics, its performance is highly sensitive to parameter configurations. This behaviour is systematically investigated and optimised in the following thesis works.

Our experimental investigation addresses three critical research questions that emerged from our theoretical analysis of BoAW:

1. *How do individual parameter choices affect classification/regression performance, and can we establish optimal parameter ranges?*

   Firstly, parameter optimisation experiments highlighted the significant impact of proper parameter optimisation. The preprocessing methods, codebook size, quantisation neighbour count, clustering algorithm choice, and the utilisation of delta features have a high impact on classification performance.

2. *To what extent is the BoAW approach dependent on corpus-specific characteristics, and can codebooks be transferred across different datasets?*

   Secondly, we challenged the conventional approach of corpus-dependent codebook generation by investigating the corpus independence. Our experiments demonstrated that codebooks created on one database could be reused effectively across different datasets.

3. *How does the inherent stochastic nature of the clustering process affect result reliability, and what strategies can mitigate this variability?*

Lastly, we examined the stochastic behaviour of the method, a consequence of its reliance on randomised clustering initialisations. We found that repeated runs of the BoAW feature extraction can produce variable results, potentially undermining reliability.

Each investigation follows a systematic experimental design, with controlled manipulation of specific variables while maintaining consistency across other parameters. This approach allows us to isolate the effects of individual design choices and build a comprehensive understanding of the Bag-of-Audio-Words method's behaviour across different scenarios. Each experiment was built upon the insights gained from the other. The first two experiments utilise a classification. The third experiment explores the behaviour through regression. This diverse and iterative approach allows to establish increasingly sophisticated guidelines for BoAW implementation while ensuring that our recommendations are both theoretically aus nd practically applicable across diverse paralinguistic tasks. The cumulative insights from these three investigations form the basis for our practical guidelines and recommendations for BoAW implementation in computational paralinguistics applications.

3.4.1 Parameter Optimisation

Thesis Point I/1. - Using the Bag-of-Audio-Words approach for emotion recognition [94].

The BoAW method has many adjustable parameters that affect the quality of the extracted utterance-level feature vectors, and furthermore affect the efficiency of the final classification or regression as well. In our first research, we investigated how changing the settings of the parameters in an emotion detection task affects the final result. The primary aim of this study was to analyse the effectiveness of the Bag-of-Audio-Words method and identify the most suitable parameter set for emotion recognition. We iteratively optimised the parameters, taking into account the findings from each iteration.

Database

In each experiment, we trained and evaluated our classification model on the Hungarian emotion database discussed in section 2.3.1. Earlier studies working with the same database were able to achieve a classification accuracy score of 66–70% [21, 87, 87].

Frame-Level Features

The frame-level feature set was the ComParE features with delta values, described in section 2.3.1. For feature extraction, we used the open-source openSMILE [16] software package with the $IS13$ ComParE config file.

Utterance-Level Features

Our utterance-level feature set was created with the Bag-of-Audio-Words techniques. Therefore, we created two codebooks in parallel (one for 65 frame-level features and one for their Δvalues). Because of this, the codebook sizes given in this section, the indicated codebook sizes have to be multiplied by 2 to get the number of features used. For the codebook building, we used an open-source program called openXBOW [71].

Classification

In the end, the classification was performed with an SVM, introduced in section 2.3.3. It was implemented with the help of the LIBSVM library [9]. The C complexity parameter was tested in the range 10^{-5} to 10^{0}. In the optimisation configurations the following powers of 10 were used: -5; -4; -3; -2; -1 and 0. In the optimisation part of our experiments, we worked with the training set, based on 10-fold cross-validation. The cross-validation methodology is introduced in section 2.3.2. In the test scenario, we trained a model on the whole training set with the optimal C parameter value found above and evaluated it on the test set. As an evaluation metric, we used the Unweighted Average Recall, introduced in section 2.3.4. The reason we use this metric, that we have imbalanced classes.

Results

In our experiments, we tested the effect of the following parameters and design choices:

1. preprocessing method (normalisation and standardisation
2. the quantisation neighbour number (5 and 10)
3. clustering method (k-means and k-means++)
4. resampling method (upsampling)
5. the effect of Δvalues
6. the codebook size N (32, 64, 128, 256, 512, $1\,024$, $2\,048$, $4\,096$, $8\,192$)

Feature-preprocessing	Maximum UAR	Codebook size
No preprocessing	36.32%	8 192
Normalization	46.73%	4 096
Standardization	45.42%	1 024

Table 3.1: *Preprocessing: The best results were obtained without preprocessing, with normalisation and standardisation, when we evaluated our technique with cross-validation.*

The first option, an architectural decision we investigated, was the preprocessing technique, where we tested two types: normalisation and standardisation. Preprocessing is always a good choice because databases contain outliers, which have a detrimental effect on learning effectiveness. From our results in Table 3.1, it is apparent that the data without preprocessing proved to be the weakest in all cases. By comparison, normalisation and standardisation gave performance improvements that were nearly the same. Another advantage of normalising or standardising the input is that significantly fewer clusters are required for optimal performance than leaving the input unchanged ($8\,192$), so we found that in both normalisation and standardisation, a size of $1\,024$ was big enough to achieve the best performance. This lower codebook size also helps the performance of the SVM, because in a smaller feature space, the speed and success of the learning will also increase.

The second option, a parameter we investigated how many closest codewords have to be assigned to a frame-level feature vector when creating a histogram to achieve the optimal performance. We tested two setups: 5 and 10 neighbours. Based on the results of our previous optimisation, all three quantisation options were also evaluated with normalisation and standardisation. From our new results in Table 3.2, we may conclude that more than one neighbour gives better results in the majority of cases. This can be seen for both preprocessing techniques (normalisation and

Feature-preprocessing	a	Maximum UAR	Codebook size
Normalization	1	46.73%	4 096
	5	48.93%	4 096
	10	49.14%	16 384
Standardization	1	45.42%	1 024
	5	46.16%	8 192
	10	47.37%	8 192

Table 3.2: *Number of neighbours: The best results obtained for* $1, 5, 10$ *neighbours with normalisation and standardisation.*

Clustering-algorithm	Feature preprocessing	a	Maximum UAR	Codebook size
k-means	Normalization	5	48.93%	4 096
		10	49.14%	16 384
k-means	Standardization	5	46.16%	8 192
		10	47.37%	8 192
k-means++	Normalization	5	50.94%	4 096
		10	47.77%	4 096
k-means++	Standardization	5	50.08%	8 192
		10	47.74%	4 096

Table 3.3: *Clustering algorithm: The best results for k-means and k-means++ algorithms with cross-validation.*

standardisation). There was no significant difference between the $a = 5$ and $a = 10$ values.

The third option, a parameter we investigated, was the clustering algorithm, where we tested two techniques: k-means and k-means++. Based on our earlier results, we decided to test them with normalisation and standardisation, and with 5 and 10 neighbours in the quantification step. Based on the results in Table 3.3, we can say that both clustering methods have the same trend. Since we did not find any significant difference between the trends of k-means and k-means++, so other tests were performed using the k-means algorithm.

The fourth option, an architectural choice we investigated, was the upsampling method. Because our database is relatively small and has imbalanced classes, we decided to use upsampling on our samples as utterance-level BoAW features. In this scenario, we tested how upsampling affects our results. Based on the results in Table 3.4, we can state that upsampling gave an improvement of about 10% compared to all of our previous results.

In the last optimising scenario, we tested the effect of using Δvalues. From our results in Table 3.5, we can conclude that Δ values can help to reduce the number

Feature-preprocessing	a	Maximum UAR	Codebook size
Normalization	5	58.88%	2 048
	10	60.42%	256
Standardization	5	55.93%	128
	10	58.59%	1 024

Table 3.4: *Upsampling: The best results obtained with upsampling in cross-validation training.*

Feature-preprocessing	a	Maximum UAR	Codebook size
Normalization	5	58.63%	512
	10	57.48%	512
Standardization	5	56.08%	512
	10	59.00%	512

Table 3.5: *Deltas: The best results of cross-validation using deltas.*

of necessary codewords to a moderate size. Another advantage is that performance trends are less hectic than before and much more predictable.

Summary of Guidelines

Our results indicated an increasing performance in emotion classification. Precise tuning of multiple parameters was necessary to achieve optimal efficiency. We are provided clear recommendations on parameter configurations that might be helpful when using the BoAW technique:

- Transform the input dataset to the same scale by normalisation or standardisation. Preprocessing is always a good choice.

- For greater generalisation ability, it is worth including more neighbours in the quantising step, such as 5 or 10.

- It is worth choosing the size of codebook from a medium-large range (e.g. between 128 and $4\,096$). If possible, try to keep the codebook size low to get a better generalisation. The connection between increasing codeword quantities and decreasing model performance also seems to suggest that the larger the codebook size we choose, the greater the chance of over-fitting.

- Clustering with the k-means or with the k-means++ algorithms could be equally good.

- By balancing the frequency of classes seen during learning, we can improve our generalisation ability. Upsampling can help to achieve it.

- We should calculate and use Δvalues. With that, we can reduce the number of necessary codewords to a moderate size, and the training trends are less random than before and much more predictable.

3.4.2 Corpus Independence

Thesis Point I/2. - Investigating the Corpus Independence of the Bag-of-Audio-Words Approach [92].

In our previous paper, we investigated the influence of model parameters and feature preprocessing in a corpus-dependent environment, but this raises the question of topic independence. In our next research, we focus on a comprehensive corpus-independence analysis of the Bag-of-Audio-Words feature extraction method. The initial clustering step is typically corpus-dependent and performed on the training set of the investigated database. However, this approach has limitations, such as the need to create new codewords for each dataset, resulting in increased computational time and potential overfitting. In this paper, we discuss the effect of using a predefined codebook. We address the question of whether a codebook from another database can produce similar or better results than using a codebook from the original database.

Database

Our experiments consist of three databases: the Hungarian emotion database, the EmoDB, and the BEA speech database. All of their details are discussed in section 2.3.1. We experimented with constructing codebooks on each of these databases. Then, different BoAW representations were created for the Hungarian database with the help of these different codebooks. In each experiment, we trained and evaluated our classification model on these representations. We examined how the classification accuracy scores vary based on the codebook used.

Frame-Level Features

The frame-level feature set was the ComParE features with delta values, described in section 2.3.1. We used the open-source openSMILE [16] feature extractor, with the $IS13$ ComParE config file.

Utterance-Level Features

Our utterance-level features were created with the BoAW technique. For the codebook building, we used an open-source program called openXBOW [71]. We applied standardisation to the BoAW feature vectors before the classification model was trained. Because we extracted 2×65 features from each recording, we created two codebooks in parallel: one for 65 frame-level features and one for their derivatives. Therefore, the codebook sizes given in the results have to be multiplied by 2 to get the number of features actually used.

Classification

In the end, the classification was performed with an SVM, introduced in section 2.3.3. It was implemented with the help of the LIBSVM library [9]. We optimised the C complexity parameter of the SVM in the range of 10^{-5} to 10^{0}. In the optimisation part of our experiments, we worked with the training set, based on speaker-independent 10-fold cross-validation. The cross-validation methodology is introduced in section 2.3.2. In the test scenario, we trained one SVM model on the whole training set with the optimal C parameter found above and evaluated it on the test set. As an evaluation metric, we used the Unweighted Average Recall, introduced in section 2.3.4. The reason we use this metric, that we have imbalanced classes.

Results

Based on the best practices from our previous study [94], we tested the corpus independence within a wider parameter set to get a comprehensive view. In our experiments, we tested the effect of the following parameters and design choices:

1. two feature preprocessing methods (normalisation and standardisation)
2. the quantisation neighbour number (5 and 10)
3. the codebook size N (32, 64, 128, 256, 512, $1\,024$, $2\,048$)

As the baseline, we conducted a model evaluation with a traditional BoAW feature set on the Hungarian emotion database. The codebook was created from the training set of the Hungarian emotion database. Then, calculated features based on this codebook. Finally, evaluated a classification model with 10-fold cross-validation.

In the first case, we wanted to know whether working with a codebook from another database could produce similar or better results than a codebook created

Feature-transformation	**a**	**UAR**		**Codebook size**
		CV	**Test**	
Normalization	5	59.52%	70.07%	1 024
	10	60.13%	62.70%	256
Standardization	5	57.34%	66.59%	128
	10	58.81%	70.70%	256

Table 3.6: *EMODB: best results with normalisation and standardisation and $5/10$ quantisation neighbours, when we evaluate our technique with cross-validation and do it on the test set.*

Database	Feature-transformation	a	UAR		Codebook size
			CV	Test	
1-hour news	Normalization	5	56.48%	62.77%	512
		10	58.55%	65.86%	1 024
	Standardization	5	60.74%	67.29%	1 024
		10	58.82%	69.48%	1 024
2-hour news	Normalization	5	57.08%	70.17%	1 024
		10	57.11%	56.53%	32
	Standardization	5	57.16%	66.21%	2 048
		10	58.41%	63.62%	2 048
5-hour news	Normalization	5	57.67%	61.61%	2 048
		10	59.80%	66.33%	1 024
	Standardization	5	55.75%	65.82%	128
		10	56.54%	64.79%	2 048
10-hour news	Normalization	5	58.51%	62.04%	2 048
		10	58.13%	67.47%	1 024
	Standardization	5	59.05%	65.72%	1 024
		10	58.27%	71.86%	1 024

Table 3.7: *News: best results with normalisation and with standardisation, when we evaluate our technique using cross-validation and a test set.*

from the original database. It is important, the booth of the databases was made for the same paralinguistic use-cases. In this part, the codebooks were created from *EmoDB*. Then, we built a BoAW representation for the Hungarian emotion database and performed classification using these features.

Based on the results shown in Table 3.6, it is apparent that a codebook created from a different database led to significant improvements. Since the main step in creating a codebook is an unsupervised clustering, the question arises as to whether it can affect the success of the classification if the database used to create the codebook was designed for a different purpose than the database which was used in the classification step. So, for the next part, the new codebooks were prepared from subsets of the above-mentioned non-emotion Hungarian television recordings database.

Thereafter, with these predefined codebooks, we created the BoAW representation of the Hungarian emotion database. We examined four cases to determine whether the length of the database used affects the performance of the classifier or not. An analysis was performed for 1-hour, 2-hour, 5-hour, and 10-hour. The values obtained

are shown in Table 3.7. The results from the test set shown were very similar to our previous results with the *EmoDB* codebooks. Using a codebook from a different database always improved the cross-validation compared to the *Baseline* (except in one case where there was a slight(0.58%) reduction in the performance for the 2-hour database, with 10 neighbours and normalisation).

Summary of Guidelines

The findings indicate similar classification performance across all cases, suggesting that the Bag-of-Audio-Words codebooks have practical corpus independence. Based on our tests, it can be clearly stated that each predefined codebook can be successfully used to extract BoAW feature representations of another database. The baseline result with the Hungarian emotion database's own codebook was 64.32%. Compared to this, when we used other database codebooks, we got better results. The best score of the tests with the German emotion database codebook was $66 - 70.70\%$. The best score of the tests with the Hungarian speech database codebook was $66 - 71.86\%$. With these results, we could not find a clear answer to whether it is advisable to use a codebook between any two databases created for similar purposes but a different language or for a similar language but different purpose, so we can not draw any clear conclusion about whether it is good to proceed from one database to another. In both cases, our results varied on a similar scale, with no significant difference. This corpus-independence aspect enables the reuse of codebooks generated from different datasets, thereby facilitating the practical implementation of the BoAW method.

3.4.3 Ensembling Strategies

Thesis Point I/3. - Handling the stochastic behaviour of the Bag-of-Audio-Words method [93].

Our previous experiments showed there are general parameter settings and codebooks that can be reused in the case of emotion recognition. However, the BoAW method contains non-consistent calculations, including random numbers, so it can give us slightly different features for each re-run, even when we use the same settings. It can heavily influence the quality of the extracted features. Due to this, we decided to investigate how this behaviour affects the global parameter settings and the performance. We wanted to focus on this effect only, so we did not use cross-corpus codebooks.

Database

In each experiment, we trained and evaluated our classification model on the Slepiness database discussed in section 2.3.1. Previous results for this database produced scores between $.260$-$.383$ [25, 82].

Frame-Level Features

The frame-level feature set was the ComParE features with delta values, described in section 2.3.1. We used the open-source openSMILE [16] feature extractor, with the $IS13$ ComParE config file.

Utterance-Level Features

Our utterance-level features were created with the BoAW technique. For the codebook building, we used an open-source program called openXBOW [71].

Regression

In the end, we have to predict the factor of sleepiness on a scale from 1 to 10. Therefore, we trained a regression model for prediction calculation. The regression was performed using an Support Vector Regression Machine, discussed in section 2.3.3. It was implemented with the help of the LIBSVM library [9]. In the training scenario we evaluated it with multiple C complexities in the range 10^{-5} to 10^{-3}. For the evaluation, we used Pearson and Spearman correlation, introduced in section 2.3.4.

Results

Based on the best practices from our previous studies [91, 94] and to test the stochastic behaviour, we made the following architectural and parametric decisions:

- used standardisation as a feature preprocessing methods
- used the k-means++ algorithm for clustering
- counted on the 5 closest neighbours at quantisation
- the codebook size were tested from 32 to $8\,192$
- use different random seeds for clustering

In order to test the robustness of the BoAW algorithm, we ran the feature extraction 10 times with 10 different start-up random seeds and ran our experiments on that pool. The following seeds were used: 1964, 423, 1355, 86, 1052, 1549, 139, 731, 951. This showed us the deviation of the output. We tried to handle the stochasticity with three different ensembling methodologies:

- Model performance ensembling - Average models: From the different feature sets, we built 10 different models and calculated their final performance value (correlation), then took the average of the correlations.

- Prediction ensembling - Average predictions: From the different feature sets, we built 10 different models and calculated predictions, then took the average in case of each sample.

- Feature ensembling: We calculated the average feature for each sample, and then we built one model on it. The drawback of this is that our feature space is now 10 times larger. In the case of a codebook size of 32×2 it will be 640, 64×2 it will be $1\,280$ and so on. Unfortunately, it has a major drawback in the computing time and the memory used. As we wished to reduce these effects, we ran a Principal Component Analysis (PCA) on the concatenated feature data. It is a dimension-reduction method, and it projects the original feature set into a lower-dimensional space by reducing the number of features. In our investigation, we decided to keep 95% and 99% of the original information.

Our main results are shown in Table 3.8. As a baseline, the standalone performances of the 10 different feature sets were calculated. The best case, the worst case and a randomly selected case are highlighted in the first three rows. The differences between the best model and the worst model predictions may be up to $.033$ on the dev set and $.03$ on the test set, depending on the codebook size. This difference tells us that the BoAW algorithm should be more robust, because if we made only one feature extraction, we may just get an unbiased, edge result.

	Pearson			**Spearman**		
	dev	**test**	***codebook size***	**dev**	**test**	***codebook size***
Best model	.334	.382	8 192	.341	.369	8 192
Worst model	.325	.379	4 096	.326	.373	8 192
Random model	.329	.375	4 096	.331	.363	4 096
Average model	.328	.378	4 096	.331	.369	8 192
Prediction average	**.331**	**.382**	4 096	**.336**	**.373**	8 192
PCA 95	.322	.354	3 319	.322	.328	4 036
PCA 99	.322	.353	4 780	.320	.341	4 780
Challenge original BoAW	–	–	–	.269	.260	2 000
Challenge original best one	–	–	–	.367	.383	–

Table 3.8: *Best results with the Pearson/Spearman correlation*

			Pearson				Spearman			
Concatenated	**Reduced size**		**PCA 95%**		**PCA 99%**		**PCA 95%**		**PCA 99%**	
codebook size	**95 %**	**99 %**	**dev**	**test**	**dev**	**test**	**dev**	**test**	**dev**	**test**
640	165	379	.291	.331	.293	.335	.291	.323	.293	.328
1 280	331	753	.302	.337	.302	.338	.297	.332	.297	.333
2 560	700	1 482	.311	.355	.311	.356	.309	.346	.309	.346
5 120	1 369	2 661	.314	.362	.314	.362	.316	.351	.317	.351
10 240	2 337	3 968	.317	.364	.317	.365	.316	.350	.317	.350
20 480	3 319	4 780	.322	.354	.322	.353	.321	.344	.320	.341
40 960	4 036	5 132	.318	.339	.317	.336	.322	.328	.320	.323
81 920	4 473	5 283	.302	.328	.303	.322	.303	.313	.303	.309

Table 3.9: *PCA results*

If we are comparing the Average model and the Average prediction techniques, we can see that the prediction ensembling gave slightly better results by $.003$ and $.004$ on the dev and test sets.

