Machine Learning for Non-Intrusive Speech Quality Assessment

by

Mona Hakami

A thesis
submitted to the Victoria University of Wellington
in fulfilment of the
requirements for the degree of
Doctor of Philosophy
in Engineering and Computer Science.

Victoria University of Wellington
2021

Abstract

This thesis presents two studies on non-intrusive speech quality assessment methods. The first applies supervised learning methods to speech quality assessment, which is a common approach in machine learning based quality assessment. To outperform existing methods, we concentrate on enhancing the feature set. In the second study, we analyse quality assessment from a different point of view inspired by the biological brain and present the first unsupervised learning based non-intrusive quality assessment that removes the need for labelled training data.

Supervised learning based, non-intrusive quality predictors generally involve the development of a regressor that maps signal features to a representation of perceived quality. The performance of the predictor largely depends on 1) how sensitive the features are to the different types of distortion, and 2) how well the model learns the relation between the features and the quality score. We improve the performance of the quality estimation by enhancing the feature set and using a contemporary machine learning model that fits this objective. We propose an augmented feature set that includes raw features that are presumably redundant. The speech quality assessment system benefits from this redundancy as it results in reducing the impact of unwanted noise in the input. Feature set augmentation generally leads to the inclusion of features that have non-smooth distributions. We introduce a new pre-processing method and re-distribute the features to facilitate the training. The evaluation of the system on the ITU-T Supplement23 database illustrates that the proposed system outperforms the popular standards and contemporary methods in the literature.

The unsupervised learning quality assessment approach presented in this thesis is based on a model that is learnt from clean speech signals. Consequently, it does not need to learn the statistics of any corruption that exists in the degraded speech signals and is trained only with unlabelled clean speech samples. The quality has a new definition, which is based on the divergence between 1) the distribution of the spectrograms of test signals, and 2) the pre-existing model that represents the distribution of the spectrograms of good quality speech. The distribution of the spectrogram of the speech is complex, and hence comparing them is not trivial. To tackle this problem, we propose to map the spectrograms of speech signals to a simple latent space.

Generative models that map simple latent distributions into complex distributions are excellent platforms for our work. Generative models that are trained on the spectrograms of clean speech signals learned to map the latent variable Z from a simple distribution P_Z into a spectrogram X from the distribution of good quality speech. Consequently, an inference model is developed by inverting the pre-trained generator, which maps spectrograms of the signal under the test, X_t, into its relevant latent variable, Z_t, in the latent space. We postulate the divergence between the distribution of the latent variable and the prior distribution P_Z is a good measure of the quality of speech.

Generative adversarial nets (GAN) are an effective training method and work well in this application. The proposed system is a novel application for a GAN. The experimental results with the TIMIT and NOIZEUS databases show that the proposed measure correlates positively with the objective quality scores.

Acknowledgments

I am grateful to everyone who has helped me to achieve my dream of completing this PhD. I would like to express my sincere gratitude to my supervisor, Prof. Bastiaan Kleijn, for the patient guidance and advice he has provided throughout my time as his student. I would like to thank my second supervisor Prof. Dale Carnegie and also Dr. Diana Siwiak for showing faith in me and their incredible guidance. It would not have been possible for me to bring this work to completion without their encouragement and words of wisdom.

I cannot begin to express my gratitude to my family and my friends for all of the love, support, encouragement and prayers they have sent my way along this journey. I would like to thank Saman for being beside me and for his patience through the toughest moments of my life.

I owe this thesis to my mother who stood behind me, loving me unconditionally, and supporting me in any way she could, in particular taking care of my children. My mother dedicated her time and love to me during my thesis, and I would never be able to pay back the support and affection I received. I only can pass this love forward and so would like to dedicate this thesis to my son and daughter, Matine and Araminta. I appreciate how you both abide by my ignorance and I apologise for every single night you slept without me and every morning you woke up when I was not there. Words can never say how grateful I am to both of you.

Publications of the thesis

The contents of this thesis have been published in the form of the following paper and patent,

I M. Hakami and W. B. Kleijn. "Machine learning based non-intrusive quality estimation with an augmented feature set", in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017. Winner best student paper award (second prize).

II W. Xiao, M. Hakami, and W. B. Kleijn, "Devices and methods for evaluating speech quality," Apr. 20 2021, U.S. Patent 10,984,818.

Acronyms

ACR	Absolute category rating
AE	Auto-encoder
ALI	Adversarially learned inference
ANIQUE	Auditory non-intrusive quality estimation
AS-VAE	Adversarial symmetric variational auto-encoders
BEGAN	Boundary equilibrium generative adversarial networks
BiGAN	Bidirectional GANs
BLSTM	Bidirectional long short-term memory
BSD	Bark spectral distance
CGAN	Conditional GAN
CNN	Convolutional neural network
DBN	Deep belief networks
DCGAN	Deep convolutonal GAN
DCNN	Deep convolution neural network
EBGAN	Energy-based generative adversarial network
EM	Earth mover
GAN	Generative adversarial net
PM	Integral probability metric
ITU-T	International Telecommunication Union
JS	Jensen Shannon divergence
KL	Kullback–Leibler divergence
LSGANs	least squares generative adversarial networks
MAP	Maximum a posteriori

ML	Machine learning
MLP	Multi-layer perceptron
MMD	Maximum mean discrepancy
MNB	Measuring normalizing block
MOS	Mean opinion score
PCA	Principal component analysis
PCC	Pearson correlation coefficient
PESQ	Perceptual evaluation of speech quality
PSQM	Perceptual speech quality measure
QA	Quality assessment
RBM	Restricted Boltzmann machines
RGAN	Relativistic GANs
RKHS	Reproducing Kernel Hilbert spaces
RMSE	Root mean squared error
RNNs	Recurrent neural networks
RVM	Relevance vector machine
SBL	Sparse Bayesian learning
SGAN	Stacked generative adversarial networks
SNR	Signal-to-noise ratio
STFT	Short-time Fourier transform
SVM	Support vector machine
VAE	Vrational auto-encoders
WGAN	Wasserstein GAN
WGAN-GP	Wasserstein GAN with gradient penalty
WSSD	Weighted slope spectral distance
XNV	Correlated Nystrom views

Contents

1

Introduction

After more than a century of experience with speech transmission in telecommunication, it may come as a surprise that an automatic assessment of the quality of speech is still an issue. This chapter is an introduction to the problem of non-intrusive speech quality assessment. Section 1.1 explains why developing a reliable estimation of speech quality is critically important and presents the motivations for developing non-intrusive quality assessment systems based on machine learning methods. Section 1.2 provides a brief overview of the approaches utilised and the principles adopted for the machine learning based systems proposed in this thesis. Section 1.3 presents the most important contributions of this thesis. The structure of the thesis is outlined in Section 1.4.

1.1 Motivation

Over the last few decades, the telecommunication industry has grown rapidly. Many different services exist, and customers generally have many

options to choose from. In telecommunication, particularly in speech transmission, the success of speech processing applications depends on the opinion of end-users about the perceived speech quality [1]. Therefore, a reliable valid estimation of speech quality has become critically important. Speech quality assessment systems enable the service providers and the developers of the new services to assess and monitor the quality of service on a regular basis.

The most valid method for assessing voice quality is *subjective* assessment [2, 3]. In subjective assessment, human subjects are asked to listen to the transmitted speech utterances and score their quality. Hence, subjective tests, in general, are costly and time-consuming. Consequently, *objective* quality assessment algorithms, which provide an automatic assessment of voice quality, are more desirable [1].

Initial objective algorithms [4, 5, 6, 7] estimated the distortion introduced by the system under test by comparing the degraded signal that is processed by the system with the original undistorted signal [8]. The original undistorted signal is called the reference signal. These algorithms that require both the reference signal and the degraded signal are called *full-reference* or *intrusive* methods.

In contrast to intrusive methods, *non-intrusive* methods that are sometimes called *single-ended* [9, 10, 11] do not depend on a reference signal. Non-intrusive methods are essential tools for online applications, such as monitoring speech quality of in-service systems, where the source speech signal is not available [1]. However, the performance of non-intrusive models is generally lower than the intrusive models. Furthermore, the design of a non-intrusive system is normally complicated and is based on training the model on a database created with human subjects [1].

The standard non-intrusive algorithms (e.g. [12]) mostly attempt to find an explicit relation between the audio signals and their quality. The parameters of such methods are based on a database with speech utterances and their rating. This is challenging and often requires quality as-

sessment specialists to determine the contribution of each feature and their interaction to the overall audio quality. In contrast, machine learning based quality assessment methods (e.g. [13]) attempt to mimic quality perception and avoid designing an explicit model. These machine learning based methods that replace the knowledge of assessment specialists with "supervised learning" algorithms still rely on databases of speech and associated quality ratings from human subjects. However, these methods automate the training by replacing human expertise with "supervised learning" algorithms. As will be explained in Chapter 2, such systems are beneficial mainly because they are not limited to particular services and are adaptable to multiple applications. For example, a system developed for narrow-band data can be used for wide-band data if we re-train that system with a wide-band database.

In this thesis, we develop two machine learning based non-intrusive speech quality assessment systems. The first one, similar to earlier machine learning based work, is based on supervised learning methods. Recent supervised learning based quality assessment methods [13, 14, 15, 16, 17, 18] are mainly improved by either employing more data or applying more powerful learning algorithms for training. Recent powerful machine learning algorithms [19, 20, 21, 22] that are mostly based on deep learning also require large databases for training. Hence, accessing an extensive training database that contains speech utterances and their relative ratings is a key to the success of many machine learning based speech quality assessment systems. However, free data available for training in this field is limited. The motivation for the first approach we propose in this thesis is to improve the performance of machine learning based speech quality assessment systems by improving the input features rather than enlarging the training data.