For comparison, we also show the results of the original Interspeech challenge. It can be seen that our method gives better results than the challenge original BoAW baseline, and it is close to the best submission with $.336$ on the dev set and $.373$ on the test set.d

When we are analysing our results of Feature ensembling with PCA, on one hand, we can see a performance drop, but on the other hand, we were able to decrease the size of the codebook, which gives us an improvement in the computational load.

Table 3.9 shows the Feature ensembling results in more detail. The second and third columns contain the number of dimensions after the PCA transformation. As we can see, there was a slight loss in efficiency with the 95% and the 99% compression. An interesting pattern can be seen in the test results. The curve of the test set does not follow the curve of the development set. The concatenated features that were reduced to a third or a quarter performed better. We think that this is due to a drawback of the PCA method. PCA finds linear combinations of the features, but sometimes it fails when the number of features is equal or even larger than the size of the database. As we can see in Table 3.9, when the concatenated codebook size was more than twice the count of our recordings ($\approx 5500 \times 2$), the test results started to decrease.

Summary of Guidelines

Overall, we found that the usage of the Bag-of-Audio-Words technique can also perform effectively in regression. This approach not only captures essential acoustic

information but also proves adaptable for various paralinguistic analyses. Although it has a stochastic behaviour, we can overcome this problem with prediction ensembling or feature ensembling. The best results came from prediction ensembling, but it requires higher-dimensional feature spaces. On the other hand, we can have good results with feature ensembling plus PCA as well, while decreasing the feature space. PCA ensures computational efficiency and maintains the original information, but before applying it, we have to make sure that the original number of features is lower than the number of data samples.

3.5 Concluding Remarks

In this chapter, we conducted an in-depth analysis of the Bag-of-Audio-Words technique as a foundational traditional feature extraction method in various computational paralinguistics tasks (emotion recognition and speaker identification). Our systematic investigations addressed critical challenges, including parameter optimisation, corpus independence, and the stochastic nature inherent in clustering computations.

- The first investigation (Parameter Optimisation in section3.4.1) establishes the foundation by systematically exploring the impact of individual BoAW parameters on emotion classification performance. We examine feature preprocessing options (normalisation, standardisation, delta feature computation), clustering parameters (codebook size, algorithm selection, quantisation strategies), and class balancing techniques. This investigation aims to identify optimal parameter ranges and establish clear guidelines for a traditional method.

  Proper parameter optimisation has a high impact on the performance of BoAW technique, and we were able to define gold standards to narrow the pool of possible hyperparameters. We identified that normalisation or standardisation of input features consistently leads to better generalisation and higher classification accuracy, while upsampling helps mitigate class imbalance effects. Optimal codebook sizes generally lie between 128 and 4096 clusters, balancing detail capture and computational feasibility. Furthermore, assigning multiple neighbours (5 or 10) during quantisation enhances the representational precision of the BoAW histograms. These findings provide clear practical guidelines that researchers and practitioners can follow to achieve robust BoAW-based paralinguistic models.

- Building on the first findings, the second investigation (Corpus Independence in section3.4.2) challenges the traditional assumption that BoAW codebooks must be generated from task-specific training data. We evaluate the transferability of codebooks across different corpora, examining whether general-purpose

or cross-domain codebooks can achieve comparable performance to corpus-specific approaches. This investigation has significant practical implications for resource-constrained applications and rapid prototyping scenarios if we can use already-made cross-corpus codebooks.

We are proving that the BoAW feature extraction method can be applied across different datasets without requiring dataset-specific parameter optimisation. Our experiments demonstrated that codebooks created on one database could be reused effectively across different datasets without significant performance loss, and even improve classification results by reducing overfitting. The ability to reuse existing codebooks substantially reduces the need for computationally expensive retraining on specific corpora and facilitates faster prototyping and deployment in real-world applications. This insight is particularly valuable given the scarcity and cost of collecting large, labelled paralinguistic corpora.

- The third investigation (Ensemble Strategies in section3.4.3) addresses another aspect, the impact of stochastic variability inherent in the clustering process. We propose and evaluate the ensembling strategy to improve result stability and reliability, incorporating dimensionality reduction techniques to manage the computational overhead of ensemble approaches.

  We demonstrated that different ensembling strategies can increase robustness and stabilise the regression performance. On the other hand, feature ensembling exponentially increases the feature dimensionality, posing computational challenges. We mitigated this issue using Principal Component Analysis for dimensionality reduction, which preserved classification performance while drastically lowering feature space size. Nonetheless, the effectiveness of PCA depends on the relationship between feature dimensionality and sample size, suggesting caution when applying it under extreme conditions.

In summary, the Bag-of-Audio-Words remains a competitive and practical feature extraction method for low-resource environments, especially in scenarios constrained by limited data or computational resources. The contributions in establishing golden standards for parameter settings, demonstrating corpus-independent applicability, and presenting strategies to control the stochastic nature of BoAW through ensembling and PCA serve as a valuable reference for both academic research and industrial applications.

Chapter 4

HMM/DNN as a Hybrid Feature Extraction Method

4.1 Chapter Overview

The following research direction aims to investigate beyond traditional machine learning methods and explore hybrid methodologies that combine both traditional and modern approaches. Although the previous findings demonstrate the strengths and practical guidelines of traditional techniques (like Bag-of-Audio-Words), the field of computational paralinguistics could greatly benefit from investigating how hybrid models (such as HMM/DNN) might address persistent challenges.

Hidden Markov Models, were once at the forefront of automatic speech recognition. It was built for modelling probabilistic sequences of observations. In the foundational period of HMM-based speech recognition, systems employed discrete probability distributions and were primarily designed for isolated word recognition with small vocabularies in speaker-dependent configurations. After a while, the integration of Gaussian Mixture Models into HMM-based speech recognition systems proved to be more effective for larger vocabulary recognition tasks. After DNNs were raised, HMM/GMM models evolved into HMM/DNN hybrids, replacing the GMM module with a DNN. It was able to process complex relationships in data. The efficient training and utilisation of HMM/DNN hybrids soon became widely adopted. [4, 19, 58]

While this method gains more interest in the field of paralinguistics as well, its initial training raises problems. This hybrid solution presents challenges in architecture design, computational management, and hyperparameter tuning. Hidden Markov Models were originally designed for frame-level classification. Otherwise, computational paralinguistics requires utterance-level classification/regression and typically has a small corpora. This drawback, combined with the fact that Deep Neural Network (DNN)s require a massive amount of data, makes them more challenging to apply in paralinguistics. [22, 58]

Section 4.2 summarises the related works and briefly presents previous research results with HMM/DNN models. Section 4.3 introduce the HMM/DNN architecture. Thesis Work II/1 is covered in Section 4.4. Results and final thoughts are summarised in Section 4.5.

4.2 Related Works

In recent times, Recurrent Neural Network structures with components like Long-Short Term Memory [36] and Gated Recurrent Unit [11] have been recognised as the benchmark in Automatic Speech Recognition. However, various studies suggest that the HMM/DNN model outperforms these networks. Key factors toward the hybrid model are that it includes simpler training processes, reduces computational demands, and has smaller memory requirements. In addition, non-recurrent neural networks have demonstrated competitive performance when the quantity of training data is limited. Hybrid paralinguistic approaches merge the advantages of traditional statistical methods with those of modern machine learning techniques. [26, 35, 63, 70]

The original HMM-based speech recognition systems employed discrete probability distributions. In HMM/GMM architectures, each hidden state of the HMM is associated with a GMM. GMMs serve to model data distributions. The multi-modal nature of speech acoustics is better captured through Gaussian mixture distributions than through unimodal density functions. The HMM/GMM combination leverages the probabilistic nature of GMMs for modelling the emissions from hidden states in HMMs. These models have a localised GMM component to provide frame-level phonetic probability estimates, while the HMM component processes these into overall phone sequences. Compared to this, HMM/DNN models have an architectural challenge. Unlike GMMs, which are generative methods, DNNs are discriminative classifiers and they estimate different probability values. This issue can be addressed by applying Bayes' theorem. This synergy enhances the model's ability to capture intricate patterns in speech data, resulting in improved accuracy and robustness. Pre-trained neural networks are commonly used in the HMM/DNN hybrids. [4, 19, 22, 58] The HMM/DNN solution was applied to numerous use-cases, like emotion recognition [49], story segmentation [103], laughter event detection [22], social signal identification [26] and children's speech recognition [20].

4.3 The HMM/DNN Hybrid Technique

The HMM/DNN model has two parts. The first part is the Deep Neural Network, which excels in feature extraction and non-linear mapping. The second part is a

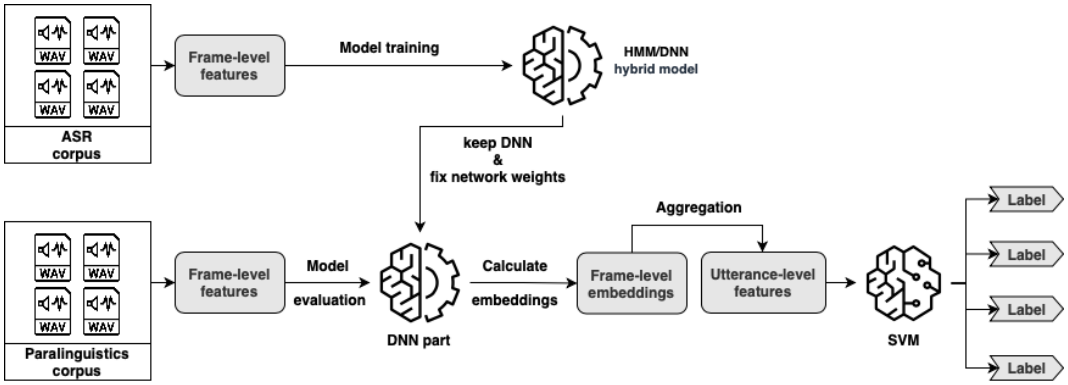

Figure 4.1: *Hybrid HMM/DNN Model Workflow for Paralinguistic Tasks.*

Hidden Markov Model. It handles temporal modelling. The outputs of the first DNN will be the input of the HMM.
The HMM expects a *class-condition likelihood*: $p(x_i \mid c_k)$,
but the DNN gives frame-level estimations, a *posterior probability*: $P(c_k \mid x_i)$.

Because of this, before utilising the output of the DNN into the HMM, we have to transform it. The transformation can be processed with Bayes' theorem. The *posterior* estimation should be divided by *a priori probabilities* of the phonetic classes ($P(c_k)$). Then we get the *class-condition likelihood* value within a scale factor. The a priori probabilities are usually estimated using simple statistical methods. The scale factor can be ignored because it does not influence the subsequent search process. This methodology can yield effective models for capturing complex paralinguistic features.

Figure 4.1 shows the complete flow of training and using a HMM/DNN model. First, we need to train our hybrid model. We can see the acoustic HMM/DNN model training in the top left corner of Figure 4.1. Here, we need a larger ASR corpus that has time-aligned phonetic labels. From this corpus, we have to extract frame-level features. The extraction can be handled using various techniques, such as calculating filter banks, deltas, spectrograms, or employing neural networks. We can utilise these frame-level features to train our hybrid model for a general language structure. Once the training of the hybrid model is complete, we need to make a slight modification to our model to utilise it for DNN embedding extraction. We have to detach the DNN from the hybrid model and fix its weights. In this case, we are no longer interested in the original output layer of our DNN, which produced the posterior estimates. Now we will focus on the output of the last few hidden layers, as they can provide more abstract information. Typically, this type of higher-level information can describe paralinguistic aspects. This is what we will refer to as frame-level embeddings. The process of embedding extraction is shown in the bottom left corner of Figure 4.1. First, we have to extract frame-level features from a paralinguistics

corpus. We have to use the same feature set as we used for the ASR corpus. Afterwards, we can feed them into the detached Deep Neural Network. Figure 4.1 shows the final utterance-level classification workflow in the bottom right corner. When we have the frame-level embeddings, we transform them into utterance-level feature vectors by aggregating them using a statistical function along the time axis. These utterance-level feature vectors can be used directly as input to any traditional classification or regression model, such as SVMs. The final output will be a label (class or real number) for each recording [58].

4.4 Experiments

4.4.1 Topic independence

Thesis Point II/1. - Using Hybrid HMM/DNN Embedding Extractor Models in Computational Paralinguistic Tasks [96].

In this study, we propose a method that combines Automatic Speech Recognition solutions with paralinguistic approaches to address the aforementioned challenges of computational paralinguistics. We trained a hybrid HMM/DNN acoustic model on a general ASR corpus, which allowed us to generate embeddings to serve as features for multiple paralinguistic tasks.

Database

On the one hand, we preferred databases that were easily accessible to the research community, and thus, the databases used in the ComPare challenge were chosen. Another aspect was that our results should be easily comparable with other research papers. We selected three German-language databases to minimise the potential impact of language differences on the results. To cover various paralinguistic topics, we utilised three paralinguistic corpora: AIBO, URTIC, and iHEARu-EAT. Although these corpora cover multiple topics, the recording conditions (such as sampling rate, language and background noise) are quite similar. The fourth database utilised in our experiments (called BEA) was used for training our hybrid acoustic model, and it is not a paralinguistic but a speech database. More details about these databases are discussed in section 2.3.1.

Frame-Level Features

For all four corpora, F-bank features were calculated with the HTK tool [102]. The final number of features in a frame-level vector was $41 * 3 = 123$. The technique

is discussed in section 2.3.1. For frame-level embedding extraction, we used the activation values of the middle five hidden layers (i.e, layer 1, 2, 3, 4, 5). Each layer generated 1024-sized frame-level feature vectors.

Utterance-Level Features

To transform frame-level feature vectors into utterance-level feature vectors, the statistical approaches used were the following: arithmetic mean, standard deviation, kurtosis, skewness, and zero ratio. The zero ratio represents how many times an output neuron fired (meaning a feature has a non-zero value in the embedding, as we used ReLU neurons). The final size of the aggregated vector is independent of the original recording's length. It depends only on the number of neurons in the given hidden layer and the chosen aggregation technique. More details about all of the approaches are discussed in section 2.3.2. These utterance-level feature vectors can be fed into any traditional classification or regression model. In the case of AIBO and URTIC, we always standardised and downsampled the actual training set before feeding it into the Support Vector Machine. In the case of iHEARu-EAT, we only performed a speaker-wise standardisation.

Classification

During the classification step, our classifier was a Support Vector Machine, introduced in section 2.3.3. We optimised the complexity parameter using 10 powers between 10^{-5} and 10^{0}. It was implemented with the help of the LIBSVM library [9]. For optimal results, we separated all paralinguistic corpora into train, development and test sets. We determined the optimal parameters of the classifier while training it on the training set and evaluating on the development set. After optimisation, we measured the overall efficiency while training on the combined train and development sets and evaluating with the test set. As an evaluation metric, we used the Unweighted Average Recall, introduced in section 2.3.4.

Results

A summary of our best results from the first series is represented in Table 4.1. More detailed results are visually shown in the published paper. As a baseline, we evaluated an X-Vector Neural Network on each dataset. The results are represented in the last row of this table.

In case of the AIBO database: Regarding the layers, we can state that the 4th layer consistently outperforms the baseline. Moreover, the 4th layer achieved the best performance scores with all the aggregation techniques used. In the view of

	AIBO			URTIC			iHEARu-EAT		
	Layer	DEV	TEST	Layer	DEV	TEST	Layer	DEV	TEST
mean	4	**45.2%**	**44.0%**	4	*67.3%*	*69.3%*	2	*71.4%*	*79.0%*
std	4	*44.8%*	*44.4%*	2	66.4%	68.1%	2	**73.3%**	**74.4%**
kurtosis	4	42.5%	40.3%	1	64.2%	60.8%	4	69.7%	69.0%
skewness	4	43.0%	41.2%	1	63.5%	68.3%	4	70.3%	67.3%
zero ratio	4	44.3%	42.1%	2	**67.4%**	**68.8%**	3	70.0%	75.5%
all	5	**45.5%**	**44.2%**	4	66.0%	65.3%	4	**76.6%**	**74.6%**
x-vector baseline	–	41.8%	35.6%	–	66.9%	57.1%	–	58.7%	53.8%

Table 4.1: *Results of different aggregation techniques with the three different corpora.*

aggregation, there were no significant differences between the robustness of aggregations. Kurtosis and skewness had the worst overall performance scores. The mean and standard deviation performed the best. In most cases, they outperformed the X-Vector baseline. The mean and standard deviation of the 4th and 5th layers achieved better performance scores than their average layer performance score, yielding the best results overall.

In case of the URTIC database: Here, we can state that the 3rd and 4th layers always reach or outperform the average performance of a conversion technique. But, in the majority of cases, they cannot beat the X-Vector baseline. The standard deviation is a bit more robust than the others, but again, there is no significant difference. The conversions of kurtosis and skewness statistics again had the worst performance scores. Here, the best results can beat the baseline. One of them is the mean of the 3rd and 4th layers. The other is the zero ratio statistic conversion with the 2nd and 4th layers.

In case of the iHEAR-uEAT database: Here, we can state that all of our embeddings consistently outperform the baseline. Similar to URTIC, the 2nd and 4th layers perform best and, in most cases, outperform the local average performance (represented by a black column). The robustness behaviour is similar, but the zero ratio and mean are slightly improved. The rest of the aggregations behave just like before. The mean and the standard deviation with the 2nd and 4th layers give the overall best results.

We can see that the HMM/DNN embeddings outperform the X-Vectors. The kurtosis and skewness aggregations perform the worst. The mean, standard deviation, and zero ratio techniques behave similarly.

We also wanted to explore the expressive power of the embeddings, so we began combining all five techniques as well. In the second series of experiments, we used Sequential Forward Selection (SFS) to combine multiple aggregated feature vectors.

This methodology is discussed in more detail in the section 2.3.2. To combine a subset of aggregations, we concatenated their utterance-level feature sets. The size of each utterance-level feature vectors was as follows: $1\,024$ as one technique, $2\,048$ as a concatenation of two different aggregated vectors, $3\,072$ as a concatenation of three different aggregated vectors, $4\,096$ as a concatenation of four different aggregated vectors and $5\,120$ when we concatenated all the different aggregated vectors. A summary of our results from the second series is given in Table 4.2.

In case of the AIBO database: With layer 4 features, all of the combinations perform better than their X-Vector baseline. With layer 5 features, all of the combinations performed better here as well. The combination of mean, skewness, standard deviation, and kurtosis yielded the best performance score, but we can achieve almost the same result without the kurtosis. The mean and standard deviation techniques always gave improvements. Although layer 5 had better performance scores than layer 4, we should also consider that calculating more than one aggregation requires more time and memory.

In case of the URTIC database: With layer 2 features, the best combination (zero ratio + mean + standard deviation) had the same performance score on the dev set, like the zero ratio only option. But the combination gave a better performance score on the test set. With layer 4 features, the first three combinations can outperform the X-Vector baseline. We can state that the 4th layer has the best generalisation if we use the combination of mean and zero ratio techniques. Kurtosis and skewness aggregations always underperform the others.

In case of the iHEAR-uEAT database: With layer 2 features, all combinations outperformed the baseline on both the development and test sets. In the case of combining four techniques, the zero ratio slightly improves our model and increases its ability to generalise. The best combination is std+kurtosis+zero ratio+mean. With layer 4 features, all of the combinations outperformed the baseline. When we combined three techniques (std + skewness + zero ratio), it slightly improved our model. We can state that a model trained on features from the 2nd layer can generalise better if we use the combination of mean, zero ratio and skewness techniques. The zero ratio always produces a good improvement.

Summary of Guidelines

For frame-wise computing, we followed standard ASR principles and utilised DNNs to perform frame-level feature extraction. Afterwards, to aggregate these features, we used more or less traditional computational paralinguistics techniques such as standard deviation and kurtosis. We found that combining the aggregation tech-

AIBO				URTIC				iHEARu-EAT			
Layer	Combination	DEV	TEST	Layer	Combination	DEV	TEST	Layer	Combination	DEV	TEST
4	mean	**45.2%**	**44.0%**	2	zero	**67.4%**	**68.8%**	2	std	73.3%	74.4%
4	me-ze	44.4%	42.0%	2	ze-me	67.1%	70.0%	2	st-ku	74.8%	75.9%
4	me-ze-st	44.4%	44.3%	2	ze-me-st	**67.4%**	**69.6%**	2	st-ku-ze	74.8%	78.9%
4	me-ze-st-ku	44.4%	44.5%	2	ze-me-st-sk	66.6%	69.4%	2	st-ku-ze-me	**74.9%**	**78.3%**
5	mean	44.5%	42.3%	4	zero	66.9%	67.1%	4	std	70.5%	73.8%
5	me-sk	44.3%	40.2%	4	ze-me	**67.7%**	**69.5%**	4	st-sk	74.9%	74.8%
5	me-sk-st	45.1%	43.7%	4	ze-me-st	67.6%	68.5%	4	st-sk-ze	**76.0%**	**75.0%**
5	me-sk-st-ku	**45.3%**	**44.2%**	4	ze-me-st-sk	66.8%	67.9%	4	st-sk-ze-me	**76.0%**	**74.3%**

Table 4.2: *The best results were obtained by doing SFS. The base aggregation and layers came from the best corpus-specific aggregations.*

niques effectively led to further improvements, depending on the task and the layer of the neural network from which the local embeddings were sourced. Based on our experimental findings, we conclude that our proposed method presents a competitive and resource-efficient approach for a wide range of computational paralinguistic tasks. By integrating ASR with paralinguistic techniques, we address the challenges of handling varying-length utterances and working with limited datasets.

In an overall conclusion, we can provide clear recommendations on parameter configurations that might be helpful when using the HMM/DNN hybrid technique:

- Extracting embeddings from the 4th layer always gives the best performance scores.
- Combining at least three aggregation techniques will always improve our results in any paralinguistic task. But we should carefully select the aggregation techniques used, as the best combination may be task-dependent.
- When choosing the exact number of aggregations to combine, taking into account Occam's razor principle.
- We should always consider including the mean, standard deviation and/or zero ratio in the combination.
- The ratio of non-zero activations as an aggregation function proved to be useful in combination. Mean and standard deviation consistently performed best, while non-traditional techniques can enhance their performance.

4.5 Concluding Remarks

In this chapter, we conducted an in-depth analysis of the HMM/DNN technique as a hybrid feature extraction method in various computational paralinguistics tasks

(emotion recognition, cold identification and eating monitoring). Our systematic investigations addressed critical challenges, including the careful selection of aggregation types.

In our first experiments, we tested five aggregation techniques individually. Our results indicate that the hybrid acoustic model performed better than X-Vector Neural Networks did. The mean, standard deviation and zero ratio techniques achieve practically the same performance scores.

After obtaining these results, we wanted to improve the expressive power of the embeddings. We chose to investigate the performance of combined aggregation techniques. We tested the possible combinations of the five methods using Sequential Forward Selection. Our results indicate that we successfully extracted features for different paralinguistic tasks using our HMM/DNN hybrid acoustic model-based feature extraction method. We can see that combining three techniques consistently improves our results in any paralinguistic task. On the other hand, the combination of four techniques will behave inconsistently. Although it improves the results on the development set, the results on the test set often decrease. For this reason, when choosing the number of aggregations to combine, it is worth considering Occam's razor principle, which states that unnecessarily complex models should not be preferred over simpler ones.

Using the 2nd or the 4th layer of the model is always a good choice. As for aggregations, the mean, standard deviation, and zero ratio always help improve performance, but we must combine these techniques carefully. In the case of kurtosis and skewness aggregations, we can observe varied behaviour in all databases. In the first stage, individually, they had the worst performance for each database and each layer. In the second stage of our research, in combination, they showed a similar tendency. They gave the lowest scores in 13 of 18 cases.

Our results indicate that our proposed method presents a competitive and resource-efficient approach for a wide range of computational paralinguistics tasks. The contributions in establishing golden standards for aggregation selection.

Chapter 5

Deep Neural Networks as State-of-the-Art Feature Extraction Models

5.1 Chapter Overview

Following the successful application of Deep Neural Networks to Automatic Speech Recognition, these methodologies have also gained increased attention in the field of computational paralinguistics [23, 43, 59]. Nowadays, the most effective solutions are integrating these State-of-the-Art models [10, 13]. These modern techniques can automatically learn complex representations directly from raw audio. It makes them able to capture intricate patterns and contextual relationships in speech data. Additionally, DNN approaches can provide benchmarks when we are using traditional and hybrid techniques in low-resource environments. In this chapter, the focus is on two State-of-the-Art Deep Neural Networks: the Sequence-to-Sequence Autoencoder (Seq. Autoencoder) model and the Wav-to-Vec 2.0 Neural Network (Wav2Vec 2.0) model. There are two main works with different core thesis points. The first research highlights the performance of an Seq. Autoencoder model in a cross-corpus environment. The primary focus is on the importance of audio preprocessing. The second study demonstrates the use of the Wav2Vec 2.0 model as a frame-level feature extractor and explores the effects of various aggregation strategies.

Structure of the chapter: Section 5.2 summarises the related works and briefly presents previous research results with Deep Neural Networks. Section 5.3 introduces the Seq. Autoencoder model architecture. Section 5.4 introduces the Wav2Vec 2.0 model architecture. The Work III/1 is covered in Section 5.5.1. The Work III/2 is covered in Section 5.5.2. Results and final thoughts are summarised in Section 3.5.

5.2 Related Works

Low-resource environments often make it unsuitable for using Deep Neural Networks, and deep learning methods are still in their early stages of development in computational paralinguistics. Traditional machine learning methodologies tend to perform better than large end-to-end DNNs. Deep learning methods have been shown to be more effective on raw features such as F-bank energies than hand-crafted attributes such as MFCCs. [57, 79, 81, 82, 88].

5.2.1 Embedding extraction in paralinguistic

There is a growing interest in general feature extractors that are non-specific to any paralinguistic tasks, such as X-Vectors, Seq. Autoencoders, Wav2Vec 2.0s and other neural networks. First, these methodologies were constructed for a specific purpose, but later they were employed as a frame-level feature extractor in computational paralinguistics [80, 99]. X-Vector technique was developed for speaker verification, but was later employed as a frame-level feature extractor in case of predicting the level of sleepiness [38], detecting Alzheimer's disease [64], and recognising emotions [69]. Seq. Autoencoder technique was developed for creating compressed and representative embeddings for data samples. It has a long history in machine learning, dating back to the 1990s. The basic idea is to train a neural network to reconstruct the input (not necessarily audio), while the network structure contains a small-sized *bottleneck layer*. Evaluating the fully trained network and using the activation values of the bottleneck layer leads to a compressed representation of the input. In computational paralinguistics, these representations can be used as frame-level features. Such techniques were successfully used in the past on various tasks like translation [52], acoustic event classification [2] and categorising the sounds of primates [80]. Wav2Vec 2.0 technique was developed to learn general speech feature representations by pre-training on large amounts of unlabeled speech data. It was also used as a frame-level feature extractor in other tasks like speaker recognition [50], stuttering detection [83] and emotion recognition [18]

The main advantage of the above-mentioned approaches is that they do not have to be trained on limited-sized speech data. The small size of paralinguistic datasets makes it difficult to train a feature extractor DNN model from scratch, so usually a standard ASR corpus is used for this purpose. Deep Neural Network embeddings can reduce the feature space dimension while preserving important information. It has been effective in capturing complex relationships in the data. In computational paralinguistics, these embeddings are representations of spoken language. They are a condensed and meaningful representation of the speech signal, capturing various paralinguistic cues. For example, they may encode information about the speaker's

emotional state, the speaker's identity, or the underlying prosodic characteristics of the speech. The advantage of embeddings in paralinguistics is that they can potentially capture more subtle and complex patterns that might be missed or difficult to model with handcrafted features [13, 50, 84].

On the other hand, DNN models require large amounts of labelled data for training and are computationally more demanding compared to traditional feature-based approaches. The training is done by passing speech audio recordings through a Deep Neural Network model. The model typically consists of multiple layers of interconnected neurons, which learn to extract high-level features and patterns from the raw audio input. After the training, the last few layers of the model are detached. The output of one of the internal layers, often called the bottleneck layer, is considered as the DNN embedding [13, 50, 84].

5.2.2 End-to-end systems in paralinguistic

Traditional paralinguistic systems often involve multiple stages of processing. As we presented before, the system might have separate modules for frame-level feature extraction, followed by an intermediate step to convert them to utterance-level feature vectors, and finally, a classifier or a regression model. These systems require engineering domain-specific features [29, 62].

In the context of paralinguistics, end-to-end Deep Neural Networks refer to neural network models that are used to directly map raw audio or speech signals to a specific paralinguistic attribute without the need for explicit feature extraction or intermediate processing steps. One of the most important aims is to streamline this process by directly learning complex representations from the raw audio and performing the target paralinguistic task in a single integrated model. The DNN is trained on a large unlabelled dataset, learning to extract high-level features and patterns from the audio itself, making it capable of implicitly capturing various paralinguistic cues. It's worth noting that while end-to-end DNNs have gained popularity in various speech-related tasks, they might not always outperform traditional feature-based systems, especially when domain-specific knowledge is crucial for the task at hand. The choice between an end-to-end DNN approach or a traditional system depends on the specific requirements, available resources, and the complexity of the paralinguistic task [30, 40, 81, 88]

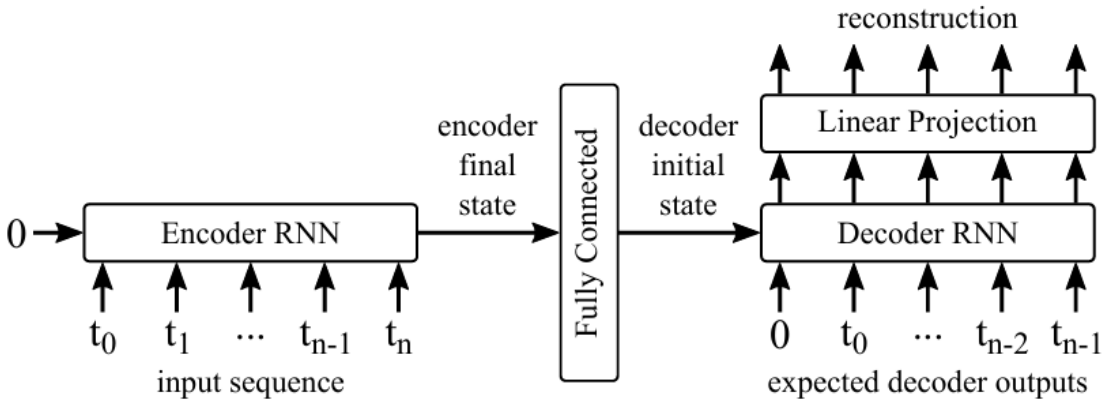

Figure 5.1: *An overview of the recurrent Seq. Autoencoder [2].*

5.3 Sequence-to-Sequence Autoencoder (Seq. Autoencoder)

Figure 5.1 shows the structure of the Seq. Autoencoder used in this chapter. In case of an Seq. Autoencoder frame-level features are fed into the *encoder* part of the neural network. This part consists, e.g. Long-Short Term Memory (LSTM) or Gated Recurrent Unit (GRU) cells in a recurrent manner over the time axis. The output of the fully connected bottleneck layer comes from the hidden states of the last cells. This will be the *encoded representation*. On top of the encoder network, another layer of LSTM or GRU cells is applied (i.e. the *decoder* part). It is expected to reconstruct the input frame-by-frame. Depending on the direction of this layer, the network can be unidirectional or bidirectional. The whole network is trained for input reconstruction, using the RMSE error function between the input vectors and decoder outputs. After training, we can detach the decoder. If the encoder is evaluated for a frame-level feature vector of an utterance, we will get the encoded representation. This might be used as the compressed form (or, in practice, as a feature vector) [2].

Although the Sequence-to-Sequence Autoencoders in theory can handle utterances with any duration, due to implementation constraints of `Tensorflow`-based toolkits, in practice, only objects with a limited duration could be processed.

5.4 Wav-to-Vec 2.0 Neural Network (Wav2Vec 2.0)

Figure 5.2 shows the structure of the Wav2Vec 2.0 model used in this chapter. The model has two main parts:

- a Convolutional Neural Network (CNN) block
- a BERT-based transformer block

In the first part, the raw input waveform is transformed into a sequence of high-level feature representations, known as the latent speech representation. The CNN incorporates ”dilation” between the filter weights, allowing it to capture information from a wider range of time steps in the input sequence without increasing the number of parameters.

Moving on to the second part, the output of the CNN is further processed into a sequence of high-level feature vectors, which capture the relationships between the input waveform and the extracted features. This part utilises a contextualised transformer architecture based on the widely used BERT model. The transformer consists of a multi-head self-attention mechanism and a position-wise feed-forward network.

The model can be trained with the cross-lingual representation (XLSR) learning approach, which involves two steps:

- pretraining the model by self-supervised learning on large unlabeled datasets of speech in different languages
- fine-tuning this model on a smaller labelled corpus with the target speech language (e.g. German)

In this way, the model learns to share discrete tokens across languages. The first pretraining step divides the input into small segments while applying random masking. The masking is done in a way that ensures that the model does not rely on any specific frequency components. Then we use the Contrastive Predictive Coding (CPC) approach, where the model is trained to distinguish between positive and negative pairs of examples. It has to maximise the similarity between different augmentations of the same input waveform (positive pairs) and minimise the similarity between different audio’s augmentations (negative pairs). In the second fine-tuning step, the original output layer is replaced with task-specific layers (typically with a recurrent neural network (RNN) and a Softmax layer). Then, the modified network is optimised with a Connectionist Temporal Classification (CTC) loss [3].

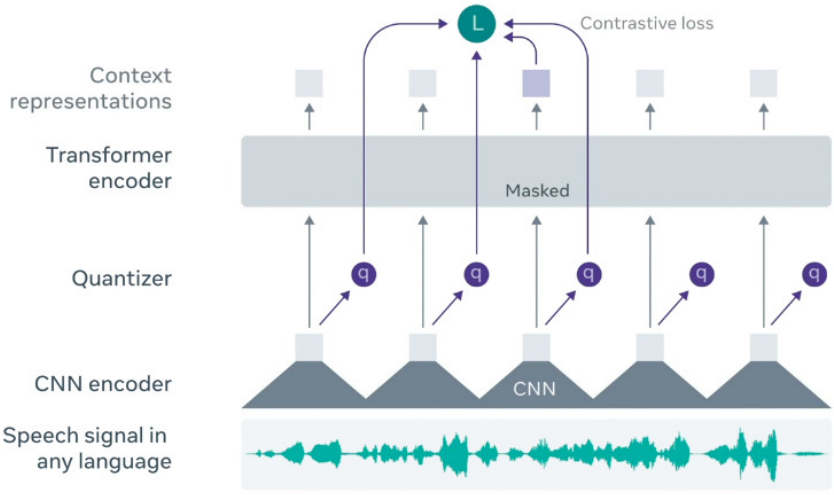

Figure 5.2: *The fine-tuned wav2vec 2.0 framework structure [3].*

5.5 Experiments

In this chapter, we conducted an extensive examination of State-of-the-Art Deep Neural Network models as feature extractors in computational paralinguistic tasks. Our experiments focused on two prominent architectures: the Sequence-to-Sequence Autoencoder and the Wav-to-Vec 2.0 Neural Network. While these techniques offer the potential to automatically learn complex representations directly from raw audio, their effectiveness are depends on the audio preprocessing techniques and the selected aggregation methodologies. These aspects are systematically investigated and optimised in the following thesis works:

- How do preprocessing decisions, aggregation strategies and cross-corpus training methods affect the reliability of Seq. Autoencoder?

  In this research, the experiments with different noise reduction thresholds demonstrate the crucial importance of audio preprocessing choices. Then we investigated how different aggregation techniques influence the classification performance in the case of Seq. Autoencoders. On the other hand, these experiments proved that cross-corpus training strategies can improve performance when feature extractors DNNs are trained on general ASR data rather than task-specific paralinguistic corpora.

- To what extent can aggregation strategies enhance the performance of Wav2Vec 2.0 embeddings, and can we establish optimal aggregation combinations across diverse paralinguistic tasks?

In the next research, a comprehensive experiment about diverse aggregation strategies demonstrates the importance of aggregation choices. Our results challenge the conventional choice of simple statistical measures. We are demonstrating that percentile-based aggregations and multi-technique combinations significantly outperform traditional approaches across various paralinguistic subtopics.

Each investigation follows a systematic experimental design, with controlled manipulation of specific variables while maintaining consistency across model architectures. This approach allows us to isolate the effects of individual design choices in the case of using DNNs in computational paralinguistics. These experiments allow us to establish increasingly sophisticated guidelines for Seq. Autoencoder and Wav2Vec 2.0 models. We ensured that our recommendations are both theoretically and practically applicable across diverse paralinguistic tasks. The cumulative insights from these two investigations form the basis for our practical guidelines and recommendations for DNN implementations in computational paralinguistics applications.

5.5.1 Audio Preprocessing and Aggregation Methods

Thesis Point III/1. - Using Spectral Sequence-to-Sequence Autoencoders to Assess Mild Cognitive Impairment [46, 90].

In our first research, we investigated how changing the audio preprocessing method in a clinical task affects the final result. We expect specific behaviour because of the fact that DNNs process directly the spectrum of the raw waveform, without relying on hand-crafted features. In our second research, we investigated how the aggregation process affects the final results. As our previous research showed, we expected different behaviours by different aggregations. On the other hand, mostly Seq. Autoencoder models are trained on the same corpus that is used during the classification experiments. However, in the medical speech processing area, the amount of data is extremely limited due to the availability of subjects with the given disease, and the fact that trained doctors are required to diagnose patients. To resolve this, we trained our model on a general ASR audio dataset. We expected that we could demonstrate the robustness of the feature extraction technique in a cross-corpus environment.