The motivation for the second approach is to remove the need for expensive training data that includes the subjective rating of the speech utterances. We do this by applying "unsupervised learning" algorithms.

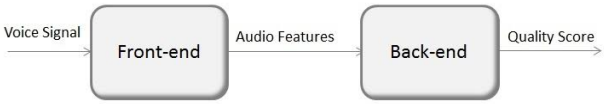


Figure 1.1: High level structure of a non-intrusive quality assessment system.

Our proposed speech quality assessment is the first unsupervised learning based approach in this field. In the following section, we briefly address this gap in the literature. We also provide an introductory explanation to the approaches and principles of applying "machine learning" into the field of "non-intrusive speech quality assessment".

1.2 Approaches and principles

As will be explained in Chapter 4, in supervised learning terminology, non-intrusive quality estimation can be described as a multi-class classification or a regression problem, where the input and output are the signal features and the quality score respectively [1]. Several non-intrusive methods have recently been proposed in the context of quality assessment, using machine learning algorithms for estimating the score of audio signals [23, 24, 25, 26, 27, 28, 29, 30, 31]. Figure (1.1) shows the high-level structure of a non-intrusive quality assessment system.

In the front-end module, a feature vector containing the attributes describing the audio is constructed. This is a common practice because a model with thousands of samples per second of audio as input will generally be very complicated and several conventional machine learning algorithms cannot be efficiently trained on high-dimensional data.

In the back-end module, the features are mapped into a quality score. In many protocols, the human subjects provide a discrete score, and in this case, the back-end module is a classifier. However, one might be interested

in predicting the average rating of an utterance. In that case, the back-end module becomes a regressor.

More recent speech quality assessment methods [13, 18, 32, 33] that are end-to-end use the raw audio waveform and the related spectrograms as input to the back-end module. In such systems, the front-end and back-end modules are integrated and feature extraction is part of the predicting function which is based on deep learning algorithms.

The structure in Figure (1.1) indicates that the overall performance of the supervised quality assessment system depends upon two main aspects. The first aspect is a feature set that is sensitive to signal variations due to different types of distortion. The second aspect is a rich model that can learn the complex relationship between the audio features and the quality score.

The most recent machine learning based speech quality assessment methods in the literature [13, 15, 17, 18] are mainly centred on the second aspect. Such methods improve the performance of the back-end module by enlarging training data or applying more complex learning algorithms with a more extensive set of parameters, which also depend on an enlarged training data. Reviewing these methods and analysing the scores reported in the literature for non-intrusive quality assessment confirms that the availability of training data is one important aspect that restricts the performance of the quality predictor.

In machine learning based speech quality assessment, training data refers to speech utterances and their subjective scores. Such databases are called labelled databases as the subjective scores are the labels for the speech utterances. Unfortunately, labelled training databases are mostly proprietary and creating one is costly and time-consuming as it requires the set-up of subjective evaluation experiments. Hence, in the first phase of this thesis, we concentrate on the first aspect and instead of improving the back-end that is dependant on the availability of labelled data, we improve the performance of the system by enhancing the features extracted

from the front-end module. We study the effect of including more features on the performance of the machine learning based speech quality assessment and analyse the performance gain from our proposed enhanced feature set.

In the second phase, we seek to find a more comprehensive solution to the problem of the availability of labelled data for the development and training of the back-end module. The ITU-T coded speech data set, Supplement 23 [34], is the most well-known public labelled database for speech quality assessment. Recently other public labelled databases have been introduced by the authors of [35, 36]. These databases seem to be the only public databases that contain speech utterances and their relevant ratings. Conversely, a large number of utterances are readily available if their quality score is not required. Accordingly, it is beneficial to develop a speech quality assessment that learns from unlabelled data. The semi-supervised quality assessment systems introduced by the authors of [37, 38, 39] benefit from these public data for feature extraction. However, the quality predictor at the end still relies on the databases that contain speech utterances labelled by the subjects. In this thesis, we present the first unsupervised machine learning based non-intrusive quality assessment that removes the need for labelled training data.

The unsupervised speech quality assessment we propose in this thesis is inspired by the functionality of the biological brain. Individuals naturally have an opinion about the quality of input signals based on the pre-existing model in their brains that is built on their listening habits. We postulate that quality of speech is correlated with the similarity between what is heard and the model of speech that exists in the brain.

In this approach we utilise Generative Adversarial Nets (GANs) [20] to build a generic model of speech . Then we develop an inverted generator to project the signals into the latent space and rate the quality based on the distance between the test signal and the distribution of good quality speech in the latent space. In Chapter 5, we explain this approach in

more detail and verify its effectiveness. The following section presents the contribution of this thesis.

1.3 Contributions of the thesis

The most important contributions of this thesis are summarised as follows:

- We introduce the novel idea of augmenting the feature set with raw features that are presumably redundant. We report on the case where input features are noisy and illustrate that the proposed augmented feature set improves the performance by reducing the effect of input noise. We provide a detailed analysis of this performance gain and its mathematical model (Chapter 4).
- We present a new pre-processing method that redistributes the features to have a smooth distribution. We explain the pre-processing method in detail and demonstrate that it facilitates the training of quality assessment (Chapter 4).
- We build a new quality assessment system based on supervised learning algorithms and use the enhanced feature introduced in this thesis. The experimental results confirm the performance gain from the enhanced feature set and demonstrate that the proposed system outperforms the current methods in the literature (Chapter 4).
- We introduce the first unsupervised quality estimation system (Chapter 5).
  - Data used for training is standard speech signals, which are not required to be labelled by subjects.
  - We use a new quality metric, which is based on how different the input is from the pre-existing model trained on clean speech signals. Therefore, the predictor does not need to learn

the statistics of degraded speech files and in principle is not limited to specific types of corruptions in the training data.

- Our system is based on the novel idea of employing the generative models for quality assessment. We use the generative adversarial net (GAN) to mimic the pre-existing models in the brain.
- We utilise divergence metrics to measure the distance between the prior distribution of the good quality signals and the distribution of the test signal in the latent space. We use this criterion to assess the quality of audio and demonstrate that it is highly correlated with the scores from subject tests.

1.4 Structure of the thesis

The remainder of this thesis is organised as follows:

In Chapter 2, we provide an introductory explanation as the background to machine learning based non-intrusive speech quality assessment and review the approaches that are proposed for speech quality assessment. In Chapter 3, we briefly review the machine learning basics and the contemporary methods used in this thesis.

In Chapter 4 we investigate supervised learning for speech quality assessment. We propose a new non-intrusive speech quality assessment based on a neural network and demonstrate the performance gain from the enhanced feature set. To achieve the higher performance we introduce two novel enhancement procedures to the feature set: 1) Augmentation, 2) Standardization. We published the main part of Chapter 4 in 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) and won a best student paper award [1].

In Chapter 5, we focus on unsupervised learning for speech quality assessment. We introduce a novel application for the popular generative

model called a GAN and build our quality assessment based on the correlation of data points in the input manifold of the generator.

In Chapter 6 we provide a summary of this thesis and discuss its outcomes. We address the shortcomings and the potential solutions for them to extend this work in the future.

2

Background on speech quality assessment

This chapter presents a brief background of basic concepts in speech Quality Assessment (QA) and reviews the related work followed by the detailed description of two current standards in this field.

2.1 Introduction

Although multimedia has grown during the last decades, speech is still one of the main media of communication between humans. Speech is also increasingly used for human-machine interactions. There are numerous service providers for customers to choose from, and many of these services involve different far-end and near-end environments and multiple links over different networks. In all these services, it is essential to ensure a high quality of speech, and hence a reliable system to estimate the quality is required.

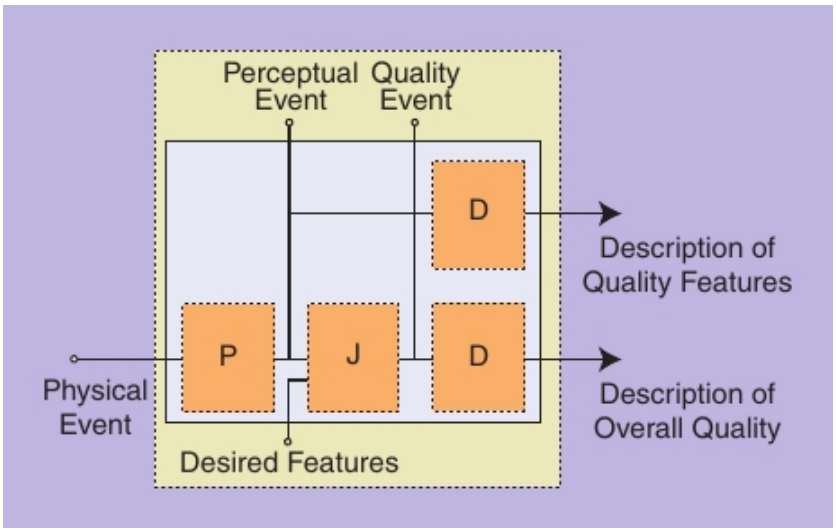


Figure 2.1: Schematic representation of a participant in a quality judgment experiment from Figure (1) in [40].

As shown in Figure 2.1, speech quality can be viewed as the result of three processes: perception (P), judgement (J), and description (D). The perception process is triggered when the sound wave reaches the human ears, which is called "physical event" (the term "event" here specifies the instance of a phenomenon that occurs in time and space) [40]. The result of the perceptual process is a "perceptual event", which can be described by features such as loudness, coloration, or noisiness that are quantified. In the judgement process, the perceptual event is compared to the desired features and results in a "quality event", which represents the judgement of the overall quality. The description process quantifies the quality event and expresses the judgement in terms of opinion scores. As denoted by the dotted line containing the yellow square in Figure 2.1, the processes, the events, and the desired features are internal to the perceiving human [40]. The models developed in this thesis simulate this internal scheme to replace quality assessment judgment experiments (which are laborious) with an automatic mechanism.

Measurements of quality have many dimensions such as intelligibility, naturalness, clarity, pleasantness, brightness [41]. A speech quality assessment system may aim to evaluate the overall quality, or they may measure individual dimensions and predict multiple quality features [42]. A single metric does not generally provide sufficient detail for system designers, so it might not be satisfactory for network planning purposes. Although single metric based models are not suitable for diagnosing the sources of poor quality, they are sufficient for network monitoring as it gives an overall perception of an auditory event. In general, the use of a single metric is more common than the use of a multidimensional metric [41, 42]. For practical purposes, the focus of this thesis is on the models with a single metric. The quality assessment models that are proposed in Chapter 4 and 5 provide an overall perception of an auditory event, which is sufficient to predict the end-user opinion of a speech communication system.

Speech quality may refer to a purely listening-only situation, or it may reflect a conversational situation where both sides are talking (and hence there are constant changes between talking and listening) [43]. The *true* speech quality is often addressed as conversational quality since it is the common application of speech services. In conversational tests, two people are usually questioned about the quality aspects of the conversation after they had a conversation over the system under the test. However, because of the complexities involved in conversational tests, the most frequently measured quantity is the listening quality [30, 31, 42, 43]. In the listening context, which is the focus of this thesis, the speech quality is mainly affected by speech distortion due to speech codecs, background noise, and packet loss, whereas in the conversations, the impact of other degradations such as talker echo and path delay must also be considered.

This chapter will provide a description of speech quality assessment. In Section 2.2, we review the literature related to quality assessment, introduce different types of such models, and explain why we are interested in non-intrusive objective algorithms. The quality estimation methods P.563

[12] and ANIQUE+ [44] are established standards for single-metric non-intrusive objective quality assessment, which are designed for listening quality. As will be explained in Section 4.5, the front-end modules of these two standards are the reference for our work. These two standards are explained in more detail in Sections 2.3 and 2.4.

2.2 Relevant works

As explained in the previous section, the focus of this thesis is speech quality assessment in the listening context, which aims to measure an individual dimension of quality using a single metric. This section presents an overview of methods in this context.

Subjective assessment is the most valid method for assessing voice quality, in which human subjects are asked to listen to the speech utterances and to score their quality (see Figure 2.2). One of the most widely used listening tests is an Absolute Category Rating (ACR) method, described in the International Telecommunication Union (ITU-T) Recommendation P.800 [2]. In this test, a number of subjects are asked to rate the quality of a number of short speech sentences processed by the system under test in a five-point scale (5: excellent, 4: good, 3: fair, 2: poor, and 1: bad). The average rating is commonly referred to as Mean Opinion Score (MOS) [10].

In general, conducting a subjective test is based on the recruitment of human participants who sit in a laboratory and listen to the test materials to evaluate their quality under certain conditions specified in the standard. Since subjective tests require human listeners, they are generally time-consuming and expensive. As a result, objective quality assessment algorithms were introduced to provide an automatic assessment of voice quality [1]. These methods replace the listener panel with a computational algorithm. However, listening tests are still required for the development and training of the objective quality assessment algorithm.

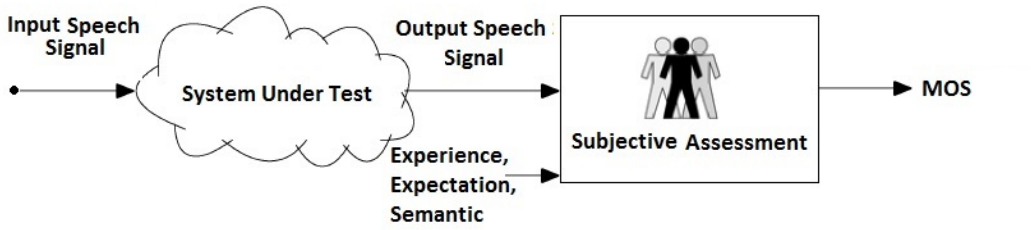


Figure 2.2: Subjective listening quality assessments, adopted from Figure (1) in [8].

Although some models have been designed to predict individual quality features such as discontinuity, noisiness, coloration, and intelligibility [33, 45], most single-metric objective methods provide quality estimations according to the MOS scale in ACR [40]. In these scenarios, objective methods aim to deliver estimated MOS values that are highly correlated with the MOSs obtained from subjective listening experiments [46]. The estimated MOS value is called *objective* MOS, and the MOS obtained from the subjective listening test is called *subjective* MOS [47]. The measure for the success of an objective method is based on comparing the objective MOS predicted from the method with the subjective MOS. Pearson Correlation Coefficient (PCC) and Root Mean Squared Error (RMSE) are common metrics in this field [7, 12]. RMSE measures the closeness of objective MOS to subjective MOS based on the mean square of the residual errors, and PCC measures the closeness of their fit based on their correlation. These two standard metrics are explained in Section 4.6.1. In general, a non-intrusive quality assessment is considered to have higher performance on a test database in comparison with another method if the computed PCC and the RMSE are relatively higher and lower than the PCC and RMSE of the other method, respectively.

In [40], an objective model is classified as either a signal-based model, a parametric model, or a hybrid model. In signal-based models such as

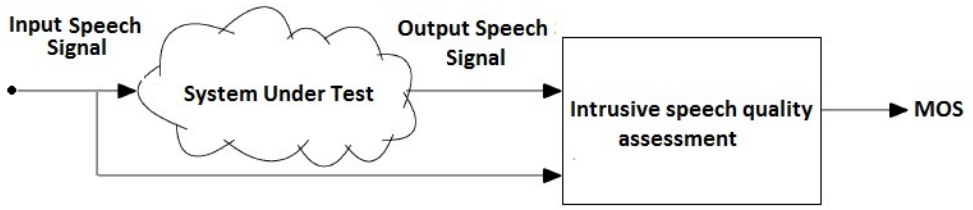


Figure 2.3: Objective intrusive listening quality assessments, adopted from Figure (1) in [8].

[12, 44, 45], the estimation of the quality of speech only depends on the voice signal and the parameters extracted from speech. On the other hand, parametric models such as [48, 49] depend on system parameters estimated at run time or during the network planning. One significant advantage of the parametric models over signal-based models is that they are employed at design time when the system is not implemented. The parametric models remove the need for a prototype implementation of the transmission channel and the simulation of the signals, which is necessary for the signal-based model. Subsequently, Hybrid models such as [50] combine both concepts from signal-based and parametric models and make use of both types. Hybrid models depend on both signal and system parameters and benefit from diverse information that is readily or economically accessible or more reliable [40].

The focus of this thesis is the building of a generic non-intrusive method that is not dependent on a specific system. Hence the system parameters are not reflected, and both methods proposed in Chapter 4 and 5 are signal-based models that are platform-independent. In the following, we review the quality assessment systems in the literature in this context.

Karjalainen originally introduced a first perceptual objective quality assessment algorithm in 1985 [4]. Since then, many objective speech quality estimation models have been proposed. Some examples are Weighted-

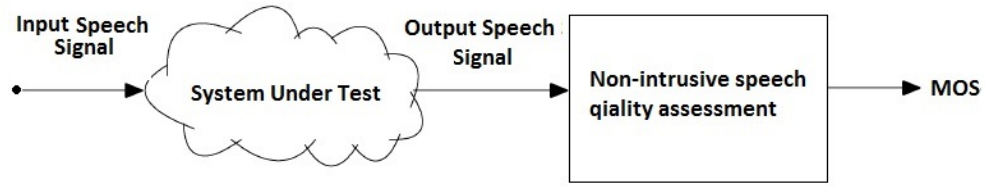


Figure 2.4: Objective non-intrusive listening quality assessments, adopted from Figure (1) in [8].

Slope Spectral Distance (WSSD) [5], Bark Spectral Distance (BSD) [51, 52], Perceptual Speech Quality Measure (PSQM) [6], Measuring Normalizing Block (MNB) [53, 54], Perceptual Evaluation of Speech Quality (PESQ) [7], and Perceptual Objective Listening Quality Assessment (POLQA) [55]. To estimate the distortion introduced by the system under test, these algorithms compare the signal processed by the system (degraded signal) with the original undistorted signal (reference signal) [8]. Such algorithms are generally called full-reference (or double-ended) models and are considered to be *intrusive* as they require both a reference and a degraded signal (see Figure 2.3) [56].

In contrast with subjective experiments, objective algorithms enable extensive testing to be performed over short periods [8]. They can also be used for network monitoring by injecting test signals into a communication network [8]. However, the need for a reference signal adds extra load on the network. Furthermore, live calls cannot be evaluated since clean reference signals are not available [8].

To eliminate the need for a reference signal in full-reference methods, non-intrusive methods, which do not depend on a reference signal, are introduced (see Fig.2.4). Non-intrusive methods (sometimes called single-ended methods) are essential tools for monitoring speech quality of in-service systems, where the source speech signal is not available [1]. How-

ever, the design of a non-intrusive model is more complicated than that of an intrusive model, and its performance is generally lower than systems that use a reference signal [1].