Database

In our study, we trained and evaluated our classification model on the MCI database discussed in section 2.3.1. That is, besides the 25 MCI and the 25 control subjects (HCI), we utilised the speech recordings of 25 mild Alzheimer's (mAD) patients as

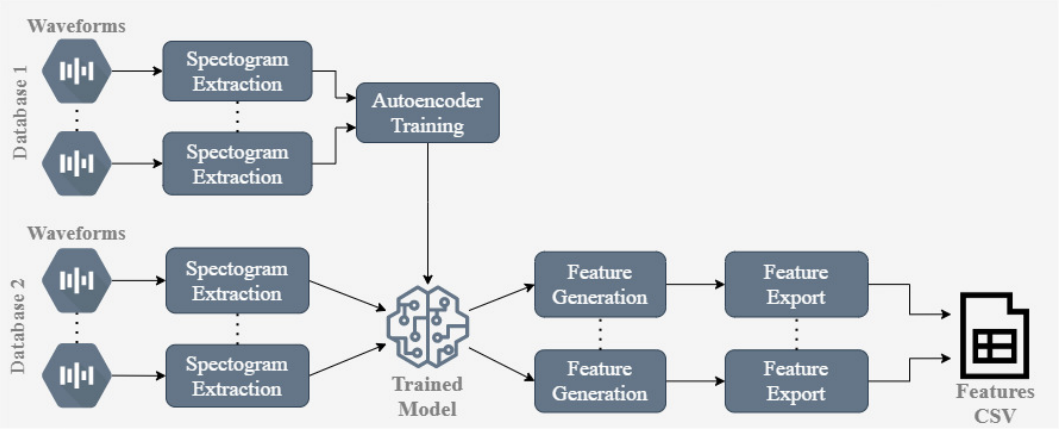

Figure 5.3: *The general workflow of the sequence-to-sequence autoencoder-based feature extraction process that we applied.*

well. They were also matched to the other groups in terms of age, gender and level of education.

Mild Cognitive Impairment (MCI) is a heterogeneous clinical syndrome characterised by the deterioration of memory, language, and problem-solving skills. It is often viewed as the transitional stage between normal ageing and dementia. However, in contrast to those with dementia, the cognitive impairments that occur in MCI are not severe enough to affect the patients' ability to carry out simple everyday activities [1, 67]. Recently, several studies have been published on detecting MCI and other forms of dementia [34, 65]. In this study we apply Seq. Autoencoder for extracting features in order to distinguish the speech of MCI and healthy control subjects.

Frame-Level Features

For the general workflow of our feature extraction approach employed, see Figure 5.3. The first step of the process is the extraction of Mel-scale spectrograms from raw waveforms. We split all recordings into 5-second-long chunks. The Mel-sceptras are normalised into the interval $[-1, 1]$ to match the expected input range for neural networks. We applied 128 log-scale Mel-spectrogram filters with 0.08ms wide windows and a 0.04ms overlap. We used the `AuDeep` toolkit [17], which was written in Python. It normalises all the computed spectrograms to 0 dB. Thresholds were applied after the spectrogram normalisation. We experimented with removing background noise by clipping power levels below a given dB value.

The next step is the model training. To train the Sequence-to-Sequence Autoencoder, we used a subset of the BEA corpus, introduced in section 2.3.1. We employed a small subset, consisting of the speech of 16 subjects with a total duration of 3 hours

and 59 minutes. We used the Adam optimiser with a learning rate of 0.001, and applied dropout with a 0.2 probability. We trained our models with a mini-batch size of 64 for 32 epochs. In our study, each recurrent layer consisted of 128 GRU cells, and the decoder was bidirectional.

Frame-Level Predictions

In this research, we do not create utterance-level features. Instead of it, we performed classification at the level of the 5-second chunks. All features were standardised before utilising them in the classification step. Besides the embeddings extracted from the Seq. Autoencoder, we used one further attribute: the number of chunks associated with the given speaker.

Classification

To aggregate frame-level predictions of a speaker, first, we simply took the (unweighted) arithmetical mean of the posterior scores when we tested the clipping values. In the next experiments, we took a look at the efficiency of other forms of aggregations: median, geometric mean and harmonic mean.

During the classification step, our classifier was a Support Vector Machine, introduced in section 2.3.3. We optimised the complexity parameter using 10 powers between 10^{-5} and 10^{1}. The following powers of 10 were used: -5; -4; -3; -2; -1; 0 and 1. It was implemented with the help of the LIBSVM library [9]. We used 25-fold cross-validation, where each fold consisted only one healthy and one MCI subject. Performance was measured with classification accuracy, equal error rate (EER), and the AUC value (discussed in section 2.3.4).

Results

For reference, we also trained an X-Vector on the same BEA dataset, but on 60 hours and 14 seconds of data with 165 speakers, using F-bank features.

First, we tested the effect of the following noise clipping thresholds: -30 dB, -45 dB, -60 dB, -75 dB. Moreover, we tried concatenating the feature vectors of these four variations (“Merged” approach), and without clipping as well (“Unclipped”). Results are represented in Table 5.1.

We can see that clipping the power levels below a certain dB threshold clearly affects the classification performance. In case of a two-class (MCI and HCI) classification, the largest threshold (-75 dB) led to the best accuracy and AUC scores, although the values corresponding to the -60 dB case were also quite similar. Concatenating the four variations led to a clear fall in the values: although the accuracy

Feature extraction approach		Acc.	AUC
Sequence-to-sequence autoencoders	-30 dB	64%	0.694
	-45 dB	60%	0.706
	-60 dB	68%	0.734
	-75 dB	**72%**	**0.763**
	Merged	68%	0.643
	Unclipped	68%	0.715
x-vectors		60%	0.680

Table 5.1: *The accuracy (Acc.) and AUC scores obtained with the different approaches tested.*

is only slightly lower than the best value, the AUC score is the lowest one for all six cases.

In the next experiment, we investigated how these features could be used to discriminate three speaker categories (MCI, mAD and HCI). We retrained our SVM models on a 3-class task with cross-validation. Table 5.2 shows our results. Now we focus only on the AUC values of the individual speaker categories. In the case of MCI and HC case, the -75 dB threshold again led to the best results. We also observe that the mAD patients could be distinguished from the other speakers with the lowest efficiency. In the -30 dB case, they could not be identified at all. This is surprising as distinguishing healthy controls from the mAD subjects with more prominent symptoms is usually known as an easier task than detecting MCI.

In our last experiments, we investigated the influence of different aggregation techniques. For this, we took a look at the efficiency of the arithmetic mean, median, geometric mean and harmonic mean. Results are shown in Table 5.3.

In the 2-class setup (see the upper half - MCI), we can see that employing the median of the chunk-level posterior estimates was slightly better than using the standard arithmetic mean. Geometric and harmonic means gave almost identical or even

Feature extraction approach		AUC		
		HC	MCI	mAD
Autoencoders	-30 dB	0.706	0.618	0.503
	-45 dB	0.714	0.633	0.569
	-60 dB	0.732	**0.706**	0.606
	-75 dB	**0.771**	**0.710**	0.589
	Merged	0.701	0.622	0.598
	Unclipped	0.682	0.703	**0.629**
x-vectors		0.753	0.546	0.606

Table 5.2: *The AUC scores obtained for the approaches tested in the 3-class case.*

Speaker Category	Aggregation	AUC		
		HC	MCI	mAD
MCI	Arithmetic	0.763	0.763	—
	Median	**0.782**	**0.782**	—
	Geometric	0.760	0.760	—
	Harmonic	0.749	0.749	—
MCI + mAD	Arithmetic	0.771	0.710	0.589
	Median	0.755	**0.712**	0.586
	Geometric	**0.789**	**0.716**	**0.606**
	Harmonic	**0.801**	**0.733**	**0.611**

Table 5.3: *The AUC scores obtained for the different aggregation formulas applied.*

slightly worse values. In the 3-class setup (see the lower half - MCI + mAD), we note the opposite trend. Compared to the arithmetic mean, relying on the median value made the AUC score of the HC speaker category slightly worse (although the AUC values corresponding to the MCI and mAD patients were practically unaltered). Utilising the geometric and the harmonic means improved all three AUC values. These opposing trends, however, seem to indicate the lack of robustness of these aggregation strategies.

Summary of Guidelines

Our study lies in the use of Sequence-to-Sequence Autoencoders to detect mild cognitive impairment and mild Alzheimer's disease. The experiments highlighted important methodological choices that improve performance in paralinguistic classification tasks involving Deep Neural Networks. The following guidelines provide practical recommendations for implementing Seq. Autoencoder feature extraction in computational paralinguistics, especially for clinical applications with limited data availability:

- Audio preprocessing can increase the performance. Removing background noise by clipping power levels below -75 dB enhances classification accuracy.
- Among various power-level clipping strategies, individual threshold conditions outperform combined feature sets.
- Aggregation strategies are crucial, but different methods of aggregating chunk-level predictions (arithmetic mean, median, geometric mean, harmonic mean) showed varying robustness. Median worked better for two-class tasks, while harmonic mean improved three-class distinctions. It is highlighting the need for task-specific aggregation selection.

- A cross-corpus training technique for Sequence-to-Sequence Autoencoders is feasible and beneficial.

According to estimates, the prevalence of MCI ranges from 15% to 20% in individuals of 60 years and older, while the annual progression rate from MCI to dementia is between 8% and 15% [67]. MCI may be present up to 15 years before the clinical manifestation of dementia [48], and this time window offers a chance for early MCI detection, which can provide an opportunity to reduce the rate of cognitive decline [31]. From our results, this approach gives a competitive performance. Taking this into account, automatic speech analysis could prove to be a cheap, easy-to-apply, remote and non-invasive tool for detecting the symptoms of MCI.

5.5.2 Aggregation Strategies

Thesis Point III/2. - Aggregation Strategies of Wav2vec 2.0 Embeddings for Computational Paralinguistic Tasks [95].

In this study, we focus on the utterance-level aggregation step. Although researchers tend to use task-specific aggregations, including only the most popular metrics such as mean and standard deviation, our aim is to show that there are other efficient techniques available too. Some of them can handle different paralinguistic subtopics at the same time. With Wav-to-Vec 2.0 Neural Network embeddings, we investigated 11 aggregation strategies, including both traditional and less frequently employed ones. We conducted experiments on three different databases to find general trends across various paralinguistic subtopics.

In the first phase, we examined how altering a single aggregation method affects the classification performance. We expected not only traditional metrics are give reliable results. In the second phase, we expanded the study to examine how combining multiple aggregation strategies would influence results. Based on our previous findings, we expected that different combinations would lead to diverse outcomes, while it will increase the performance as well.

Database

We performed our experiments on three public paralinguistic corpora, which covered a variety of topics: AIBO, URTIC and iHEARu-EAT. All of them are discussed in more detail in section 2.3.1. All three corpora had native German speakers. This allowed us to justifiably employ the same Wav2Vec 2.0 model for frame-level embeddings extraction, as it was fine-tuned for German speech. In case of training, development and test cuts:

- iHEARu-EAT: The database was divided into a training set (14 speakers), a development set (6 speakers) and a test set (10 speakers) in a speaker-independent manner.
- URTIC: The corpus was divided into three sets (train, dev, test), each containing 210 speakers. The training and development sets contained 37 infected and 173 uninfected participants.
- AIBO: The Ohm subset was divided into a training set (7578 utterances, 20 children) and a development set (2381 utterances, 6 children), while the Mont subset served as the test set (8257 utterances).

Frame-Level Features

To extract frame-level embeddings, we employed a self-supervised and fine-tuned Wav-to-Vec 2.0 Neural Network model. It is introduced in section 5.4. After the training and the fine-tuning, we can use the network as an embedding extractor by freezing the weights and removing the last few layers. We experiment with two setups, where we extract embeddings from:

- the last layer of the CNN block, where the size of the embeddings was 512
- the last layer of the Transformer block, where the size of the embeddings was $1\,024$

When we feed the paralinguistic utterances into the model, the output of the last remaining layer serves as the embeddings.

Utterance-Level Features

During aggregation, we used 11 different statistical methods to convert frame-level embeddings into an utterance-level feature vector. Besides the traditional approaches of *mean*, *median* and *standard deviation*, we experimented with the *skewness*, the *kurtosis*, the*minimum*, the *maximum* and the 1st, 25th, 75th, 99th *percentiles*. All of them are discussed in section 2.3.2 Note that the median is identical to the 50th percentile. The 1st and 99th percentiles are frequently used as alternatives to minimum and maximum, because they are not that sensitive to outliers [61].

Classification

In the end, the classification was performed with an SVM, introduced in section 2.3.3. It was implemented with the help of the LIBSVM library [9]. The C complexity parameter was tested in the range 10^{-5} to 10^{0}. In the optimisation configurations the

following powers of 10 were used: -5; -4; -3; -2; -1 and 0. To avoid peeking and determine the optimal hyperparameter settings, we trained our models on the train set and evaluated them on the development set. In the end, we measured the final performance of the best parameters by training the model on the concatenation of the train and dev sets and evaluating it on the test set. As an evaluation metric, we used the Unweighted Average Recall, introduced in section 2.3.4.

In the case of the AIBO and the URTIC corpora, we always standardised utterance-level features. Due to the unbalanced class distribution and the relatively large size of these corpora, we also employed downsampling on them, as these techniques proved to be beneficial in our previous experiments. In the case of iHEARu-EAT, we performed speaker-wise standardization, where the test set speaker IDs were determined by using the single Gaussian-based bottom-up Hierarchical Agglomerative Clustering algorithm [45, 100].

Results

Based on the best practices from our previous study [96] and to test the influence of different aggregation methodologies with different architectures of SotA solutions, we tested the following setups:

- we compared the performance of the convolutional and the transformer (i.e. hidden) layer embeddings
- we used 11 different statistical methods:
  - mean
  - standard deviation
  - skewness
  - kurtosis
  - minimum (0th percentile)
  - 1st percentile
  - 25th percentile
  - median (50th percentile)
  - 75th percentile
  - 99th percentile
  - maximum (100th percentile)

The best results for the aggregation functions are shown in Figure 5.4.

First, to investigate architectural choices, we can analyse the behaviour of different layers. From our results, convolutional embeddings significantly outperformed

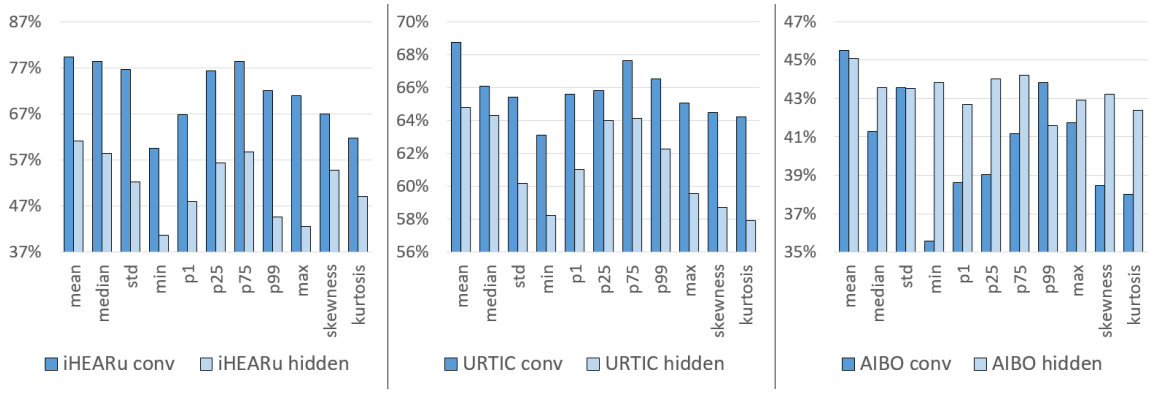

Figure 5.4: *Development results got from convolutional and transformer (i.e. hidden) layer embeddings while using different aggregation techniques. The x axis represents the aggregation method and the y axis represents the UAR value.*

the hidden representations on the iHEARu-EAT corpus. On the other two corpora, it also had a slight advantage against hidden embeddings. Although the hidden layer performed better for percentage aggregations on the AIBO database, it varied greatly, proved unreliable and lost robustness. The other significant difference of the AIBO database compared with the others is that it contains recordings of children's speech. Changes in tones and speech skills can produce slight differences in the analysis results.

Next, if we take a closer look at the aggregations, we can observe general trends. The mean aggregation produced the best results on each database, which, as it is perhaps the most frequently used method, is not that surprising. Standard deviation appears to be a promising alternative for a potential combination. Regarding percentiles, the central ones (i.e. 25%, 75%) have competitive performances, so we should pay more attention to these non-traditional aggregations. We would like to recommend their usage more, especially the 75th percentile. The traditional median metric (which is the same as the 50% percentile) had a varying performance depending on the database, while it follows the curve of the percentiles. This curve shows which part of the ordered sequence is the best descriptor. Last, but not least, for almost all corpora, we obtained very low results with the minimum and maximum aggregations (where the minimum is practically the 0th, while the maximum is the 100th percentile). It tells us that Wav2Vec 2.0 embeddings frequently contain outlier values, which has a significant drawback in classification. Instead of these, the 1st and 99th percentiles are promising alternatives. Although low percentiles may also be minor outliers, the trend clearly shows that their use is more advisable than the minimum and maximum. Lastly, we tested skewness and kurtosis aggregations, but they gave a significantly lower performance overall.

iHEARu-EAT		
Aggregation	**Dev**	**Test**
Mean	79.4%	83.7%
Median	78.4%	82.6%
75th percentile	78.4%	81.2%
mean+std+min+p25	**82.2%**	**85.4%**
All	80.5%	85.0%

Table 5.4: *The best development and test results for different aggregation strategies and their combinations for the iHEARu-EAT paralinguistic corpora.*

Due to the above-mentioned observations, we decided to continue our research with convolutional embeddings, because they behave more robustly and have a global pattern in all three databases. In case of aggregation techniques, we wanted to further improve our solution, so we used Sequential Forward Selection to combine multiple aggregated feature vectors.

With the iHEARu-EAT database Table 5.4 contains an overall statistic. The first three rows show the three best aggregations from the previous experiment of the convolutional layer, which are the mean, the median and the 75th percentile. It serves as a baseline.

The next row shows the best result obtained with the combination approaches. We were able to improve the performance scores up to the 4th iteration. This subset of aggregations determined on the development set contains the mean, standard deviation, minimum and the 25th percentile.

The last row shows the UAR scores we obtained when we combined all of the aggregation methods. It has a score close to the best combination, but we noticed that if we include too much unnecessary information, we can decrease the generalisation ability of our classification model.

With the URTIC database, Table 5.5 contains an overall statistic. The first three rows show the three best aggregations from the previous experiment of the convolutional layer, which are the mean, the 75th percentile and the 99th percentile. It serves as a baseline.

The next row shows the best result obtained with the combination approaches. We found that we can improve the development results up to the 3rd iteration. This subset of aggregations determined on the development set contains the mean, 99th percentile and the maximum.

The last row shows the UAR scores we obtained when we combined all of the aggregation methods. Here we got an increase in the test values. In our opinion, these results indicate that there is a significant difference between the feature distribution of the development and the test sets, because different aggregation types seemed to

be important only in the case of these sets.

With the AIBO database, Table 5.6 contains an overall statistic.

The first three rows show the three best aggregations from the previous experiment of the convolutional layer, which are the mean, the 99th percentile and the standard deviation. It serves as a baseline.

The next row shows the best result obtained with the combination approaches. We found that we could improve development scores up to the 3rd iteration. This subset of aggregations determined on the development set contains the mean, the 99th percentile and the 75th percentile.

The last row shows the UAR scores we obtained when we combined all of the aggregation methods. Here we observed the same behaviour as that for the URTIC database. In our opinion, these results indicate that there is a significant difference between the feature distribution of the development and the test sets, because different aggregation types seemed to be important only in the case of these sets.

In the view of the three databases, we can state that there are database-specific behaviours. Lower than middle percentile values work better for iHEARu-EAT while higher values perform better for URITC and AIBO corpora. On the other hand, there is a global tendency to need 3 or 4 iterations of SFS to improve the efficiency of our model. As we can see, combinations bring improvements to the development and test set as well, which means they increase the generalisation ability of the model. It highlights the importance of diverse aggregations that can be easily calculated alongside traditional metrics. It is always worth containing one or two non-traditional percentile values. These significant improvements were obtained using simple, easy-to-implement and quick-to-calculate aggregation techniques. All metrics can be calculated in parallel from frame-level features. Each new metric introduces as many new features as we originally had. The combination of 3-4 aggregations leads to an utterance-level feature vector of length 1536-2048. This does not drastically increase the dimensionality for a casual set of features extracted from other Deep Neural Networks, which are commonly used in paralinguistics.