In 1994-1996 attempts were made to build non-intrusive measurement systems by comparing the features of the received speech signal and a set of codewords derived from the undegraded source [9, 57]. Unlike the previous methods that focused on the auditory system to assess the perceived quality, the method in [11] focuses on the speech production system. This method uses the parameterisation of a vocal tract model and applies high order statistical analysis on them. It identifies telecommunication network distortions by identifying the states that are unlikely to be produced by the vocal tract.

In 2002, the ITU-T opened a competition to provide a standard non-intrusive method that does not require a reference signal for quality estimation [8]. A collaboration between three quality assessment specialised companies, Psytechnics, Ltd., Swissqual, and Opticom, resulted in the definition of a new standard in May 2004, which is known as ITU-T Recommendation P.563 [8]. In 2007 a new American national standard known as ANIQUE+[44] was introduced, which outperforms the P.563 standard. P.563 and ANIQUE+ are the current established standards for single-metric non-intrusive quality assessment in a listening situation. They are freely available, and since they have good performance, their feature sets are expected to be informative and represent different types of distortion. As will be explained in Section 4.5, the feature set proposed in Chapter 4 contains features extracted from P.563 and ANIQUE+.

While P.563 and ANIQUE+ have demonstrated acceptable accuracy for many telecommunications scenarios, their quality prediction performance is compromised for scenarios involving noise suppression, dereverberation and wireless-VoIP tandem connections [40]. In 2010 a non-intrusive measure named Speech-to-Reverberation Modulation energy Ratio (SRMR) [45] was developed for both narrow-band and wide-band reverberant and

dereverberated speech. The SRMR metric has a relatively simple predicting function using features based on modulation envelopes of a speech signal, which are known to be useful cues for objective speech quality and intelligibility estimation. SRMR outperforms P.563 and ANIQUE+ for tasks involving estimation of multiple dimensions of perceived coloration, as well as quality measurement and intelligibility estimation of reverberant and dereverberated speech [45].

The normalized SRMR ($\text{SRMR}_{\text{norm}}$) [58] proposed updates in SRM to reduce the effects of pitch and speech content on the SRMR metric. Compared to the original SRMR implementation, $\text{SRMR}_{\text{norm}}$ led to improved speech intelligibility prediction and exhibited lower variability. Both SRMR and $\text{SRMR}_{\text{norm}}$ use the same predicting function, which is equal to the ratio between the energy in the lower and higher modulation frequencies. Recent speech quality approaches have focused on applying machine learning techniques to train the predicting function of the system. The method proposed in [16], improved $\text{SRMR}_{\text{norm}}$ by using the same modulation-based features but with a different predicting function based on a model tree trained with a small corpus of speech data.

The work described in [16] is based on a machine learning method that learns from the training data, and unlike P.563 and SRMR avoids designing an explicit model for mapping signal features to the quality score. Such machine learning based models [14, 46, 59] propose non-intrusive evaluation algorithms based on the statistical model approach and mimic the quality perception to estimate the subjective Mean Opinion Score (MOS). Machine learning based quality assessment systems are desirable as they are adaptable to various applications and hence are not restricted to particular services [1].

The machine learning based non-intrusive systems proposed in the literature predicts the quality by applying a predicting function to a feature vector extracted from the processed signal. These systems may differ in the considered features, the predicting function, or both. The work de-

scribed in [60] makes use of a classifier to predict the discrete value of quality score while the works described in [23, 24, 25, 26, 27, 28, 29, 31, 61, 62] apply regression methods (with shallow architectures) to estimate the subjective Mean Opinion Score (MOS) assigned to a speech file. On the other hand, approaches in [16, 30] use a combination of classification and regression algorithms as the predicting function. More recent works [13, 15, 17, 18, 63, 64, 65] focus on deep machine learning methods, which are shown to be more powerful. Another advantage of methods such as [13, 18] is that the raw audio waveform and the related spectrograms are used as input to the predicting function. Accordingly, feature extraction is part of the overall system, and hence features are expected to be more informative about the quality.

The machine learning based works cited above are based on supervised learning and need a database with speech, and their rating, for training. The non-intrusive method proposed in Chapter 4 is similar. Chapter 4 compares the performance of the proposed method with those methods from the list cited above that report results on the public database. The experimental results in Chapter 4 show that the proposed method has a higher PCC than the others.

The labelled data that are publicly available for development and training of supervised learning based quality assessment is limited, and collecting more training data is usually expensive. The ITU-T coded speech data set, Supplement 23 [34], is the most well-known public labelled database that is commonly used for training or the evaluation of objective speech quality systems [14, 23, 24, 26, 28, 29, 44, 8, 46, 61, 62, 64, 66]. Supplement 23 database contains speech affected by noise, packet loss and various codecs and their corresponding subjective quality score. Speech utterances in Supplement 23 have been down-sampled to 16 kHz. In [35], a new public data set TCD-VoIP has been created, which allows comparison of quality assessment systems with regards quality issues that occur in Voice-over-Internet Protocol (VoIP). TCD-VoIP, which is used in [17, 65, 67]

to compare quality metrics, contains wide-band speech that is corrupted with platform-independent VoIP degradations along with subjective quality scores. The five types of VoIP degradations in this database are independent of the hardware, network or codec in use and listed as 1) background noise, 2) intelligible competing speakers, 3) echo effects, 4) amplitude clipping, 5) and choppy speech.

Supplement 23 and TCD-VoIP have limited data samples. On the other hand, proprietary data sets that are large in size (for example the ones used in [14, 15, 16]) are not publicly available. Since large data sets with human subjective scores are not publicly available, recent methods focus on estimating objective scores that are computed with intrusive methods. For example, the works in [18, 30] are trained with speech files that are rated with PESQ and hence aim at predicting the PESQ score. Similarly, the works in [13, 63, 68] are capable of predicting POLQA scores. However, since objective measures can only approximate human perception, using objective quality scores as training targets is a significant limitation, and hence recently [36] conducted a large-scale listening test on real-world data and collected 180,000 subjective quality ratings through Amazon's Mechanical Turk (MTurk) [69].

Due to the limitation of the available labelled data, having an unsupervised learning based quality assessment that eliminates the requirement for labelled data is desired. The authors of [70] used deep learning to learn the features of speech spectra in an unsupervised manner. They modified the architecture of an auto-encoder and used the features extracted from a subband autoencoder for non-intrusive objective quality assessment. However, such systems are described as semi-supervised as the non-intrusive method at the end, which maps the features to the score, still requires labelled data. To our knowledge, the method we propose in Chapter 5 is the first method that removes the need for labelled training data.

In the following two subsections, we explain P.563 and ANIQUE+,

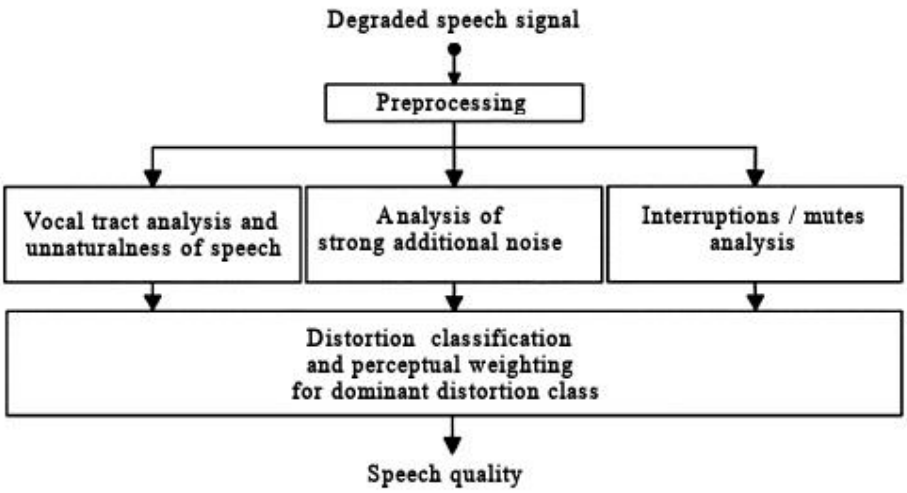


Figure 2.5: The architecture of ITU-T Recommendation P.563, adopted from [12].

which, as previously mentioned, are two current standards for non-intrusive listening quality assessment. As explained, their front-end modules form the basis for the method introduced in Chapter 4.

2.3 P.563

ITU-T standard P.563 [12] is an established standard method for single-ended speech quality assessment that is developed and used for the evaluation of narrow-band speech codecs. The P.563 standard represents one of the three current well known non-intrusive quality measurements and is widely used as benchmark for developing new metrics [36, 45, 58, 64, 65].

In the P.563 standard [8] the distorted signal is first pre-processed, then passed through a distortion estimation stage and finally a subsequent perceptual mapping stage. In the final stage, several features that represent parameters of the speech signal under test are used to predict speech quality by applying a predicting function, which combines decision rules as

well as a linear combination of features [16]. This procedure implies that the distortions to be discovered have to be known in advance, and unknown distortions (e.g., new codecs, signal enhancement elements in the system etc.) cannot be detected and considered in the output quality score [71]. This is the major disadvantage of P.563 (as well as ANIQUE+ and other non-intrusive systems that are based on a similar concept). For example, although the algorithm has demonstrated acceptable accuracy for transmission systems with echo cancelers [12], other research [72] has reported poor performance for reverberant and dereverberated speech [45]. Additionally, P.563 is designed explicitly for narrow-band speech and hence, cannot assess the quality of wide-band or super-wide band speech. The motivation of the method proposed in Chapter 4 is to overcome these problems by building a quality assessment that is based on supervised learning algorithms. The advantage of supervised learning based quality assessment in this context, is that they permit learning new types of distortion providing speech utterances degraded with those types of distortions are available for training.

Furthermore, P.563 starts with established features, which might constrain the solution space accordingly [64]. This can affect the performance of the standard considering some sources of variability might not be taken into account. Hence another point of improvement to consider is to enhance the feature set. In Chapter 4, we propose to build an enhanced feature set using the raw features from P.563 and ANIQUE+. In the following, we explain P.563 in more detail.

Figure (2.5) shows the structure of the P.563 model [8]. As noted, the model consists of three stages in this model [8]: 1) the preprocessing stage, 2) the distortion estimation stage, which in [12] is modelled as a combination of three basic principles for evaluating distortions, and 3) the perceptual mapping stage.

In the first section of this model, the degraded speech signal is preprocessed, and then distortion parameters are calculated in the second

stage. The majority of distortion parameters are computed based on the filtered signals that are the outputs of the preprocessing stage. Finally in the last stage, these parameters are linearly weighted, and the listening speech quality is estimated. In the following, we briefly review the modules in Figure (2.5).

2.3.1 Preprocessing

In order to compute the distortion parameters in the second stage, each signal is first preprocessed. The preprocessing begins with applying a filter in the frequency domain. The filter characteristic is similar to the modified IRS receive characteristic given in ITU-T Rec. P.830 [73]. IRS is an intermediate reference system [74] that represents the characteristics of a standard telephone handset.

The IRS filtered signal is later used to compute parameters that are based on the assumption that the subjective tests have been carried out using a standard telephone handset [12]. Following that a speech level adjustment to –26 dBov is applied. dBov or dB(overload) is the amplitude of a signal compared with the maximum which a device can handle before clipping occurs. Hence, this speech level adjustment makes sure the amplitude of the signal is always smaller than the clipping amplitude with the ratio -26db. –26 dBov approximately corresponds to –20 dBm0 [1], which is a typical nominal value for mean active speech level measured according to Recommendation P.56 [73, 75]. Once the signal level has been normalised, it is filtered using a 4^{th} order Butterworth high-pass filter at 100 Hz cut-off frequency [12]. All the parameters that do not use the IRS filtered signal use this normalised signal, except the mutes parameters that are explained in Section 2.3.4 and require the raw signal.

The preprocessing block [8] also performs Voice Activity Detection (VAD)

[1] dBm0 is an abbreviation for the power in dBm measured at a zero transmission level. dBm or decibel-milliwatt is a unit of level used to indicate that a power level is expressed in decibels with reference to one milliwatt.

using the technique implemented in ITU-T P.862 [7]. VAD is used to identify portions of the signal that contain speech. The output of the VAD is used to estimate the speech level, which is required to normalise the signal to -26dBov. VAD information is also used in the following P.563 modules.

In the following, we explain the four subsequent P.563 modules. The first three modules compute distortion parameters. They are followed by the last module that estimates the speech quality.

2.3.2 Unnatural speech

This block looks for unnaturalness in the speech signal. Unlike most quality assessment models that focus on the auditory system, this component of P.563 focuses on the source of human voice and models the speech production system [8]. This component estimates the distortion by identifying sounds that can not be plausibly produced by the vocal tract.

The concept of using vocal tract models for quality assessment was first introduced by Gray [11]. This method is based on modelling the vocal tract as a set of acoustic tubes with section areas that vary over time. High order statistical analysis of the model identifies illegal states or variations that indicate the presence of distorted speech [8]. P.563 implemented Gray's method in this component to assess how human-like the speech is.

This component of P.563, which detects unnatural speech, computes the largest set of parameters [12]. These parameters are subdivided into two groups: 1) speech statistics and 2) vocal tract analysis.

The speech statistics parameters:

These parameters are mainly based on cepstral and LPC analyses, which are standard signal processing techniques. The two higher-order moments, kurtosis and skewness of these parameters, are computed here for further analysis of the signal properties.

The vocal tract parameters:

The second group of parameters include the parameters that are related to I) the vocal tract model analysis, II) unnatural periodicity, and III) full-reference psychoacoustic model. In the following, we explain these parameters.

I) For vocal tract model extraction, the human vocal tract is modelled as eight concatenated lossless acoustic tube sections [8]. The vocal tract parameters [8] are equal to the tube section areas, which are calculated using Linear Predictive (LP) analysis [76]. These parameters are only computed for voiced sections of speech [8]. The LP reflection coefficients [8] are calculated using the autocorrelation method [77] and the Schur recursion [78]. The resulting parameters from the eight tube sections are then averaged to reduce the information down to three parameters, which model the cavity articulators (ARTs) [8]. Consequently, a set of statistical measures are calculated on voiced sections of the entire signal, describing the size and rate of change of tube section areas and cavities as a function of time [8]. Finally, the overall vocal tract variations are estimated, which are a good basis for detecting distorted speech [8]. This is because the vocal tract is controlled by muscles and fast variations in the acoustic tube model or excessively large sections are not possible in undistorted speech [8].

II) For computing the parameters that are related to the unnatural periodicity, the signal is investigated for the occurrence of repeated speech frames and highly periodic sections that are not speech. In P.563, the signal periodicity that is described as artificial or robotic, is measured by analysing the signal in the frequency range between 2.2 and 3.3 kHz and computing the cross-correlations of short adjacent time signal frames [8]. The signal frames are then classified as periodic or non-silent [8]. The signal is declared to be robotic if the percentage of periodic frames among the non-silent frames in a signal is large [8]. Following that, this module extracts other parameters related to the occurrence of the repeated frame.

It detects the repeated frames in the signal using the usual high cross-correlation of repeated frames [12]. This module also investigates the signal for unnatural beeps. The detectors in this module identify complex tones and mark them as an unusual beep if they have a short duration [12].

III) The parameters related to a full-reference psychoacoustic model present a general description of the received speech quality based on the intrusive model. Since intrusive models are full-reference and require a reference signal, an intermediate speech reconstruction model is adopted to generate one. The speech reconstruction module recovers a quasi-clean speech signal from the degraded input signal using a speech enhancement technique [12]. The recovered signal is used as the input for the subsequent perceptual full-reference speech quality measurement model, which is a modified version of ITU-T P.862 [8]. The full-reference model that evaluates the difference between the pseudo reference signal and the degraded speech signal can only measure distortion that the speech enhancement system has removed [8]. Hence the parameters computed in this clause reflects only part of the degradation and are not accurate enough to predict speech quality [8].

2.3.3 Analysis of strong additional noise

The noise analysis calculates different characteristics of noise. Noise that is considered here can be either static and present over the whole signal (at least during speech activity) such that the noise power is not correlated with the speech signal, or the noise power shows dependencies on the signal power envelope [12]. Hence this functional block computes two subsets of parameters: 1) Static noise level and SNR and 2) Multiplicative noises and segmental SNR.

The SNR estimation is performed by calculating the level of speech and noise sections that are identified by VAD in the preprocessing stage [8]. This calculation does not reflect the noise that exists within speech

sections. Hence computing an additional set of parameters is required. For example, the local background noise parameter estimates noise occurring during speech events by locating intervals between phonemes and calculating their energy [8]. A phoneme described here is an interval in which the signal envelope doubles within 100 ms and decreases to near to its original value within 400 ms [8].

Multiplicative noise [8] is most commonly introduced to the signal by (mainly cascaded) logarithmic PCM and ADPCM systems and waveform speech CODECs [12]. Multiplicative noise, which follows the signal envelope, is detected by a separate functional block. This functional block works mainly based on the evaluation of spectral statistics [8]. It is assumed [8] that the noise has a flat spectral characteristic and forms a "noise-floor" in the spectral domain [8]. Due to its multiplicative conjunction with the speech, it is only present during speech activity and so the evaluation has to be concentrated on active speech regions [8]. The output of VAD is used to restrict the analysis to the active voice frames. This analysis is applied on the telephone bandpass filtered signa [8]l.

2.3.4 Analysis of interruptions, mutes and time clipping

Mutes and interruptions can be partly described by the outcomes of the vocal tract functional block. However, for analysing this type of degradation, a separate functional block is required to detect and rate unnatural mutes and time clippings. This functional block detects two different types of signal interruption: 1) muting or speech interruptions and 2) temporal speech clipping.

Muting or interruption of the speech is when part of the signal is removed and replaced with comfort noise or silence [8]. This type of interruption is frequent in telecommunications, especially in packet-based real-time transmission systems where packets may be lost or dropped by jitter buffers [8].

The second type is the temporal clipping of the speech sections. This

happens when a signal becomes interrupted, for example, when Voice Activity Detection (VAD) or Digital Circuit Multiplication Equipment is used. This clipping cuts off the front or back end of speech sections during the time that the transmitter is detecting the presence of speech [12].

The algorithm used in this functional block of P.563 detects and estimates unnatural silence intervals in a speech utterance by analysing abrupt variations in the signal envelope. This makes it possible to distinguish between normal speech ends and abnormal signal interruptions.

2.3.5 Dominant distortion classification and speech quality estimation

The authors of [12] observed that when different types of degradation occur simultaneously, human listeners focus on the dominant distortion. P.563 is based on this assumption and models the behaviour of a person facing multiple distortions. This model is composed of three steps: 1) decision on the main distortion class, 2) evaluation of speech quality for the corresponding distortion class, 3) overall calculation of speech quality.