URTIC		
Aggregation	**Dev**	**Test**
Mean	68.7%	63.1%
75th percentile	67.6%	66.4%
99th percentile	66.5%	66.2%
mean+p99+max	**69.5%**	64.8%
All	67.3%	**67.1%**

Table 5.5: *The best development and test scores for different aggregation strategies and their combinations for the URTIC paralinguistic corpora.*

AIBO		
Aggregation	**Dev**	**Test**
Mean	45.5%	42.7%
99th percentile	43.8%	42.9%
Standard dev.	43.5%	43.3%
mean+p99+p75	**47.0%**	42.7%
All	44.2%	**44.0%**

Table 5.6: *The best development and test results for different aggregation strategies and combinations for the AIBO paralinguistic corpora.*

Summary of Guidelines

Overall, we found that DNN architectural choices and embedding aggregation choices have a high impact on the classification performance. Certain non-traditional aggregation metrics can be highly effective for almost any paralinguistic subtopic. Traditional metrics vary in performance depending on the dataset. We also performed SFS initialised with the mean to test the combination of different metrics. Our results indicate that the effective summarisation of frame-level embeddings is a nontrivial task, and classification performance can be improved significantly using multiple aggregation functions. In addition, we present a novel rule set for Wav2Vec 2.0 embeddings where we identify general patterns and provide guidelines for selecting appropriate design choices.

- Convolutional embeddings behave more robustly and have a global pattern in different databases
- The mean aggregation is always a good choice, but it is not the only one. Our first results indicate that middle percentile aggregations are competitive techniques.
- The traditional standard deviation and median aggregations are heavily topic dependent.
- Wav2Vec 2.0 embeddings can be expected to contain extreme values, which are not really useful for classification. Owing to this, aggregation methods that are sensitive to outliers might perform less robust. Obvious examples are the minimum and maximum, which were clearly outperformed by the first and 99th percentiles.
- Choosing only one aggregation technique leads to a suboptimal classification performance. The peak of performance fell on the combination of the first 3-4 techniques. The best combinations typically include the mean, a non-traditional

percentile value below and/or above the median. This combination can improve the generalization ability of the model, while keeping the feature space below 2048.

If we are following these rule, the performance of the classification models can have a high improvement, while the computational demand does not increase drastically due to possible parallelisations.

5.6 Concluding Remarks

In this chapter, we conducted an in-depth analysis of Deep Neural Networks as State-of-the-Art feature extraction models in various computational paralinguistics tasks (mild cognitive impairment detection, emotion recognition, cold identification and eating monitoring). Our systematic investigations addressed critical challenges, including audio preprocessing methodologies, aggregation strategy selection, and cross-corpus training approaches.

- The first investigation (Audio Preprocessing and Aggregation Methods in section 5.5.1) systematically explores the impact of data and prediction processing decisions on Sequence-to-Sequence Autoencoder performance. We examined noise reduction strategies (power level clipping thresholds), cross-corpus training approach, and aggregation techniques for chunk-level predictions. This investigation aims to identify optimal preprocessing parameters and establish clear guidelines about these aspects.

  The experiments with different noise reduction thresholds demonstrated the importance of audio preprocessing choices. We identified that removing background noise by clipping power levels below -75 dB consistently enhances classification accuracy across different clinical conditions. Cross-corpus training strategies proved feasible and beneficial, and they allow feature extractors DNNs to be trained on general ASR data. Individual threshold conditions significantly outperformed the combined methodology, suggesting that focused preprocessing approaches are more effective than ensemble-based noise reduction techniques.

- The second investigation (Aggregation Strategies in section 5.5.2) challenges the conventional reliance on only the most common statistical aggregations. It is comprehensively evaluating diverse aggregation methodologies for Wav2Vec 2.0 embeddings. It evaluates 11 different aggregation functions across two model architectural designs and three paralinguistic tasks. It examines both individual and combined configurations to establish optimal aggregation combinations as well.

Our first results demonstrate the influence of architectural design choices of Deep Neural Networks. Convolutional embeddings behaved more robustly than transformer embeddings, showing consistent global patterns across different databases. But Wav2Vec 2.0 embeddings could frequently contain outlier values that negatively impact classification performance.

Our next results demonstrated that percentile-based aggregations significantly outperform traditional mean calculation. Non-traditional metrics, particularly the 75th percentile, proved highly effective for diverse paralinguistic subtopics. Aggregation methods sensitive to extreme values (minimum/maximum) less robust than alternatives like 1st and 99th percentiles.

Our third results demonstrated that a combination of 3-4 aggregation techniques consistently improved performance while maintaining feature dimensionality below 2048, enabling practical implementation without excessive, computational overhead. This trend also showed up across various paralinguistic tasks. These representations demonstrated competitive performance across diverse paralinguistic attributes.

In summary, Deep Neural Networks represent a powerful method in computational paralinguistics, particularly when proper preprocessing and aggregation strategies are employed. The contribution of these researches are about establishing preprocessing guidelines for paralinguistic applications, demonstrating the effectiveness of percentile-based aggregations, and revealing the importance of architectural choices. These results serve as valuable references for both academic researches and practical implementations. However, their effectiveness remains highly dependent on careful optimisation, highlighting the importance of comprehensive methodological tests in the case of deep learning approaches for paralinguistic analysis.

Key Findings of the Thesis - Guidelines

Traditional Method - BoAW - Importance of Parameter Optimisation

- The first focus was on exploring the possibilities within different parameter optimisation strategies and investigating the following parameters: feature transformation (delta features), database scaling (normalisation, standardisation, upsampling), codebook size, clustering and quantisation.
- The second focus was on testing whether corpus-independent processing is achievable with proper parameter optimisation.
- The third focus was on handling stochastic behaviours.

  Revealed Guidelines:

  - Preprocessing is always a good choice. Transform the input dataset to the same scale by normalisation or standardisation.
  - For greater generalisation ability, it is worth including more neighbours in the quantising step, such as 5 or 10.
  - It is worth choosing the size of codebook from a medium-large range (e.g. between 128 and $4\,096$). If possible, try to keep the codebook size low to get a better generalisation.
  - Clustering with the k-means or with the k-means++ algorithms could be equally good.
  - By balancing the frequency of classes seen during learning, we can improve our generalisation ability. Upsampling can help to achieve it.
  - We should calculate and use Δvalues. With them, we can reduce the number of necessary codewords to a moderate size, and the training trends are more consistent than before.
  - Codebooks have practical corpus independence. Each predefined codebook can be successfully used to extract BoAW feature representations of other databases.
  - We can control stochastic behaviour to ensure more reliable processing. Prediction ensembling or feature ensembling is a good choice.

Hybrid Method - Importance of Feature Aggregation

- The focus is on exploring different feature aggregation strategies.

Revealed Guidelines:

- Extracting embeddings from the 4th layer of a HMM/DNN model is always gives the best performance scores.
- Combining three aggregation techniques will always improve the results in any paralinguistic task. But we should carefully select the aggregation techniques used, as the best combination may be task-dependent.
- When choosing the number of aggregations to combine, take into account Occam's razor principle.
- Always consider including the mean, standard deviation and/or zero ratio in the combination.
- The ratio of non-zero activations as an aggregation function proved to be useful in combination. Mean and standard deviation consistently performed best, while non-traditional techniques can enhance their performance.

DNNs - Seq. Autoencoder/Wav2Vec 2.0 - Architecture Dependence

- Seq. Autoencoder - The focus was on audio preprocessing and on different feature aggregation strategies.

  Revealed Guidelines:

  - Audio preprocessing can increase the performance. Removing background noise by clipping power levels below -75 dB enhances classification accuracy.
  - Among various power-level clipping strategies, individual threshold conditions outperform combined feature sets.
  - Aggregation strategies are crucial, but different methods of aggregating chunk-level predictions (arithmetic mean, median, geometric mean, harmonic mean) showed varying robustness. Median worked better for two-class tasks, while harmonic mean improved three-class distinctions. It is highlighting the need for task-specific aggregation selection.
  - A cross-corpus training technique for Sequence-to-Sequence Autoencoders is feasible and beneficial.

- Wav2Vec 2.0 - The focus was on different layers and different feature aggregation strategies.

  Revealed Guidelines:

- Convolutional embeddings behave more robustly and have a global pattern in different databases
- The mean aggregation is always a good choice, but it is not the only one. Our first results indicate that middle percentile aggregations are competitive techniques.
- The traditional standard deviation and median aggregations are heavily topic dependent.
- Wav2Vec 2.0 embeddings can contain outlier values, which are not really useful for classification. As a result, aggregation methods that are sensitive to outliers may perform less robustly. Obvious examples are the minimum and maximum, which were clearly outperformed by the first and 99th percentiles.
- Choosing only one aggregation technique leads to a suboptimal classification performance. The peak of performance fell on the combination of the first 3-4 techniques. The best combinations typically include the mean, a non-traditional percentile value below and/or above the median. This combination can improve the generalisation ability of the model, while keeping the feature space below 2048.

Unified Cross-Thesis Insights

- Aggregation strategies are crucial across all methodologies. More than one strategy always improves, but the actual set of selected strategies can vary for different paralinguistic use cases.
- The combination of different aggregation strategies always improves.
- Parameter optimisation principles scale from traditional to deep learning approaches, but applying normalisation/standardisation and noise reduction is always a good choice.
- Ensemble methods prove a valuable impact in the case of stochastic behaviour.
- In case of Deep Neural Networks, it is important to investigate the output of different layers for embedding extraction.
- Just as neural networks do not always provide the best solution for everything, traditional techniques are not always the most effective. We are highlighting the importance of fusing traditional and modern approaches. Traditional and deep-learning methodologies can boost each other.

Bibliography

[1] Alzheimer's Association. 2020 Alzheimer's disease facts and figures. *Alzheimer's & Dementia*, 16(3):391–460, 2020.

[2] S. Amiriparian, M. Freitag, N. Cummins, and B. Schuller. Sequence-to-sequence autoencoders for unsupervised representation learning from audio. In *Proceedings of DCASE*, pages 17–21, 2017.

[3] Alexei Baevski, Michael Auli, and Alexis Conneau. Wav2vec 2.0: Learning the structure of speech from raw audio, 09 2020.

[4] J. Baker. The dragon system–an overview. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 23(1):24–29, 1975.

[5] Daniel Bone, Matthew P Black, Ming Li, Angeliki Metallinou, Sungbok Lee, and Shrikanth Narayanan. Intoxicated speech detection by fusion of speaker normalized hierarchical features and GMM supervectors. In *Proc. Interspeech 2011*, pages 3217–3220, 09 2011.

[6] Felix Burkhardt, Astrid Paeschke, M. Rolfes, Walter Sendlmeier, and Benjamin Weiss. A database of German emotional speech. In *Proceedings of Interspeech*, pages 1517–1520, 09 2005.

[7] Felix Burkhardt, Markus van Ballegooy, Klaus-Peter Engelbrecht, Tim Polzehl, and Joachim Stegmann. Emotion detection in dialog systems: Applications, strategies and challenges. In *Proceedings of ACII*, pages 985–989, 09 2009.

[8] W.M. Campbell, D.E. Sturim, and D.A. Reynolds. Support vector machines using GMM supervectors for speaker verification. *IEEE Signal Processing Letters*, 13(5):308–311, 2006.

[9] Chih-Chung Chang and Chih-Jen Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:1–27, 3 2011.

[10] Jun Chen, Jieping Ye, Fengyi Tang, and Jiayu Zhou. Automatic detection of alzheimer's disease using spontaneous speech only. In *Proc. Interspeech 2021*, pages 3830–3834, 09 2021.

[11] Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation, 04 2014.

[12] N. Dehak, P. Kenny, R. Dehak, O. Glembek, P. Dumouchel, L. Burget, V. Hubeika, and F. Castaldo. Support Vector Machines and Joint Factor Analysis for speaker verification. In *Proceedings of ICASSP*, pages 4237–4240, 2009.

[13] José Vicente Egas-López and Gábor Gosztolya. Deep Neural Network embeddings for the estimation of the degree of sleepiness. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP*, pages 7288–7292, 06 2021.

[14] José Vicente Egas-López, Mercedes Vetráb, László Tóth, and Gábor Gosztolya. Identifying conflict escalation and primates by using ensemble x-vectors and fisher vector features. In *Proceedings of Interspeech*, pages 476–480, 2021.

[15] José Vicente Egas-López, Gábor Kiss, Dávid Sztahó, and Gábor Gosztolya. Automatic assessment of the degree of clinical depression from speech using x-vectors. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP*, pages 8502–8506, 05 2022.

[16] Florian Eyben, Martin Wöllmer, and Björn Schuller. Opensmile: The Munich versatile and fast open-source audio feature extractor. In *Proceedings of ACM Multimedia*, pages 1459–1462, 10 2010.

[17] M. Freitag, S. Amiriparian, S. Pugachevskiy, N. Cummins, and Björn Schuller. auDeep: Unsupervised learning of representations from audio with Deep Recurrent Neural Networks. *Journal of Machine Learning Research*, 18(173):1–5, 2018.

[18] Yuan Gao, Chenhui Chu, and Tatsuya Kawahara. Two-stage finetuning of wav2vec 2.0 for speech emotion recognition with asr and gender pretraining. In *Interspeech 2023*, pages 3637–3641, 2023.

[19] J.-L. Gauvain and Chin-Hui Lee. Maximum a posteriori estimation for multivariate gaussian mixture observations of markov chains. *IEEE Transactions on Speech and Audio Processing*, 2:291–298, 04 1994.

[20] Diego Giuliani and Bagher BabaAli. Large vocabulary children's speech recognition with DNN-HMM and SGMM acoustic modeling. In *Interspeech 2015*, pages 1635–1639, 2015.

[21] Gábor Gosztolya. Using the Fisher vector representation for audio-based emotion recognition. *Acta Polytechnica Hungarica*, 17:7–23, 01 2020.

[22] Gábor Gosztolya, András Beke, and Tilda Neuberger. Differentiating laughter types via HMM/DNN and probabilistic sampling. In *Speech and Computer, SPECOM 2019*, volume 11658, pages 122–132, 07 2019.

[23] Gábor Gosztolya, Tamás Grósz, Róbert Busa-Fekete, and László Tóth. Detecting the intensity of cognitive and physical load using adaboost and deep rectifier neural networks. In *Proc. Interspeech 2014*, pages 452–456, 09 2014.

[24] Gábor Gosztolya, László Tóth, Veronika Svindt, Judit Bóna, and Ildikó Hoffmann. Using Acoustic Deep Neural Network Embeddings to Detect Multiple Sclerosis from Speech. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP*, pages 6927–6931, 05 2022.

[25] Gábor Gosztolya. Using fisher vector and Bag-of-Audio-Words representations to identify styrian dialects, sleepiness, baby & orca sounds. In *Proc. Interspeech 2019*, pages 2413–2417, 2019.

[26] Gábor Gosztolya. Optimizing class priors to improve the detection of social signals in audio data. *Engineering Applications of Artificial Intelligence*, 107:104541, 2022.

[27] Gábor Gosztolya, Mercedes Vetráb, Veronika Svindt, Judit Bóna, and Ildikó Hoffmann. Wav2vec 2.0 embeddings are no swiss army knife – a case study for multiple sclerosis. In *Interspeech 2024*, pages 2499–2503, 2024.

[28] Félix Grezes, Justin Richards, and Andrew Rosenberg. Let me finish: automatic conflict detection using speaker overlap. In *Proc. Interspeech 2013*, pages 200–204, 09 2013.

[29] J. Grzybowska and S. Kacprzak. Speaker age classification and regression using i-vectors. In *Proceedings of Interspeech*, pages 1402–1406, 2016.

[30] Tamás Grósz, Mittul Singh, Sudarsana Reddy Kadiri, Hemant Kathania, and Mikko Kurimo. Aalto's end-to-end DNN systems for the INTERSPEECH 2020 computational paralinguistics challenge, 2020.

[31] E.A. Hahn and R.Andel. Nonpharmacological therapies for behavioral and cognitive symptoms of mild cognitive impairment. *Journal of Aging and Health*, 23(8):1223–1245, 2011.

[32] Kyu J. Han, Samuel Kim, and Shrikanth S. Narayanan. Strategies to improve the robustness of Agglomerative Hierarchical Clustering under data source variation for speaker diarization. *IEEE Transactions on Audio, Speech and Language Processing*, 16:1590–1601, 09 2008.

[33] Simone Hantke, Felix Weninger, Richard Kurle, Fabien Ringeval, Anton Batliner, Amr Mousa, and Björn Schuller. I hear you eat and speak: Automatic recognition of eating condition and food type, use-cases, and impact on asr performance. *PLOS ONE*, 11:1–24, 05 2016.

[34] R. Haulcy and J. Glass. Classifying alzheimer's disease using audio and text-based representations of speech. *Frontiers in Psychology*, 11, 2020.

[35] Geoffrey Hinton, Li Deng, Dong Yu, George E. Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara N. Sainath, and Brian Kingsbury. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Processing Magazine*, 29:82–97, 10 2012.

[36] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9:1735–1780, 11 1997.

[37] M. Shamim Hossain and Ghulam Muhammad. Cloud-assisted speech and face recognition framework for health monitoring. *Mobile Networks and Applications*, 20:391–399, 06 2015.

[38] Mark Huckvale, András Beke, and Mirei Ikushima. Prediction of sleepiness ratings from voice by man and machine. In *Proceedings of Interspeech*, pages 4571–4575, Shanghai, China, Oct 2020.

[39] Florian Hönig, Anton Batliner, Elmar Nöth, Sebastian Schnieder, and Jarek Krajewski. Acoustic-prosodic characteristics of sleepy speech - between performance and interpretation. In *Speech Prosody 2014*, pages 864–868, 2014.

[40] Georgios Ioannides, Michael Owen, Andrew Fletcher, Viktor Rozgic, and Chao Wang. Towards paralinguistic-only speech representations for end-to-end speech emotion recognition. In *Interspeech 2023*, pages 1853–1857, 2023.

[41] Jesin James, Li Tian, and Catherine Inez Watson. An open source emotional speech corpus for human robot interaction applications. In *Proc. Interspeech 2018*, pages 2768–2772, 09 2018.

[42] Laetitia Jeancolas, Dijana Petrovska-Delacrétaz Graziella Mangone, Badr-Eddine Benkelfat, Jean-Christophe Corvol, Marie Vidailhet, Stéphane Lehéricy, and Habib Benali. X-vectors: New quantitative biomarkers for early Parkinson's Disease detection from speech. *Frontiers in Neuroinformatics*, 15:1–18, 02 2021.

[43] Christian Jones and Jamie Sutherland. *Acoustic Emotion Recognition for Affective Computer Gaming*, volume 4868. Springer-Verlag, 06 2008.

[44] Sudarsana Kadiri, Rashmi Kethireddy, and Paavo Alku. Parkinson's disease detection from speech using single frequency filtering cepstral coefficients. In *Proc. Interspeech 2020*, pages 4971–4975, 09 2020.

[45] Heysem Kaya, Alexey Karpov, and Albert Salah. Fisher Vectors with Cascaded Normalization for Paralinguistic Analysis. In *Proc. Interspeech 2015*, pages 909–913, 09 2015.