The classification is performed by applying thresholds on the key parameters computed in the previous functional blocks [8]. These key parameters are namely PitchAverage, SNR, EstSegSNR, SpeechInterruptions, Sharp Declines, MuteLength, LPCcurt, Robotization If the test signal falls into more than one class, a prioritization is performed according to the annoyance order [12]. The following rank-order of the annoyance or perceptual focus was found by analyzing auditory experiments [8]:

1. high-level background noise;
2. signal interruption;
3. signal correlated noise;
4. speech robotization;

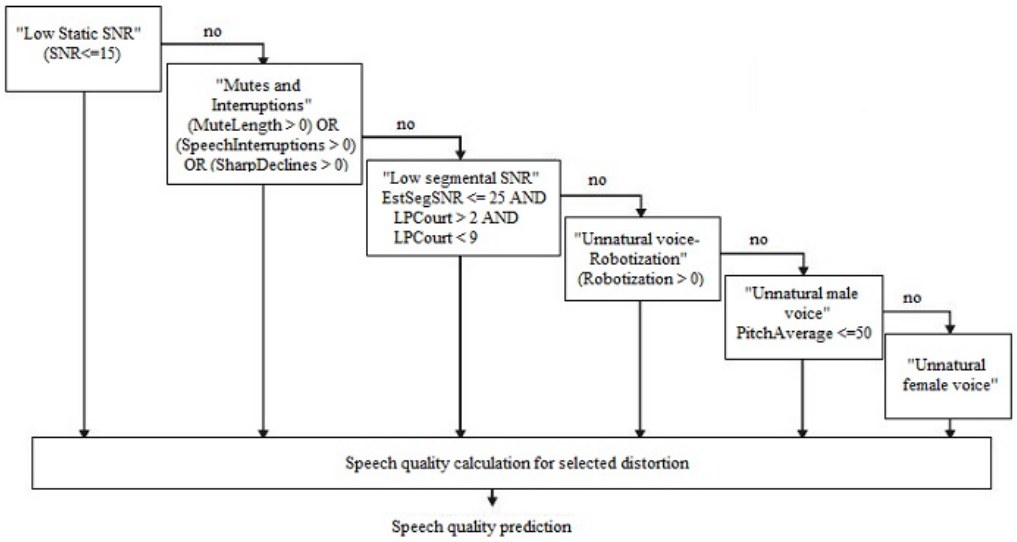


Figure 2.6: Distortion class decision from Figure (22) in [12].

5. common unnaturalness.

After assigning the input signal to a distortion class, an intermediate speech quality score is computed. To generate the intermediate speech quality, other parameters previously calculated are linearly weighted depending on their influence on speech quality in the selected distortion class [8]. In the following, we briefly review the threshold-based classification of the distortion as shown in Figure 2.6, and review the parameters used for evaluation of the speech quality in each class.

Speech quality is severely affected by the presence of high noise [8], and most of the signals with background noise have a low MOS, typically from one to three [8]. As shown in Figure 2.6, the distortion of a signal is classified as the "high-level background noise" when SNR is equal to or below 15 dB. The quality estimation of this class is computed by a linear combination of parameters that are based on the variation of the last vocal tract tube section, representing the opening of the mouth [8].

In the P.563 algorithm, the signals that are affected by interruptions and

contain mutes, are identified by sharp changes in the signal level [12]. As shown in Figure 2.6, these signals are classified as "signal interruption". The quality estimation of this class is computed by a linear combination of parameters that specify the length of the interruption events and the estimation of the noise during voice events [8].

The signal is classified as "signal correlated noise" when noise distortions vary with the signal envelope [12]. The detection of this class and the evaluation of the quality is based on the parameters that estimate the short-term signal to noise ratio during speech activity [8].

Voice signals that contain too much periodicity are classified as "speech robotization". Such signals are mostly the result of band-limitations such as those used in GSM networks [12]. The classification of robotization distortion and evaluation of the quality is based on the parameters that estimate the amount of frame repetition in the signal [8].

The signals that are not classified in one of the previous classes are considered to be in a general class of "common unnaturalness" [12]. This means that even when the signal is undistorted, it will be assessed using the same rule as "common unnaturalness" [12]. In this class, a primary distinction is made between the male and female voice based on a threshold applied to the pitch average [8]. Following that, the quality is estimated based on the parameters describing the vocal tract analysis, the basic voice quality calculated with the speech reconstruction system and background noise descriptors [8].

After computing the intermediate quality, the final step is to compute the overall quality of speech. Finally, in this functional block, the overall speech quality is calculated by linearly combining the intermediate quality result with some of the parameters that are computed in the previous blocks.

The speech quality assessment proposed in Chapter 4 employs the P.563 standard to compute the parameters explained above. It utilises these parameters to create an informative feature set, which will be an input to the

machine learning based quality predictor.

2.4 ANIQUE+

Auditory Non-Intrusive Quality Estimation plus (ANIQUE+) is an American National Standard (ANS) proposed by Kim in 2007 [44]. It is a perceptual model simulating the functional roles of the human auditory system, which adopts improved modelling of quality estimation by applying

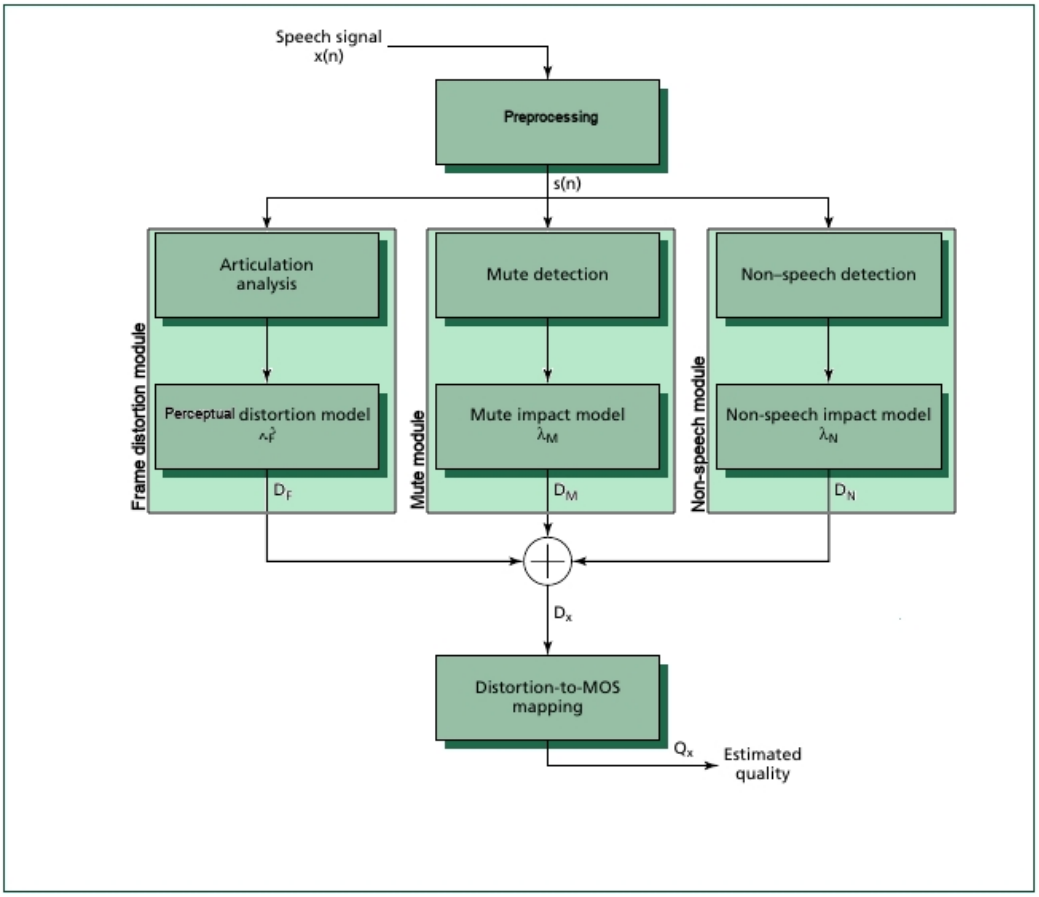


Figure 2.7: The overall block diagram of American national standard ANIQUE+, adapted from [44].

a statistical learning paradigm [44]. Similar to P.563, ANIQUE+ can be unpredictable when applied to signals with unknown distortions that are processed with different categories of algorithms [15]. Furthermore, similar to P.563, ANIQUE+ relies on signal properties and assumptions that are not always realised in real-world environments. Hence the assessment scores might not be consistent with a human perception rating [36]. However, an amplitude modulation domain processing of speech that is used in ANIQUE+ is a point of interest for our work as low-frequency modulations of speech are shown to be the fundamental carrier of linguistic information [63]. Consequently, the distortion parameters computed in ANIQUE+ form informative features and are used in the enhanced feature set proposed in Chapter 4.

Figure (2.7) shows the overall block diagram of the ANIQUE+ model. The speech signal is preprocessed in the first module. Next, the overall objective distortion is computed, which is composed of three different types of distortion: 1) the frame distortion,D_F , 2) the mute distortion, D_M, and 3) the nonspeech distortion D_N. Finally, the overall objective distortion value is linearly mapped onto an objective speech quality score. In the following, we explain these blocks in more detail.

2.4.1 Preprocessing

This module first uses the P.56 speech voltmeter [75] to normalise the level of the speech signal to -26 dBov. Next, in order to reflect the frequency characteristics of the handset used in listening tests [74], it applies the receive-side modified intermediate reference system receive filter. The preprocessed signal will be used as input to the three subsequent modules that extract distortion parameters [44].

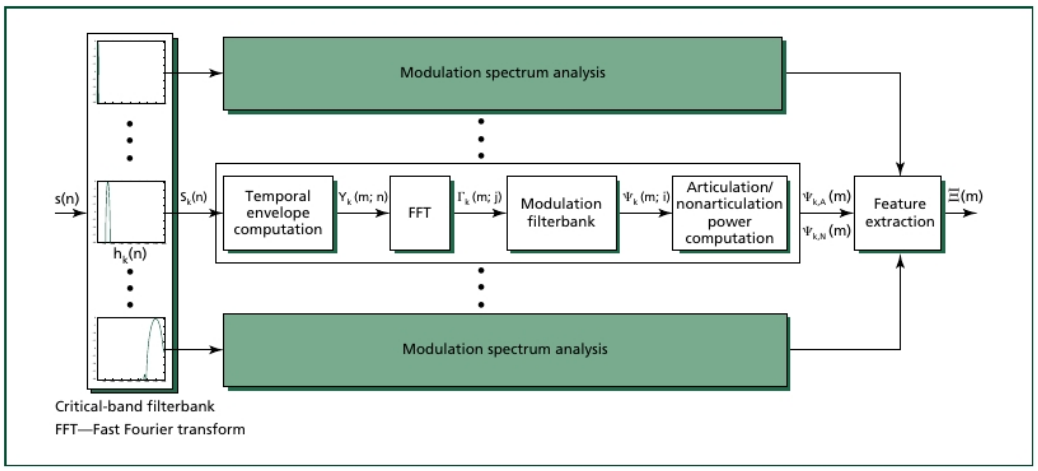


Figure 2.8: Block diagram of the articulation analysis module from Figure (2) in [44].

2.4.2 Frame distortion module

The frame distortion module is composed of two blocks: I) the articulation analysis, and II) the perceptual distortion model. The Articulation Analysis module decomposes the incoming speech signal into successive time frames and extracts the feature vector of individual frames based on modulation analysis. These feature vectors are used as the input to the overall frame distortion model [44], which computes an individual distortion value for each frame and aggregates them over the voice file to compute an overall frame distortion value.

I) The Articulation Analysis module shown in Figure 2.8 is motivated by the human auditory system at the peripheral and central levels [44], and perceptual feature vectors relevant to human speech quality perception [44]. The following briefly explains the three modules in Figure 2.8.

The first module is *critical-band filterbank*, which simulates the first stage of the human auditory system based on filtering the preprocessed signal by a bank of critical-band filters [44]. The second module is *Modulation*

spectrum analysis. It simulates the set of modulation detectors at the central level of the auditory system, each of which is tuned to a specific modulation frequency [44]. As shown in Figure (2.8), it contains four sub-blocks and its outputs are $\Psi_{k,A}(m)$ and $\Psi_{k,N}(m)$, which represent the average articulation and nonarticulation power of m^{th} frame respectively:

- *"Average articulation power"* reflects the amount of signal components relevant to natural human speech covering the modulation frequency range of 2 to 30 Hz, corresponding to the limited movement speed of the human articulation system. [44].

- *"Average nonarticulation power"* is the amount of perceptually annoying distortions produced at the rate beyond the speed of human articulation systems [44].

The articulation and nonarticulation power are the input to the third module called *Feature extraction*, which computes the feature vectors for frame distortion in ANIQUE+ model. The feature vector for the m^{th} frame is a 69-dimensional vector expressed as $\Xi(m)$, and it contains the normalised representation of articulation power, nonarticulation power, and critical band power.

the feature vector, $\Xi(m)$, for frame distortion in ANIQUE+ model contains the normalised representation of articulation power, nonarticulation power, and critical band power:

II) In the perceptual distortion [44], the distortion of individual speech frames is first estimated from the input feature vector, $\Xi(m)$, using a Multi-Layer Perceptron (MLP). The overall distortion for speech is then computed by aggregating over the distortions of individual frames.

The detailed mechanism of quality perception by human listeners is not known [44]. Hence ANIQUE+ employs a machine learning approach in which the objective model learns the relationship between feature vectors extracted for the speech frames and the quality rating associated to that speech.

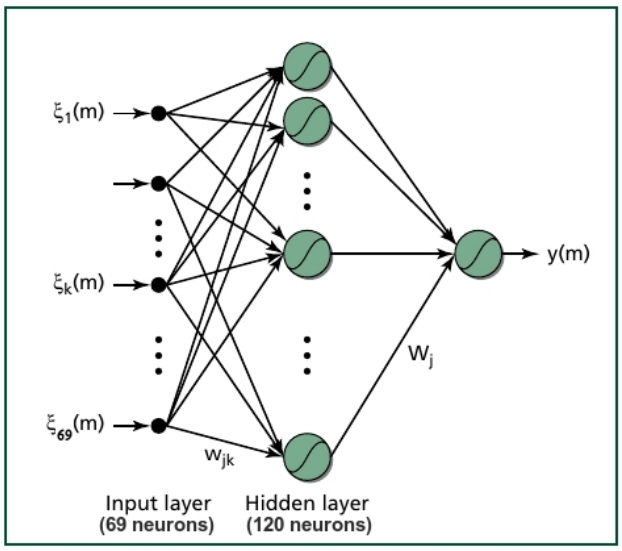


Figure 2.9: Multi-layer perceptron for perceptual distortion model, adopted from Figure (4) in [44].

The frame distortion model here is a Multi-Layer Perceptron (MLP) with one hidden layer [44]. As shown in Figure 2.9, the MLP has 69 input neurons, 120 hidden neurons, and one output neuron [44]. The estimation of parameters of the MLP is derived from the iterative steps of error back-propagation learning [79], using the training database that contains all the network distortions except mutes and nonspeech distortions [44].

After the distortions of individual speech frames are estimated using MLP, the overall distortion D_F is computed as:

$$D_F = D_S + D_B, \tag{2.1}$$

where D_S is the distortion in the speech obtained by accumulating frame distortions over the active speech frames [44], and D_B is the distortion estimated for audible background noise frames [44]. As shown in Figure (2.7), D_F computed in (2.1) is later accumulated with the output of Mute module and non-speech module to be mapped to a quality score.

2.4.3 Mute Module

Since the impact of mutes is believed to be severe, ANIQUE+ applies a separate mechanism to handle mute distortions [44]. The mute model first detects unnatural mutes in speech signals and then estimates their impact on the perceived quality [44].

I) Mute detection here is based on the assumption that natural human speech signals cannot change too abruptly [10]. Mute distortions are categorised into two groups based on when losing the speech frames starts to occur: 1) unnatural abrupt stops, and 2) unnatural abrupt starts. If losing the speech frames starts during the active speech intervals, the mute is categorised as an "unnatural abrupt stop" [44]. When the loss starts during the silence before a speech activity starts, it is categorised as an "unnatural abrupt start" [44]. In the case of an unnatural abrupt start, the beginning of mute cannot be recognised, and only the end of the mute is detected.

The mute detection module uses a speech activity profile to specify unnatural abrupt starts and stops. The speech activity profile is based on the frame log-power computed every 4 ms from an 8-ms-long segment of $s(n)$ and is also based on the time derivative of frame log-power with adaptive background noise power estimation [44]. Detection of unnatural abrupt stops and starts are performed at every downturn and upward transition in the speech activity profile, respectively [44]. Although the time derivative of frame log-power is useful to detect abrupt stops and starts, it is not enough to distinguish between the unnatural abrupts and the natural stops such as /p/ and /t/ [44]. Consequently, a feature vector is extracted to include context information at every downturn/upward transition of the speech activity profile and 15 ms before/after that [44]. The feature vector includes the 12-th order mel-frequency cepstral coefficients, frame log-power, time derivative of frame log-power, and voicing factor derived from the autocorrelation function of speech [44]. The mute detection models for unnatural abrupt stops and starts are two MLPs with one hidden layer using the feature vector as input. The two MLPs detectors are indi-

vidually trained on all the data in the training database, to detect unnatural abrupt stops and unnatural abrupt starts respectively [44].

II) The mute impact model estimates the impact of mute distortions on the perceived speech quality. Experiments revealed that although human subjects assess the quality of speech continuously over time, their opinion about the quality of speech is more affected by the recent event rather than the past ones [80, 81]. In the ANIQUE+ model, the impact of mutes is modelled based on this biological short-term memory of humans, where recent distortions play a more significant role than past ones [44]. Hence, for a speech signal that contains K mutes, the objective distortion at time t is modelled as [44]:

$$D_M(t) = \sum_{i=1}^{k} v_i \exp[-(t - t_i)/\tau] u(t - t_i). \tag{2.2}$$

Here t_i for $(i = 1, 2, ..., K)$ represents the time that each mute ends. u is a unit step function and $u(t - t_i)$ is equal to one for $t \geq t_i$, and equals to zero when $t < t_i$ [44]. In this model, the effect of each mute is raised by the amount of v_i at the end of the mute event, and it decays over time with the time constant τ [44].