[46] Mercedes Kiss-Vetráb, Egas López José Vicente, Réka Balogh, Nóra Imre, Ildikó Hoffmann, László Tóth, Magdolna Pákáski, János Kálmán, and Gábor Gosztolya. Enyhe kognitív zavar automatikus felismerése szekvenciális autoenkóder használatával. *Berend Gábor–Gosztolya Gábor–Vincze Veronika (szerk.): XVIII. Magyar Számítógépes Nyelvészeti Konferencia. Szegedi Tudományegyetem Informatikai Intézet, Szeged*, pages 175–184, 2022.

[47] Jarek Krajewski, Sebastian Schieder, and Anton Batliner. Description of the upper respiratory tract infection corpus (urtic). In *Proc. Interspeech 2017*, 01 2017.

[48] C. Laske, H.R. Sohrabi, S.M. Frost, K. López-de Ipiña, P. Garrard, M. Buscema, J. Dauwels, S.R. Soekadar, S. Mueller, and at. al. Innovative diagnostic tools for early detection of Alzheimer's disease. *Alzheimer's & Dementia*, 11(5):561–578, 2015.

[49] Longfei Li, Yong Zhao, Dongmei Jiang, Yanning Zhang, Fengna Wang, Isabel Gonzalez, Enescu Valentin, and Hichem Sahli. Hybrid deep neural network–hidden markov model (DNN-HMM) based speech emotion recognition. In *2013 Humaine Association Conference on Affective Computing and Intelligent Interaction*, pages 312–317, 2013.

[50] Weiwei Lin and Man-Wai Mak. Wav2spk: A simple DNN architecture for learning speaker embeddings from waveforms. In *Proc. Interspeech 2020*, pages 3211–3215, 09 2020.

[51] Yu-Chen Lin, Yi-Te Hsu, Szu-Wei Fu, Yu Tsao, and Tei-Wei Kuo. Ia-net: Acceleration and compression of speech enhancement using integer-adder deep neural network. In *Proc. Interspeech 2019*, pages 1801–1805, 2019.

[52] M.-T. Luong, Q.V. Le, I. Sutskever, O. Vinyals, and L. Kaiser. Multi-task sequence to sequence learning. In *Proceedings of ICLR*, 2016.

[53] Man-Wai Mak and Wei Rao. Acoustic vector resampling for GMMSVM-based speaker verification. In *Interspeech 2010*, pages 1449–1452, 2010.

[54] MathWorks. What is a support vector machine?, 2025. Accessed: 2025-08-20.

[55] Vetráb Mercedes and Gosztolya Gábor. Érzelmek felismerése magyar nyelvű hangfelvételekből akusztikus szózsák jellemzőreprezentáció alkalmazásával. *Berend Gábor–Gosztolya Gábor–Vincze Veronika (szerk.): XV. Magyar Számítógépes Nyelvészeti Konferencia. Szegedi Tudományegyetem Informatikai Intézet, Szeged*, pages 265–274, 2019.

[56] Florian Metze, Anton Batliner, Florian Eyben, Tim Polzehl, Björn Schuller, and Stefan Steidl. Emotion recognition using imperfect speech recognition. In *Proc. Interspeech 2010*, pages 478–481, 09 2010.

[57] A-R. Mohamed, G.E. Dahl, and G. Hinton. Acoustic modeling using Deep Belief Networks. *IEEE Transactions on Audio, Speech and Language Processing*, 20(1):14–22, 2011.

[58] N. Morgan and H. Bourlard. Continuous speech recognition using multilayer perceptrons with hidden markov models. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP*, volume 1, pages 413–416, 04 1990.

[59] Mustaqeem and Soonil Kwon. CLSTM: Deep feature-based speech emotion recognition using the hierarchical ConvLSTM network. *Mathematics*, 8:1–19, 11 2020.

[60] Tilda Neuberger, Dorottya Gyarmathy, Tekla Etelka Gráczi, Viktória Horváth, Mária Gósy, and András" Beke. Development of a large spontaneous speech database of agglutinative Hungarian language. In *Proceedings of Text, Speech and Dialogue*, volume 8655, pages 424–431, 09 2014.

[61] Caglar Oflazoglu and Serdar Yildirim. Recognizing emotion from Turkish speech using acoustic features. *EURASIP Journal on Audio Speech and Music Processing*, 20013, 12 2013.

[62] Stephanie Pancoast and Murat Akbacak. Bag-of-Audio-Words approach for multimedia event classification. In *Interspeech 2012*, pages 2105–2108, 2012.

[63] Maximilian Panzner and Philipp Cimiano. Comparing hidden markov models and long short term memory neural networks for learning action representations. In *Machine Learning, Optimization, and Big Data*, volume 10122, pages 94–105, 12 2016.

[64] Raghavendra Pappagari, Jaejin Cho, Sonal Joshi, Laureano Moro-Velázquez, Piotr Żelasko, Jesús Villalba, and Najim Dehak. Automatic detection and assessment of alzheimer disease using speech and language technologies in low-resource scenarios. In *Proc. Interspeech 2021*, pages 3825–3829, 09 2021.

[65] P.A. Pérez-Toro, S.P. Bayerl, T. Arias-Vergara, J.C. Vásquez-Correa, P. Klumpp, M. Schuster, Elmar Nöth, and at. al. Influence of the interviewer on the automatic assessment of Alzheimer's disease in the context of the ADReSSo challenge. In *Proceedings of Interspeech*, pages 3785–3789, 2021.

[66] P.A. Pérez-Toro, P. Klumpp, A. Hernandez, T. Arias, P. Lillo, A. Slachevsky, A.M. García, M. Schuster, A.K. Maier, E.Nöth, and J.R. Orozco-Arroyave. Alzheimer's detection from English to Spanish using acoustic and linguistic embeddings. In *Proc. Interspeech 2022*, pages 2483–2487, 09 2022.

[67] R.C. Petersen, B. Caracciolo, C. Brayne, S. Gauthier, V. Jelic, and L. Fratiglioni. Mild Cognitive Impairment: A concept in evolution. *Journal of Internal Medicine*, 275(3):214–228, 2014.

[68] Jiří Přibil, Anna Přibilová, and Jindřich Matoušek. GMM-based speaker age and gender classification in czech and slovak. *Journal of Electrical Engineering*, 68:3–12, 03 2017.

[69] Pappagari Raghavendra, Wang Tianzi, Villalba Jesus, Chen Nanxin, and Dehak Najim. X-vectors meet emotions: A study on dependencies between emotion and speaker recognition, 02 2020.

[70] Maximilian Schmitt, Nicholas Cummins, and Björn W. Schuller. Continuous emotion recognition in speech – do we need recurrence? In *Proc. Interspeech 2019*, pages 2808–2812, 09 2019.

[71] Maximilian Schmitt and Björn Schuller. openXBOW - Introducing the Passau open-source crossmodal Bag-of-Words toolkit. *J. Mach. Learn. Res.*, 18:96:1–96:5, 05 2016.

[72] Björn Schuller and Anton Batliner. *Computational paralinguistics: emotion, affect and personality in speech and language processing*. Wiley Publishing, 11 2013.

[73] Björn Schuller, Stefan Steidl, and Anton Batliner. The INTERSPEECH 2009 emotion challenge. In *Proc. Interspeech 2009*, pages 312–315, 01 2009.

[74] Björn Schuller, Stefan Steidl, Anton Batliner, Elika Bergelson, Jarek Krajewski, Christoph Janott, Andrei Amatuni, Marisa Casillas, Amdanda Seidl, Melanie Soderstrom, et al. The INTERSPEECH 2017 Computational Paralinguistics Challenge: Addressee, Cold & Snoring. In *Proc. Interspeech 2017*, pages 3442–3446, 01 2017.

[75] Björn Schuller, Stefan Steidl, Anton Batliner, Simone Hantke, Florian Hönig, Juan Rafael Orozco-Arroyave, Elmar Nöth, Yue Zhang, and Felix Weninger. The INTERSPEECH 2015 computational paralinguistics challenge: Nativeness, Parkinson's & eating condition. In *Proc. Interspeech 2015*, pages 478–482, 01 2015.

[76] Björn Schuller, Stefan Steidl, Anton Batliner, Alessandro Vinciarelli, Klaus Scherer, Fabien Ringeval, Mohamed Chetouani, Felix Weninger, Florian Eyben, Erik Marchi, Marcello Mortillaro, Hugues Salamin, Anna Polychroniou, Fabio Valente, and Samuel Kim. The INTERSPEECH 2013 computational paralinguistics challenge: Social signals, conflict, emotion, autism. In *Proc. Interspeech 2013*, pages 148–152, 08 2013.

[77] Björn W. Schuller and Felix Weninger. Ten recent trends in computational paralinguistics. In *Cognitive Behavioural Systems - COST 2102 International Training School, Dresden, Germany, February 21-26, 2011, Revised Selected Papers*, volume 7403 of *Lecture Notes in Computer Science*, pages 35–49, 2011.

[78] Björn Schuller, Anton Batliner, Stefan Steidl, Florian Schiel, and Jarek Krajewski. The INTERSPEECH 2011 speaker state challenge. In *Proc. Interspeech 2011*, pages 3201–3204, 01 2011.

[79] Björn Schuller, Stefan Steidl, Anton Batliner, Peter B. Marschik, Harald Baumeister, Fengquan Dong, Simone Hantke, Florian B. Pokorny, Eva-Maria Rathner, Katrin D. Bartl-Pokorny, Christa Einspieler, Dajie Zhang, Alice Baird, Shahin Amiriparian, Kun Qian, Zhao Ren, Maximilian Schmitt, Panagiotis Tzirakis, and Stefanos Zafeiriou. The INTERSPEECH 2018 computational paralinguistics challenge: Atypical & self-assessed affect, crying & heart beats. In *Interspeech 2018*, pages 122–126, 2018.

[80] Björn W. Schuller, Anton Batliner, Christian Bergler, Cecilia Mascolo, Jing Han, Iulia Lefter, Heysem Kaya, Shahin Amiriparian, Alice Baird, Lukas Stappen, Sandra Ottl, Maurice Gerczuk, Panagiotis Tzirakis, Chloë Brown, Jagmohan Chauhan, Andreas Grammenos, Apinan Hasthanasombat, Dimitris Spathis, Tong Xia, Pietro Cicuta, Leon J. M. Rothkrantz, Joeri Zwerts, Jelle Treep, and Casper Kaandorp. The INTERSPEECH 2021 Computational Paralinguistics Challenge: COVID-19 Cough, COVID-19 Speech, Escalation & Primates. In *Proc. Interspeech 2021*, pages 431–435, 01 2021.

[81] Björn W. Schuller, Anton Batliner, Christian Bergler, Eva-Maria Messner, Antonia Hamilton, Shahin Amiriparian, Alice Baird, Georgios Rizos, Maximilian Schmitt, Lukas Stappen, Harald Baumeister, Alexis Deighton MacIntyre, and Simone Hantke. The INTERSPEECH 2020 computational paralinguistics challenge: Elderly emotion, breathing & masks. In *Interspeech 2020*, pages 2042–2046, 2020.

[82] Björn W. Schuller, Anton Batliner, Christian Bergler, Florian B. Pokorny, Jarek Krajewski, Margaret Cychosz, Ralf Vollmann, Sonja-Dana Roelen, Sebastian Schnieder, Elika Bergelson, Alejandrina Cristia, Amanda Seidl, Anne S. Warlaumont, Lisa Yankowitz, Elmar Nöth, Shahin Amiriparian, Simone Hantke, and Maximilian Schmitt. The INTERSPEECH 2019 Computational Paralinguistics Challenge: Styrian Dialects, Continuous Sleepiness, Baby Sounds & Orca Activity. In *Proc. Interspeech 2019*, pages 2378–2382, 01 2019.

[83] Shakeel Ahmad Sheikh, Md Sahidullah, Fabrice Hirsch, and Slim Ouni. Introducing ECAPA-TDNN and Wav2Vec2.0 Embeddings to Stuttering Detection, 04 2022.

[84] David Snyder, Daniel Garcia-Romero, Gregory Sell, Daniel Povey, and Sanjeev Khudanpur. X-vectors: Robust DNN embeddings for speaker recognition. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP*, pages 5329–5333, 04 2018.

[85] Ramakrishnan Srinivasan and Ibrahiem Emary. Speech emotion recognition approaches in human computer interaction. *Telecommunication Systems, Springer*, 52:1467–1478, 03 2011.

[86] Stefan Steidl. *Automatic classification of emotion related user states in spontaneous children's speech*. Logos-Verlag Berlin, Germany, 05 2009.

[87] Dávid Sztahó, Viktor Imre, and Klára Vicsi. Automatic classification of emotions in spontaneous speech. In *Analysis of Verbal and Nonverbal Communication and Enactment. The Processing Issues*, volume 6800, pages 229–239, 09 2011.

[88] Panagiotis Tzirakis, Jiehao Zhang, and Bjorn W Schuller. End-to-end speech emotion recognition using deep neural networks. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP*, pages 5089–5093, 09 2018.

[89] Maarten Van Segbroeck, Ruchir Travadi, Colin Vaz, Jangwon Kim, Matthew P Black, Alexandros Potamianos, and Shrikanth S Narayanan. Classification of cognitive load from speech using an i-vector framework. In *Proc. Interspeech 2014*, pages 751–755, 09 2014.

[90] Mercedes Vetráb, José Vicente Egas-López, Réka Balogh, Nóra Imre, Ildikó Hoffmann, László Tóth, Magdolna Pákáski, János Kálmán, and Gábor Gosztolya. Using spectral sequence-to-sequence autoencoders to assess mild cognitive impairment. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6467–6471. IEEE, 2022.

[91] Mercedes Vetráb and Gábor Gosztolya. Az akusztikus szózsák eljárás korpuszfüggetlenségének vizsgálata. *Berend Gábor–Gosztolya Gábor–Vincze Veronika (szerk.): XVI. Magyar Számítógépes Nyelvészeti Konferencia. Szegedi Tudományegyetem Informatikai Intézet, Szeged*, pages 219–231, 2020.

[92] Mercedes Vetráb and Gábor Gosztolya. Investigating the corpus independence of the bag-of-audio-words approach. In *Text, Speech, and Dialogue: 23rd International Conference, TSD 2020, Brno, Czech Republic, September 8–11, 2020, Proceedings 23*, pages 285–293. Springer International Publishing, 2020.

[93] Mercedes Vetráb and Gábor Gosztolya. Handling the stochastic behaviour of the bag-of-audio-words method. In *2022 IEEE 20th Jubilee World Symposium on Applied Machine Intelligence and Informatics (SAMI)*, pages 000021–000026. IEEE, 2022.

[94] Mercedes Vetráb and Gábor Gosztolya. Using the bag-of-audio-words approach for emotion recognition. *ACTA UNIVERSITATIS SAPIENTIAE INFORMATICA*, 14(1):1–21, 2022.

[95] Mercedes Vetráb and Gábor Gosztolya. Aggregation strategies of wav2vec 2.0 embeddings for computational paralinguistic tasks. In *International Conference on Speech and Computer*, pages 79–93. Springer Nature Switzerland Cham, 2023.

[96] Mercedes Vetráb and Gábor Gosztolya. Using hybrid hmm/dnn embedding extractor models in computational paralinguistic tasks. *Sensors*, 23(11):5208, 2023.

[97] Laurence Vidrascu and Laurence Devillers. Detection of real-life emotions in call centers. In *Proc. Interspeech 2005*, pages 1841–1844, 09 2005.

[98] J.C. Vásquez-Correa, Juan Rafael Orozco-Arroyave, and Elmar Nöth. Convolutional neural network to model articulation impairments in patients with parkinson's disease. In *Proc. Interspeech 2017*, pages 314–318, 09 2017.

[99] Johannes Wagner, Dominik Schiller, Andreas Seiderer, and Elisabeth Andre. Deep Learning in Paralinguistic Recognition Tasks: Are Hand-crafted Features Still Relevant? In *Proc. Interspeech 2018*, pages 147–151, 09 2018.

[100] Wei Wang, Ping Lu, and Yonghong Yan. An improved hierarchical speaker clustering. *Acta Acustica*, 33:9–14, 2008.

[101] Bassem Yamout, Nabil Fuleihan, Taghrid Hajj, Abla Sibai, Omar Sabra, Hani Rifai, and Abdul-Latif Hamdan. Vocal symptoms and acoustic changes in relation to the expanded disability status scale, duration and stage of disease in patients with multiple sclerosis. *European Archives of Oto-Rhino-Laryngology*, 266:1759–1765, 07 2009.

[102] Steve Young, Gunnar Evermann, Mark Gales, Thomas Hain, Dan Kershaw, Xunying Liu, Gareth Moore, Julian Odell, Dave Ollason, Dan Povey, et al. The htk book. *Cambridge university engineering department*, 3(175):12, 2002.

[103] Jia Yu, Xiong Xiao, Lei Xie, Eng Siong Chng, and Haizhou Li. A DNN-HMM approach to story segmentation. In *Interspeech 2016*, pages 1527–1531, 2016.

[104] Teng Zhang and Ji Wu. Speech emotion recognition with i-vector feature and RNN model. In *2015 IEEE China Summit and International Conference on Signal and Information Processing (ChinaSIP)*, pages 524–528, 07 2015.

[105] Ziping Zhao, Zhongtian Bao, Zixing Zhang, Nicholas Cummins, Haishuai Wang, and Björn Schuller. Attention-enhanced connectionist temporal classification for discrete speech emotion recognition. *Proc. Interspeech 2019*, pages 206–210, 09 2019.

Summary

This PhD thesis presents a comprehensive research in the field of computational paralinguistics through a systematic investigation of feature extraction methodologies. Despite the growing number of studies in this area, there is still no consensus on a set of architectural design patterns that can be applied universally. For example, there is no consensus on whether specific methods, such as Wav2Vec 2.0 networks, are universally applicable as feature extractors for different paralinguistic tasks. Some approaches may work well for specific datasets, yet fail to generalise across multiple use-cases. This gap in the literature motivated our study. This research aims to establish global guidelines for processing various paralinguistic corpora. Experiments were conducted in tasks such as emotion recognition, cognitive impairment detection, and other speech-based classification and regression tasks. These use cases often suffer from inconsistent performance across datasets and limited consensus in the case of best practices.

Fundamental challenges were encountered while attempting to develop robust extraction strategies applicable across various datasets. First of all, most paralinguistic corpora remain small (less than 100 hours), making it harder to observe and conclude global trends. On the other hand, the extremely low amount of data is limiting the training of Deep Neural Networks. Moreover, cross-cultural generalisation is a huge challenge. For example, models trained on Western speech underperform on tonal languages (e.g., Mandarin). It highlights the necessity of multilingual speech features. Last, but not least, unified evaluation metrics play a crucial role in the realistic and comparable evaluation of research paper results. We have to promote standardised metrics (e.g., Pearson or Spearman correlation for regression) across tasks to enable direct comparisons. Lastly, computational costs play a crucial role in real-life applications. Deep Neural Networks require more GPU resources than traditional methods, making edge or low-resource deployment more difficult. Comprehensive researches in this field are crucial for the everyday development of paralinguistic systems. In this thesis, two main challenges were highlighted: the importance of parameter optimisation (such as hyperparameters and other architectural design choices) and the selection of aggregation strategies. A focus on these two aspects was chosen to enhance the understanding of features within paralinguistic analysis and to identify methods that could overall enhance the effectiveness of computational

	[94]	[55]	[92]	[91]	[93]	[96]	[90]	[46]	[95]
I/1.	•	•							
I/2.			•	•					
I/3.					•				
II/1.						•			
III/1.							•	•	
III/2.									•

Table 5.7: *The relation between the theses and the corresponding publications*

models. The thesis contributes to creating practical guidelines for the three main categories of machine learning approaches (traditional, deep learning-based and hybrid methodologies).

The dissertation consists of five major parts. In Chapter 1, a short introduction of the thesis points and the contribution of the author is provided. In Chapter 2, a brief introduction to the history of computational paralinguistics is provided, along with a description of the main technical challenges in the field (such as varying-length recordings and small corpora). This chapter also gives an overview of commonly used methodologies. In the next three chapters, different machine learning approaches are explored across the three interconnected research streams: traditional methodologies in Chapter 3, hybrid methodologies in Chapter 4, and State-of-the-Art deep learning approaches in Chapter 5. It reflects the three-stage progression of evolutionary development of the field of artificial intelligence. Traditional methods, for example, the Bag-of-Audio-Words (BoAW) technique, serve as our foundational baseline, establishing performance benchmarks while demonstrating the core principles of parameter optimisation, corpus independence, and robust feature extraction under resource constraints. Building upon these insights, hybrid methodologies bridge the gap between traditional approaches and modern deep learning solutions by combining Hidden Markov Models with Deep Neural Networks. It demonstrates how the integration of traditional statistical methods with neural architectures can achieve resource efficiency while maintaining competitive performance. Finally, the SotA DNN approaches, including the Sequence-to-Sequence Autoencoder and the Wav2Vec 2.0 model, reveal the full potential of automatically learned data features. At the same time, it validates and extends the universal principles discovered in these different streamlines. The methodological progression of the thesis points is designed to reveal comprehensive principles for feature extraction that involve specific algorithmic choices. It ultimately contributes to the development of globally applicable guidelines that researchers and developers can use, based on their specific resource constraints, dataset characteristics, and performance requirements.

Thesis Group I.

In the **first thesis group**, the key findings are related to forming general rules for traditional machine learning approaches. Detailed discussions can be found in Chapter 3. The Bag-of-Audio-Words (BoAW) technique was investigated for this purpose. The processes of parameter optimisation, the nature of corpus independence, and stochastic behaviour were explored. Three different databases were used to provide a comprehensive overview: the Hungarian emotion database, EmoDB, and Sleepiness.

Thesis Point I/1.

In Chapter 3, the Bag-of-Audio-Words technique was explored as a feature extraction method for speech emotion recognition. BoAW provides a structured way to address the issue of handling varying-length recordings for classification. It is clustering frame-level features into ”codebooks” and creating histogram representations in fixed-sized feature vectors. The focus was placed on exploring the possibilities within different parameter optimisation strategies and investigating the following parameters: feature transformation (delta features), database scaling (normalisation, upsampling), codebook size, clustering and quantisation. Experiments conducted on a Hungarian emotion database demonstrate that BoAW enhances classification accuracy, though some parameters require further tuning for optimal results. Guidelines such as applying normalisation/standardisation and upsampling, using delta features and codebook sizes (128–4096 clusters) significantly improve our results. In conclusion, proper parameter optimisation has a high impact on the performance of Bag-of-Audio-Words and gold standards were defined to narrow the pool of possible hyperparameters [55, 94].

Contribution in Thesis Point I/1.

The author implemented the Bag-of-Audio-Words feature-extraction pipeline for emotion recognition, including parameter optimisation for preprocessing, codebook generation, quantisation, and feature transformation. She has taken care of the experimental setup, ensuring speaker-independent evaluation and systematic testing of parameter ranges such as codebook size, neighbour count, clustering algorithms, and delta feature computation. The author performed data preparation, feature extraction using openSMILE and openXBOW, and integrated the output into SVM-based classification. She conducted multiple iterations of experiments to analyse the effect of each parameter on classification accuracy, performed statistical comparison of configurations, and identified optimal parameter settings, while also documenting the findings.

Thesis Point I/2.

In Chapter 3, the exploration of the BoAW technique was continued, with a focus on the corpus-independent capabilities. The research investigates whether BoAW feature extraction can be applied across different datasets without requiring dataset-specific parameter optimisation. Emotion recognition was performed on a Hungarian emotion database, with codebooks calculated from different databases: the same Hungarian emotion database, a German emotion database, and a general Hungarian speech database. Results show that classification accuracy remains consistent across different codebooks, suggesting that BoAW is practically corpus-independent. Codebooks trained on unrelated datasets achieved comparable or superior performance to corpus-specific codebooks, enabling cross-dataset generalisation. These findings support that the BoAW technique is a robust feature extraction method that can be applied across multiple datasets without significant performance loss. In conclusion, the traditional approach was challenged, where the initial clustering step was typically corpus-dependent and performed on the training set of each investigated database. This study demonstrates the importance of corpus independence. It proves that the BoAW feature extraction method can be applied across different datasets without requiring dataset-specific parameter optimisation [91, 92].

Contribution in Thesis Point I/2.

The author implemented an experimental framework for analysing corpus independence in Bag-of-Audio-Words feature extraction. She preprocessed the three different databases, constructed cross-corpus codebooks, and ran systematic tests with an emotion recognition task, while documenting all the results and conclusions.

Thesis Point I/3.

In Chapter 3, the traditional BoAW methodology was refined to investigate its stochastic nature and how its variability affects global guidelines. Since it relies on randomised clustering, it can produce different results even when using the same settings, leading to inconsistencies. This research explores the average aggregation technique to ensemble different features from multiple BoAW models. While ensembling enhances robustness, it also increases feature space size, which can negatively impact classification efficiency. To counteract this, Principal Component Analysis dimensionality reduction was applied. It can help maintain accuracy while reducing computational complexity. The findings suggest that BoAW's stochastic behaviour can be controlled, making our guidelines more reliable for paralinguistic tasks. In conclusion, this study highlighted that in addition to establishing general parameter settings and reusable codebooks for emotion recognition, the stochastic behaviour of

the BoAW method should be handled. This research highlights and demonstrates the importance of the ensemble strategy when a machine learning approach has stochastic behaviour [93].

Contribution in Thesis Point I/3.

The author implemented the experimental framework for the stochastic variability of Bag-of-Audio-Words feature extraction, while documenting all of the results. She developed an infrastructure for repeated feature extraction with multiple random seeds. The author implemented various aggregation strategies and addressed the data dimensionality issue using Principal Component Analysis.

Thesis Group II.

In the **second thesis group**, the key findings are related to the aggregation strategy selection. The study in this section explores the nature of five different strategies. Three different databases were used to cover multiple paralinguistic use-cases: AIBO, URTIC, and iHEARu-EAT. Detailed discussion can be found in Chapter 4.

Thesis Point II/1.

In Chapter 4, the hybrid HMM/DNN methodology was refined to demonstrate best practices and global guidelines for combining traditional methods with deep neural networks. HMM/DNN is a hybrid approach that combines a Hidden Markov Model with a Deep Neural Network for speech processing tasks. The DNN component excels in feature extraction and non-linear mapping, while the HMM component handles temporal modelling of speech sequences. An Automatic Speech Recognition technique and a paralinguistic feature extraction were combined by training a HMM/DNN hybrid acoustic model. This model was used to generate embeddings, which serve as features for various paralinguistic tasks, including emotion recognition, illness detection, and recognising eating conditions. This study discovers task-dependent guidelines for all these different tasks. On the one hand, the main focus was on exploring different feature aggregation strategies, including the mean, standard deviation, skewness, kurtosis, and the ratio of non-zero activations. Parallel with that, another investigation was conducted about architectural choices in the case of the DNN. The embeddings were extracted from five different layers. Results showed that there are best practices that can improve classification results. Pretraining DNNs on large ASR corpora allowed effective embedding extraction. In addition, the following guidelines were formulated: when extracting embeddings the 4th layer could be an optimal choice; when choosing the number of aggregations to combine,

it is worth taking into account Occam's razor principle; combining three techniques will always improve the results in multiple paralinguistic tasks; mean and standard deviation consistently performed best across different aggregations; the ratio of non-zero activations proved useful in certain cases, particularly in combinations. In conclusion, this research has revealed the difficulty of selecting the most reliable embeddings and the optimal aggregation strategy. Always using only one aggregation is not robust enough. Just as neural networks do not always provide the best solution for everything, traditional techniques are not always the most effective. It is essential to highlight the importance of fusing traditional and modern approaches. Traditional and deep-learning methodologies can boost each other. These policies can be competitive design choices of hybrid systems in computational paralinguistics [96].

Contribution in Thesis Point II/1.

The author implemented an aggregation pipeline, which included various aggregations, feature transformations, and classification. She has taken care of this experimental setup, ensuring systematic testing in different databases. The author performed data preparation and integrated the output into SVM-based classification. She conducted multiple iterations of experiments to analyse the effect of layers and aggregations. She performed the comparison of configurations and identified optimal parameter settings while also documenting the findings.

Thesis Group III.

In the **third thesis group**, the key findings are related to the investigation booth parameter optimisation and aggregation strategy selection in the case of Deep Neural Network (DNN) methodologies. These studies are exploring the effectiveness of noise reduction and the nature of 10 aggregation strategies. Four different databases were used in this section: a Hungarian Mild Cognitive Impairment (MCI) database, AIBO, URTIC, iHEARu-EAT. Detailed discussion can be found in Chapter 5.

Thesis Point III/1.

In Chapter 5, a Deep Neural Network methodology was used to demonstrate global guidelines. The use of Sequence-to-Sequence Autoencoders was investigated for assessing Mild Cognitive Impairment. It is a deep learning-based feature extraction method, trained to reproduce its input, forcing the data through a lower-dimensional "bottleneck" layer. Its encoder can map each frame-level raw audio feature to a fixed-size embedding. Two aspects were explored. The first experiment focused on the

influence of the audio preprocessing, especially the minimum power level of recordings. The experiments involved removing background noise by clipping power levels to a given dB value (-30, -45, -60, -75). The best performance was achieved with a -75 dB threshold, suggesting the first guideline. The second experiment focused on different feature aggregation strategies. Four aggregation methods were evaluated on chunk-level posteriors to obtain speaker-level scores. The aggregations were: arithmetic mean, median, geometric mean, and harmonic mean. In conclusion, this study highlights the importance of noise reduction and demonstrates the effect of different aggregation strategies. It established a noise-related guideline, connected to the best performance that was achieved with a -75 dB threshold. It also established a few aggregation guidelines: in case of 2-class tasks, the median marginally outperformed the arithmetic mean, while for 3-class tasks, the harmonic mean proved to be the most effective [46, 90].

Contribution in Thesis Point III/1.

The author implemented a classification pipeline that included various prediction-level aggregations and feature transformations. Based on already calculated posterior values, she has taken care of this experimental setup, ensuring systematic testing. The author performed data preparation and evaluated the posteriors. She conducted multiple iterations of experiments to analyse the effect of noise reduction and aggregations. She performed a comparison of configurations, identifying optimal noise reduction settings and aggregation techniques, while also documenting her findings.

Thesis Point III/2.

In Chapter 5, the State-of-the-Art Wav-to-Vec 2.0 Neural Network was investigated. It is a self-supervised deep learning model that learns speech representations directly from raw audio waveforms. It first uses a CNN with dilated convolutions to encode the waveform into latent feature vectors, then applies a transformer network to generate contextualised embeddings. To investigate architectural design options, frame-level embeddings were extracted from both the final convolutional layer and the last transformer layer. This study has two main focuses: architectural choices and aggregation techniques. First, the given results showed that convolutional-layer embeddings consistently outperformed transformer-layer embeddings. Secondly, aggregation techniques for Wav2Vec 2.0 embeddings were explored, ensuring robust feature extraction across different paralinguistic tasks. IT evaluates 11 different aggregation functions: mean, standard deviation, skewness, kurtosis, percentiles (1st, 25th, 75th, 99th), minimum and maximum. These methods were tested individually and in combined configurations. The results highlight multiple guidelines. Non-traditional metrics, especially percentiles, provide notable performance improve-

ments over standard mean-based aggregation. Lower percentile values worked better for food classification tasks, while higher percentiles proved more effective for emotion recognition and speech distorting infection assessments. Using three to four aggregation techniques together resulted in increased performance, highlighting the value of combining statistical measures rather than relying solely on traditional aggregation methods. It highlights the importance of non-traditional metrics, especially percentiles, that provide notable performance improvements over standard mean-based aggregation. These experiments have established multiple global guidelines; however, different paralinguistic tasks can differ in what constitutes the optimal set of strategies [95].

Contribution in Thesis Point III/2.

The author implemented the classification pipeline, which included various aggregations and feature transformations. Based on already calculated frame-level feature vectors, she has taken care of the experimental setup, ensuring systematic testing in different databases. The author performed data preparation and integrated the output into SVM-based classification. She conducted multiple iterations of experiments to analyse the effect of aggregation methodologies and the features given by different layers. She performed a comparison of configurations, identifying optimal layer settings and aggregation techniques, while also documenting her findings.

Összefoglalás

Ebben a PhD értekezésben átfogó kutatásokat mutatok be a számítógépes paralingvisztika területén, a jellemző-kinyerési módszerek szisztematikus vizsgálatán keresztül. A terület növekvő számú tanulmánya ellenére még mindig nincs konszenzus olyan metodikai és tervezési mintákról, amelyek univerzálisan alkalmazhatók lennének. Például nincs konszenzus arról, hogy olyan specifikus módszerek, mint a Wav2Vec 2.0 hálózatok, univerzálisan alkalmazhatók-e jellemző-kinyerőként különböző paralinguisztikai feladatokhoz. Egyes megközelítések jól működhetnek specifikus adathalmazokon, mégis kudarcot vallanak több használati eset általánosításában. Ez az irodalmi űr motiválta tanulmányunkat. Ez a kutatás globális irányelvek megállapítását célozza különböző paralinguisztikai korpuszok feldolgozásához. Kísérleteket végeztünk olyan feladatokban, mint az érzelemfelismerés, kognitív károsodás észlelése és egyéb beszéd-alapú osztályozási és regreziós feladatok. Ezeknél a felhasználási eseteknél gyakran inkonzisztens a teljesítmény különböző adathalmazok között és korlátozott konszenzus van a bevált gyakorlatok esetében.

Különböző kihívásokkal találkoztunk, miközben robusztus kinyerési stratégiák kifejlesztésére törekedtünk, amelyek különböző adathalmazokra alkalmazhatók. Mindenekelőtt a legtöbb paralinguisztikai korpusz kicsi (kevesebb mint 100 óra), ami megnehezíti a globális trendek megfigyelését és következtetését. Másrészt a rendkívül alacsony adatmennyiség korlátozza a mélytanuló hálózatok tanítását. Ezen túlmenően a kultúrákon átívelő általánosítás hatalmas kihívás. Például a nyugati beszéden képzett modellek alulteljesítenek tonális nyelveken (pl. mandarin). Ez kiemeli a többnyelvű beszédjellemzők szükségességét. Végül, de nem utolsósorban, az egységes értékelési metrikák kulcsszerepet játszanak a kutatási cikk eredmények realisztikus és összehasonlítható értékelésében. Szabványosított metrikákat kell támogatnunk (pl. Pearson vagy Spearman korreláció regresszióhoz) a feladatok között a közvetlen összehasonlítások lehetővé tétele érdekében. Végül a számítási költségek kulcsszerepet játszanak a valós alkalmazhatóságban. A mélytanuló hálózatok több GPU erőforrást igényelnek, mint a tradicionális módszerek, ami megnehezíti az alacsony erőforrású környezetekben való felhasználást. Az átfogó kutatások ezen a területen létfontosságúak a paralinguisztikai rendszerek mindennapi használatához. Ebben az értekezésben két fő kihívást emelünk ki: a paraméter-optimalizálás fontosságát és az aggregációs stratégiák kiválasztását. E két szempont hangsúlyozását

választottuk a paralinguisztikai analízisben található jellemzők megértésének kiemelésére és olyan módszerek azonosítására, amelyek összességében javíthatják a gépi modellek hatékonyságát. Az értekezés hozzájárul a gyakorlati irányelvek kidolgozásához a gépi tanulási megközelítések három fő kategóriájában (hagyományos, mélytanulás alapú és hibrid módszerek).

A disszertáció öt fő részből áll. Az 1. fejezetben rövid bevezetést adunk a tézispontokról és a szerző közreműködéséről. A 2. fejezetben rövid bevezetést nyújtunk a számítógépes paralinguisztika történetéről, valamint a terület fő technikai kihívásainak leírásáról (mint például a változó hosszúságú felvételek és kis méretű korpuszok). Ez a fejezet áttekintést ad a gyakran használt módszerekről is. A következő három fejezetben különböző gépi tanulási megközelítéseket vizsgálunk, három, összekapcsolt kutatási ágon keresztül: hagyományos módszerek a 3. fejezetben, hibrid módszerek a 4. fejezetben, és a legmodernebb mélytanulási megközelítések az 5. fejezetben. Ez tükrözi a mesterséges intelligencia területének evolúciós fejlődésének háromfázisú progresszióját. A hagyományos módszerek, például a Bag-of-Audio-Words (BoAW) technika, alapvető referenciapontokként szolgálnak, teljesítménymércéket állítva, miközben bemutatják a paraméter-optimalizálás, a korpusz-függetlenség és a robusztus jellemző-kinyerés alapelveit. Ezekre az ismeretekre építve a hibrid módszerek áthidalják a szakadékot a hagyományos megközelítések és a modern mélytanulási megoldások között, például Hidden Markov Modelleket kombinálva mélytanuló hálózatokkal. Ez bemutatja, hogyan érhető el erőforrás-hatékonyság a hagyományos statisztikai módszerek és neurális architektúrák integrációjával, miközben versenyképes teljesítményt tartunk fenn. Végül a legmodernebb DNN megközelítések, beleértve a Sequence-to-Sequence Autoencodert és a Wav2Vec 2.0 modellt, feltárják az automatikusan tanult adatjellemzők teljes potenciálját. A tézismunka egyidejűleg validálja és kiterjeszti az ezekben a különböző vonulatokban felfedezett univerzális elveket. Az értekezés pontok módszertani progressziója úgy van kialakítva, hogy átfogó elveket tárjon fel a jellemző-kinyeréshez, amelyek specifikus algoritmikus választásokat foglalnak magukban. Végső soron hozzájárul a globálisan alkalmazható irányelvek kifejlesztéséhez, amelyeket kutatók és fejlesztők használhatnak specifikus erőforrású, adathalmazú és teljesítménykövetelményű környezetekben.

A disszertáció 1. tézise

Az első téziscsoportban a kulcsfontosságú megállapítások a hagyományos gépi tanulási megközelítések általános szabályainak kialakításához kapcsolódnak. Részletes leírások találhatók a 3. fejezetben. A Bag-of-Audio-Words (BoAW) technikát vizsgáltuk erre a célra. A paraméter-optimalizálás folyamatait, a korpusz-függetlenség természetét és a sztochasztikus viselkedést kutattuk. Három különböző adatbázist hasz-

náltunk átfogó áttekintés biztosítására: a magyar érzelemadatbázist, az EmoDB-t és a Sleepiness adatbázist.

I/1. Tézispont

A 3. fejezetben a Bag-of-Audio-Words technikát vizsgáltuk beszéd érzelemfelismerési jellemző-kinyerési módszerként. A BoAW strukturált módot biztosít a változó hosszúságú felvételek kezelésének problémájára osztályozáshoz. Keret-szintű jellemzőket klaszterez "kódkönyvekbe" és hisztogram reprezentációkat hoz létre rögzített méretű jellemző-vektorokban. A hangsúlyt különböző paraméter-optimalizálási stratégiák lehetőségeinek feltárására helyeztük, a következő paraméterek vizsgálatával: jellemző-transzformáció (delta jellemzők), adatbázis-skálázás (normalizálás, felülmintavételezés), kódkönyv mérete, klaszterezés és kvantálás. A magyar érzelemadatbázison végzett kísérletek azt mutatják, hogy a BoAW javítja az osztályozási pontosságot, bár egyes paraméterek további hangolást igényelnek az optimális eredményekhez. Az olyan irányelvek, mint a normalizálás/standardizálás és felülmintavételezés alkalmazása, delta jellemzők használata és kódkönyv méretek (128–4096 klaszter) jelentősen javítják eredményeinket. Következtetésként a megfelelő paraméter-optimalizálás nagy hatást gyakorol a Bag-of-Audio-Words teljesítményére, és egységes standardokat határoztunk meg a lehetséges hiperparaméterek körének szűkítésére.

I/2. Tézispont

A 3. fejezetben a BoAW technika vizsgálata folytatódott, a korpusz-független képességekre fókuszálva. A kutatás azt vizsgálta, hogy a BoAW jellemző-kinyerés alkalmazható-e különböző adathalmazokon anélkül, hogy adathalmaz-specifikus paraméter-optimalizálást igényelne. Érzelemfelismerést végeztünk egy magyar érzelemadatbázison, különböző adatbázisokból számított kódkönyvekkel: ugyanaz a magyar érzelemadatbázis, egy német érzelemadatbázis és egy általános magyar beszédadatbázis. Az eredmények azt mutatják, hogy az osztályozási pontosság konzisztens marad a különböző kódkönyvek között, ami arra utal, hogy a BoAW gyakorlatilag korpusz-független. A nem kapcsolódó adathalmazokon képzett kódkönyvek hasonló vagy jobb teljesítményt értek el, mint a korpusz-specifikus kódkönyvek, lehetővé téve a kereszt-adathalmazos általánosítást. Ezek a megállapítások alátámasztják, hogy a BoAW technika robusztus jellemző-kinyerési módszer, amely több adathalmazon alkalmazható jelentős teljesítményveszteség nélkül. Következtetésként megkérdőjeleztük a hagyományos megközelítést, ahol a kezdeti klaszterezési lépés tipikusan korpusz-függő volt és minden vizsgált adatbázis, tanító halmazán végezték. Ez a tanulmány bemutatja a korpusz-függetlenség fontosságát. Bizonyítja, hogy a BoAW jellemző-kinyerési módszer alkalmazható különböző adathalmazokon anélkül, hogy

adathalmaz-specifikus paraméter-optimalizálást igényelne.

I/3. Tézispont

A 3. fejezetben a hagyományos BoAW módszertan finomítására került sor: a sztochasztikus természetének és változékonyságának globális irányelvekre gyakorolt hatásának vizsgálatára. Mivel randomizált klaszterezésre támaszkodik, különböző eredményeket produkálhat még akkor is, ha ugyanazokat a beállításokat használjuk, inkonzisztenciákhoz vezetve. Ez a kutatás feltárja az átlagos aggregációs technikát különböző jellemzők együttesére több BoAW modellből. Bár ez növeli a robusztusságot, növeli a jellemzőtér méretét is, ami negatívan befolyásolhatja az osztályozási hatékonyságot. Ennek ellensúlyozására Principal Component Analysis dimenziócsökkentést alkalmaztunk. Ez segíthet a pontosság fenntartásában a számítási komplexitás csökkentése mellett. A megállapítások azt sugallják, hogy a BoAW sztochasztikus viselkedése kontrollálható, megbízhatóbbá téve irányelveinket paralinguisztikai feladatokhoz. Következtetésként ez a tanulmány rávilágított arra, hogy az általános paraméter-beállítások és újrafelhasználható kódkönyvek megállapítása mellett az érzelemfelismeréshez a BoAW módszer sztochasztikus viselkedését is kezelni kell. Ez a kutatás kiemeli és bemutatja az együttes stratégia fontosságát, amikor egy gépi tanulási megközelítés sztochasztikus viselkedéssel rendelkezik.

A disszertáció 2. tézise

A második téziscsoportban a kulcsfontosságú megállapítások az aggregációs stratégia kiválasztásához kapcsolódnak. Az ebben a részben szereplő tanulmány öt különböző stratégia természetét vizsgálja. Három különböző adatbázist használtunk több paralinguisztikai használati eset lefedésére: AIBO, URTIC és iHEARu-EAT. Részletes leírás található a 4. fejezetben.

II/1. Tézispont

A 4. fejezetben a hibrid HMM/DNN módszertan finomítására került sor, a bevált gyakorlatok és globális irányelvek bemutatására, a hagyományos módszerek mélytanuló hálózatokkal való kombinálásához. A HMM/DNN egy hibrid megközelítés, amely a Hidden Markov Model és a Mély Neurális Háló kombinációját alkalmazza beszédfeldolgozási feladatokhoz. A DNN komponens kiválóan teljesít a jellemzők kivonásában és a nemlineáris leképezésben, míg a HMM komponens a beszédszekvenciák időbeli modellezését végzi. Automatikus beszédfelismerési technikát és paralinguisztikai jellemző-kinyerést kombináltunk egy HMM/DNN hibrid akusztikus modell képzésével.

Ezt a modellt beágyazások generálására használják, amelyek különböző paralinguisztikai feladatok jellemzőiként szolgálnak, beleértve az érzelemfelismerést, betegségészlelést és étkezési körülmények felismerését. Ez a tanulmány feladat-függő irányelveket tár fel, a különböző feladatokhoz. Egyrészt a fő hangsúly különböző jellemző-aggregációs stratégiák feltárásán volt, beleértve az átlagot, szórást, skewnesst, kurtózist és a nem-nulla aktivációk arányát. Ezzel párhuzamosan egy másik vizsgálatot végeztünk a DNN esetében a strukturális választásokról. A beágyazásokat öt különböző rétegből nyertük ki. Az eredmények azt mutatták, hogy vannak bevált gyakorlatok, amelyek javíthatják az osztályozási eredményeket. A DNN-ek nagy ASR korpuszokon való előtanítása lehetővé tette a hatékony beágyazás-kinyerést. Emellett a következő irányelveket fogalmaztuk meg: beágyazások kinyerésekor a 4. réteg lehet optimális választás; az aggregációk számának kiválasztásakor érdemes figyelembe venni Occam borotvájának elvét; három technika kombinálása mindig javítja az eredményeket több paralinguisztikai feladatban; az átlag és szórás konzisztensen a legjobban teljesített a különböző aggregációk között; a nem-nulla aktivációk aránya bizonyos esetekben hasznosnak bizonyult, különösen a kombinációkban. Következtetésként ez a kutatás feltárta a legmegbízhatóbb beágyazások és az optimális aggregációs stratégia kiválasztásának nehézségét. Mindig csak egy aggregáció használata nem elég robusztus. Ahogy a neurális hálózatok nem mindig nyújtják a legjobb megoldást mindenre, a hagyományos technikák sem mindig a leghatékonyabbak. Kiemelhetjük a hagyományos és modern megközelítések ötvözésének fontosságát. A hagyományos és mélytanulási módszerek erősíthetik egymást. Ezek az irányelvek versenyképes tervezési választások lehetnek hibrid rendszerekben a számítógépes paralinguisztikában.

A disszertáció 3. tézise

A harmadik téziscsoportban a kulcsfontosságú megállapítások mind a paraméteroptimalizálás, mind az aggregációs stratégia kiválasztásának vizsgálatához kapcsolódnak mélytanuló hálózatok esetében. Ezek a tanulmányok a zajcsökkentés hatékonyságát és 10 aggregációs stratégia természetét vizsgálják. Négy különböző adatbázist használtunk ebben a részben: egy magyar enyhe kognitív károsodás (MCI) adatbázist, AIBO-t, URTIC-et, iHEARu-EAT-et. Részletes leírás található az 5. fejezetben.

III/1. Tézispont

Az 5. fejezetben egy mélytanuló módszertant használtunk a globális irányelvek bemutatására. A Sequence-to-Sequence Autoencoderek használatát vizsgáltuk az enyhe kognitív károsodás értékeléséhez. Ez egy mélytanulás-alapú jellemző-kinyerési

módszer, amelyet úgy tanítanak, hogy reprodukálja a bemenetét, az adatokat egy alacsonyabb dimenziós ”szűk keresztmetszet” rétegen átvíve. Az encoder minden keret-szintű audio jellemzőt, rögzített méretű beágyazásra képez. Két szempontot vizsgáltunk. Az első kísérlet az audio előfeldolgozás hatására összpontosított, különösen a felvételek minimum hangszintjére. A kísérletek a háttérzaj eltávolítását foglalták magukban, a hangszintek adott dB értékre való vágásával (-30, -45, -60, -75). A legjobb teljesítményt -75 dB küszöbértékkel értük el, ami az első irányelvet adja. A második kísérlet különböző jellemző-aggregációs stratégiákra összpontosított. Négy aggregációs módszert értékeltünk, chunk-szintű posteriori becsléseken beszélőszintű eredmények megszerzéséhez. Az aggregációk a következők voltak: aritmetikai átlag, medián, geometriai átlag és harmonikus átlag. Következtetésként ez a tanulmány kiemeli a zajcsökkentés fontosságát és bemutatja a különböző aggregációs stratégiák hatását. Egy, zajjal kapcsolatos irányelvet állapítottunk meg, amely a -75 dB küszöbértékkel elért legjobb teljesítményhez kapcsolódik. Néhány aggregációs irányelvet is megállapítottunk: 2-osztályos feladatok esetében a medián marginálisan felülmúlta az aritmetikai átlagot, míg 3-osztályos feladatok esetében a harmonikus átlag bizonyult a leghatékonyabbnak.

III/2. Tézispont

Az 5. fejezetben a State-of-The-Art Wav-to-Vec 2.0 neurális hálózat vizsgálata olvasható. Ez egy önfelügyelt mélytanulási modell, amely beszéd reprezentációkat tanul közvetlenül nyers audio felvételekből. Először CNN-t használ dilatált konvolúciókkal a látens jellemző-vektorokká kódoláshoz, majd transzformer hálózatot alkalmaz kontextualizált beágyazások generálására. Az strukturális tervezési opciók vizsgálatához keret-szintű beágyazásokat nyertünk ki, mind a végső konvolúciós rétegből, mind az utolsó transzformer rétegből. Ennek a tanulmánynak két fő fókusza van: strukturális választások és aggregációs technikák. Először az adott eredmények azt mutatták, hogy a konvolúciós réteg beágyazásai konzisztensen felülmúlták a transzformer réteg beágyazásait. Másodszor, a Wav2Vec 2.0 beágyazások aggregációs technikáit vizsgálták, biztosítva a robusztus jellemző-kinyerést különböző paralinguisztikai feladatok között. 11 különböző aggregációs függvényt értékeltünk: átlag, szórás, skwness, kurtózis, percentilisek (1., 25., 75., 99.), minimum és maximum. Ezeket a módszereket egyenként és kombinált konfigurációkban teszteltük. Az eredmények több irányelvet emelnek ki. A nem hagyományos metrikák, különösen a percentilisek, jelentős teljesítményjavulást nyújtanak a standard átlag-alapú aggregációhoz képest. Az alacsonyabb percentilis értékek jobban működtek étel-osztályozási feladatoknál, míg a magasabb percentilisek hatékonyabbnak bizonyultak érzelemfelismerés és beszédet torzító betegségek értékelésénél. Három-négy aggregációs technika együttes használata növekedett teljesítményt eredményezett, kiemelve a statisztikai

számítások kombinálásának értékét, ahelyett, hogy kizárólag hagyományos aggregációs módszerekre támaszkodnánk. Kiemeli a nem hagyományos metrikák, különösen a percentilisek fontosságát, amelyek jelentős teljesítményjavulást nyújtanak a standard átlag-alapú aggregációhoz képest. Ezek a kísérletek több globális irányelvet állapítottak meg; azonban a különböző paralinguisztikai feladatok eltérhetnek abban, hogy mi alkotja a stratégiák optimális halmazát .

Első kontribúció

Az **első téziscsoportban** a hozzájárulásaim a hagyományos gépi tanulási módszerek általános szabályainak kialakításához kapcsolódnak. A részletes bemutatás a 3. fejezetben található.

I/1. A szerző implementálta a Bag-of-Audio-Words jellemzőkinyerési folyamatot érzelemfelismeréshez, beleértve a paraméter-optimalizálást, az előfeldolgozást, kódkönyv generálást, kvantálást és jellemző-transzformációt. Gondoskodott a kísérleti beállításokról, biztosítva a beszélőfüggetlen értékelést és a paramétertartományok szisztematikus tesztelését, mint például a kódkönyv mérete, szomszédszám, klaszterező algoritmusok és delta jellemzők számítása. A szerző elvégezte az adatelőkészítést, jellemzőkinyerést az openSMILE és openXBOW programok használatával, és integrálta a kimenetet SVM-alapú osztályozásba. Többszörös iterációs kísérleteket végzett az egyes paraméterek osztályozási pontosságra gyakorolt hatásának elemzésére, statisztikai összehasonlítást végzett a konfigurációk között, és azonosította az optimális paraméterbeállításokat, miközben dokumentálta a megállapításokat.

I/2. A szerző implementált egy kísérleti keretrendszert a Bag-of-Audio-Words jellemzőkinyerés korpuszfüggetlenségének elemzésére. Előfeldolgozta a három különböző adatbázist, kereszt-korpusz kódkönyveket épített fel, és szisztematikus teszteket futtatott érzelemfelismerési feladattal, miközben dokumentálta az összes eredményt és következtetést.

I/3. A szerző implementálta a BoAW jellemzőkinyerés sztochasztikus változékonyságának kísérleti keretrendszerét, miközben dokumentálta az összes eredményt. Kifejlesztett egy infrastruktúrát a többszörös véletlenszerű seedekkel történő ismételt jellemzőkinyeréshez. A szerző implementálta a különböző aggregációs stratégiákat és kezelte az adatdimenzionalitás problémáját Principal Component Analysis segítségével.

Második kontribúció

Az **második téziscsoportban** a hozzájárulásaim a hibrid gépi tanulási módszerek általános szabályainak kialakításához kapcsolódnak. A részletes bemutatás a 4. feje-

zetben található.

II/1. A szerző implementálta az aggregációs folyamatokat, beleértve a különböző aggregációkat, jellemzőtranszformációt és osztályozást. Gondoskodott a kísérleti beállításokról, biztosítva a szisztematikus tesztelést különböző adatbázisokon. A szerző elvégezte az adatelőkészítést és integrálta a kimenetet SVM-alapú osztályozásba. Többszörös iterációs kísérleteket végzett a rétegek és aggregációk hatásának elemzésére. Elvégezte a konfigurációk összehasonlítását és azonosította az optimális paraméterbeállításokat, miközben dokumentálta a megállapításokat.

Harmadik kontribúció

Az **harmadik téziscsoportban** a hozzájárulásaim a mély neurális gépi tanulási módszerek általános szabályainak kialakításához kapcsolódnak. A részletes bemutatás a 5. fejezetben található.

III/1. A szerző implementálta az osztályozási folyamatot, beleértve a különböző predikciószintű aggregációkat és jellemzőtranszformációkat. A már kiszámított poszterior értékek alapján gondoskodott a kísérleti beállításról, biztosítva a szisztematikus tesztelést. A szerző elvégezte az adatelőkészítést és implementálta a poszteriorok értékelését. Többszörös iterációs kísérleteket végzett a zajcsökkentés és aggregációk hatásának elemzésére. Elvégezte a konfigurációk összehasonlítását és azonosította az optimális zajcsökkentési beállításokat és aggregációs technikákat, miközben dokumentálta a megállapításokat.

III/2. A szerző implementálta az osztályozási folyamatot, beleértve a különböző aggregációkat és jellemzőtranszformációkat. A már kiszámított frame-szintű jellemzővektorok alapján gondoskodott erről a kísérleti beállításról, biztosítva a szisztematikus tesztelést különböző adatbázisokon. A szerző elvégezte az adatelőkészítést és integrálta a kimenetet SVM-alapú osztályozásba. Többszörös iterációs kísérleteket végzett az aggregációs módszerek és a különböző rétegek által nyújtott jellemzők hatásának elemzésére. Elvégezte a konfigurációk összehasonlítását és azonosította az optimális rétegbeállításokat és aggregációs technikákat, miközben dokumentálta a megállapításokat.

Publications

Journal publications

Vetráb, M. & Gosztolya, G. (2022). Using the Bag-of-Audio-Words approach for emotion recognition. Acta Universitatis Sapientiae, Informatica, 14(1), 2022. 1-21 [94] https://doi.org/10.2478/ausi-2022-0001

Vetráb, Mercedes & Gosztolya, Gábor. (2023). Using Hybrid HMM/DNN Embedding Extractor Models in Computational Paralinguistic Tasks. Sensors. 23. 5208 [96] https://doi.org/10.3390/s23115208

Full papers in conference proceedings

Vetráb, M., Gosztolya, G. (2020). Investigating the Corpus Independence of the Bag-of-Audio-Words Approach. In: Text, Speech, and Dialogue. TSD 2020. Lecture Notes in Computer Science(), vol 12284. Springer [92] https://doi.org/10.1007/978-3-030-58323-1_31

M. Vetráb and G. Gosztolya, Handling the stochastic behaviour of the Bag-of-Audio-Words method, 2022 IEEE 20th Jubilee World Symposium on Applied Machine Intelligence and Informatics (SAMI), Slovakia, 2022, pp. 000021-000026 [93] https://doi.org/10.1109/SAMI54271.2022.9780776

M. Vetráb et al., Using Spectral Sequence-to-Sequence Autoencoders to Assess Mild Cognitive Impairment, In: ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2022, pp. 6467-6471 [90] https://doi.org/10.1109/ICASSP43922.2022.9746148

Vetráb, M., Gosztolya, G. (2023). Aggregation Strategies of Wav2vec 2.0 Embeddings for Computational Paralinguistic Tasks. In: Speech and Computer. SPECOM 2023. Lecture Notes in Computer Science(), vol 14338. Springer [95] https://doi.org/10.1007/978-3-031-48309-7_7

Kiss-Vetráb, M. és José Vicente, E. és Balogh, R. és Imre, N. és Hoffmann, I. és Tóth, L. és Pákáski, M. és Kálmán, J. és Gosztolya, G. (2022) Enyhe kognitív zavar automatikus felismerése szekvenciális autoenkóder használatával. In: XVIII. Magyar Számítógépes Nyelvészeti Konferencia. Szegedi Tudományegyetem, Informatikai Intézet, Szeged, pp. 175-184. ISBN 9789633068489 [46]

Mercedes, V., & Gábor, G. (2019). Érzelmek felismerése magyar nyelvű hangfelvételekből akusztikus szózsák jellemzőreprezentáció alkalmazásával. In: XV. Magyar Számítógépes Nyelvészeti Konferencia. Szegedi Tudományegyetem Informatikai Intézet, Szeged, pp. 265-274. ISBN: 9789633153932 [55]

Vetráb, Mercedes és Gosztolya, Gábor (2020) Az akusztikus szózsák eljárás korpuszfüggetlenségének vizsgálata. In: XVI. Magyar Számítógépes Nyelvészeti Konferencia. Szegedi Tudományegyetem, Informatikai Intézet, Szeged, pp. 219-231. ISBN 9789633067192 [91]

Further related publications

Gosztolya, G., Vetráb, M., Svindt, V., Bóna, J., & Hoffmann, I. (2024). Wav2vec 2.0 Embeddings Are No Swiss Army Knife – A Case Study for Multiple Sclerosis. Interspeech 2024, 2499–2503 [27]
https://doi.org/10.21437/Interspeech.2024-995

Egas-López, J. V., Vetráb, M., Tóth, L., & Gosztolya, G. (2021b). Identifying Conflict Escalation and Primates by Using Ensemble X-Vectors and Fisher Vector Features. Interspeech 2021, 476–480 [14]
https://doi.org/10.21437/Interspeech.2021-1173

Acknowledgments

First of all, I would like to thank my supervisor, Gábor Gosztolya, for directing my PhD studies. I would also like to thank my colleagues who helped me to conduct the researches presented here and to enjoy the period of my studies. Last, but not least, I wish to thank my husband and family for their constant love and support.