The value of v_i, described as the instantaneous distortion of the i-th mute is estimated by

$$v_i = p_1 \log_2(L_i) + p_2. \tag{2.3}$$

Here L_i represents the length of the i-th mute, and p_1 and p_2 are constants [44]. The optimal parameters p_1 and p_2 are found by training the mute model, λ_M, on a dataset that contains at least one mute distortion but does not include non-speech distortion [44]. The mute model, λ_M, is trained after training the perceptual model, λ_F. This means the parameters of previously trained λ_F are considered as constant when computing the parameters of λ_M.

Human subjects rate the quality at the end of speech signal. Hence the

mute distortion for a speech signal with length T is estimated as:

$$D_M = D_M(T) = \sum_{i=1}^{k} v_i \exp[-(T - t_i)/\tau] u(T - t_i). \tag{2.4}$$

As shown in Figure 2.7, D_M computed in (2.4) is later accumulated with the output of the perceptual module and non-speech module to be mapped to a quality score.

2.4.4 Non-speech module

The non-speech module shown in Figure (2.7) detects and estimates the impact of non-speech activities that are annoying and occur, for example, when bit information in packets or frames is distorted during transmission [44]. In this case, if the distorted packets are not detected at the speech decoder side, the corrupted bits are used, which results in disturbing non-speech signals [44].

I) The non-speech detection here has a simple implementation, and it only recognises the non-speech activities that have significantly abrupt changes in frame power [44]. In this simple implementation [44], the positive and negative peaks of time derivative of frame log-power $(P(t))$ are first identified for each speech activity period. They will be then marked if their absolute value exceeds a threshold that is obtained empirically. The accumulation of the reciprocal of the time interval between two adjacent marked peaks is then used as a criterion for detecting non-speech activities [44].

II) The non-speech impact model illustrates the impact of non-speech distortions on the perceived speech quality. The impact of non-speech distortion is estimated to be proportional to the accumulated frame log-power (P_{acc}) in the non-speech activity region [44]. Hence non-speech distortion in ANIQUE+ is modelled as $D_N = q_1 P_{acc} + q_2$, where q_1 and q_2 are constants [44]. The optimal parameters q_1 and q_2 are found by training the non-speech model, λ_N, on a dataset that contains speech samples with at

least one non-speech distortion [44]. The non-speech model, λ_N, is trained after training the mute model, λ_M. This means the parameters of previously trained λ_F and λ_M are considered as constant when computing the parameters of λ_N.

2.5 Summary

In this chapter, we reviewed different types of speech quality assessment and justified why the focus of this thesis is on non-intrusive methods to predict listening quality. We reported related work in this context and summarised the advantage of machine learning based quality assessment systems over conventional standards. We described that machine learning based systems are more desirable as they are not restricted to particular services and are adaptable to various applications. Moreover, machine learning based systems allow the learning of new types of distortion, and unlike conventional methods, distortions in the test speech files are not required to be known in advance.

Furthermore, we explained that quality assessment methods that are based on deep learning are more desirable than the methods that are based on a shallow architecture. Deep learning algorithms are more powerful than conventional machine learning algorithms. In deep learning based methods, feature extraction is part of the overall system, and features are potentially more informative about the quality. In the next chapter, we review contemporary machine learning algorithms that we employed in the two quality assessment systems proposed in this thesis and explain how applying them results in new non-intrusive systems that has advantages over existing methods.

In this chapter, we additionally presented a description of two current standards in QA, namely P.563 and ANIQUE+. In both P.563 and ANIQUE+, the features are handcrafted by experts. P.563 focuses on the auditory system where ANIQUE+, additionally uses features based on vo-

cal tract analysis and speech production. In P.563 the decision algorithm is well designed based on the knowledge of experts. In contrast, ANIQUE+ less relies on the knowledge of experts and learns the relationship between the distortion parameters and the quality by employing MLP and learning from databases. Since ANIQUE+ is trained with extensive training data, it predicts the quality well for many applications. To conclude, both P.563 and ANIQUE+ have demonstrated acceptable accuracy for many telecommunications scenarios. However, their quality prediction performance is compromised where the applications involve noise suppression, dereverberation and wireless-VoIP tandem connections [40]. In this chapter, we described the modules from P.563 and ANIQUE+, which compute distortion parameters that are utilised in the feature set proposed in Chapter 4.

3

Introduction to machine learning

The use of machine learning has been widespread over the past decades. As explained in Chapter 2, employing machine learning algorithms in quality assessment is beneficial in multiple ways. The objective of this thesis is to develop a new non-intrusive quality assessment system based on contemporary machine learning methods. This chapter presents an introduction to machine learning and provides an overview of recent algorithms and the methods implemented in this thesis for quality assessment.

3.1 Introduction

Research on machine learning has a long history. In 1950, Alan Turing argued that programming a computer to have adult-level intelligence would be too difficult. Hence, he suggested instead of trying to produce a programme to simulate the adult mind, we should try to produce one which simulates that of a child. If the child-level mind were then subjected to an appropriate course of education, we would obtain the adult brain [82].

Machine learning has become a very active area of research since then, and new algorithms and application areas are discovered every day. Machine learning is currently applied in various domains such as computer vision, language processing, audio classification, pattern recognition, search engines, data mining, medical diagnosis, information retrieval, and game playing.

In Section 3.2 we provide a brief introduction to machine learning. Then in Section 3.3 we review standard machine learning methods with shallow architectures and provide an explanation of relevance vector machine (RVM) and the Correlated Nystrom Views (XNV), which we implement for supervised QA in Chapter 4. In Section 3.4 we explain deep learning methods, which is centred on the learning of useful representations of data. We review deep generative models and provide a detailed explanation of Generative Adversarial Networks (GANs), which is the core of our proposed unsupervised QA in Chapter 5. In Section 3.5 we review the popular divergence and distance measures that are used for training neural networks. The unsupervised QA in Chapter 5 utilises divergence metrics to quantify the quality of speech.

3.2 Basics in machine learning

In [83], machine learning is defined as the field of study that gives computers the ability to learn without being explicitly programmed . Machine learning is often necessary when human expertise does not exist, and a machine (computer) learns a task by sample data or past experience [84]. It also is beneficial when humans can do the task without difficulty, but are unable to explain their expertise with simple logic, and hence traditional programming is not adequate. Furthermore, machine learning is valuable when solutions change in time or need to be adapted to particular cases. In the following we explain the basics of machine learning.

What is machine learning? We use algorithms to solve problems on the computer. An algorithm is a finite list of well-defined instructions for transforming the input to output. For example, in a sorting algorithm, the input is a set of numbers, and the output is their ordered list. Different people might write different algorithms for the same task. Hence there might be different algorithms for one problem. However, there are some other tasks that we do not have any algorithm for. For example, in predicting the topic of a document, the input is a set of words, and we know the output should be one word that specifies the topic. However, we do not know how to map the input to output. In such a case we cannot directly write a computer program to solve that problem, and we would like the machine (computer) to automatically extract the algorithm, using example data or past experience. In other words, we want to use data to make up for lack of knowledge [84].

Machine learning algorithms may not be able to find the approximation with high predictive accuracy but may have the ability to construct a useful approximation, which can detect particular patterns or regularities. In certain applications, the efficiency (i.e., the space and time complexity) of the learning or inference algorithm may be as important as its predictive accuracy [84].

Types of learning Machine learning algorithms can be organized into two major settings based on the type of data available during training the machine: supervised learning and unsupervised learning. We will also explain a learning type that is intermediate between these two. This is referred to as semi-supervised learning.

In *supervised* learning, we know the value of the output for the input data in the training set. In this case, our dataset is called labelled data [85]. Curve-fitting is a simple example of supervised learning. In Chapter 4, we propose a supervised quality assessment, which uses training data that contains speech utterances labelled with their quality score.

In machine learning applications, one can often find a large amount of unlabeled data without difficulty, while labelled data are costly to obtain. Therefore a natural question is whether we can use unlabeled data to build a more accurate predictor, given the same amount of labelled data. This problem is often referred to as *semi-supervised learning* [86], which has been an area of interest in the machine learning community. One prominent approach to semi-supervised learning is based on the idea of using unlabeled data to learn some meaningful representation and map the input features to an intermediate feature space, called latent space [87]. In this procedure, instead of mapping the input features to the output, the algorithm learns to map latent space to the output. The hope is that the learner finds it is easier to learn from the latent space. In the context of semi-supervised learning, learning the feature representation from unlabelled data is called pre-training (or regularisation), and the following step that uses labelled data is called fine-tuning. The parameters learnt during pre-training might change in the fine-tuning step that is supervised [88].

In *unsupervised learning*, we have an unlabeled training data set, which is simply a set of input data (that are not mapped to any output value). The unsupervised learning can be applied for partitioning the training set into appropriate subsets. Such methods have an extensive application where it is desirable to classify data into meaningful categories [89]. Generative models [20, 90], which have obtained considerable attention recently, have become another promising area in unsupervised learning. Generative models use unlabelled training sets to learn the distribution of data and model how they were generated. In Chapter 5, we explain the novel supervised learning we implemented based on generative models. Our proposed method is more similar to the assessment functionality of the human brain, which is trained with unlabeled signals.

Types of output The goal of supervised machine learning is to learn a mapping from the input space to the output space. Based on the desired

outcome, the application of machine learning can be divided into two major categories: classification and regression.

The value of the output might be real value numbers or discrete value numbers. If the predicted variable is a real number, it is called a regression problem. In regression problems, the process that performs mapping is called a function estimator or regressor. Alternatively, if the predicted variable is discrete, it is called a classification problem, in which the output can represent a categorical value. In classification problems, the process that expresses the mapping is called a classifier, a recogniser, or a categoriser. The output itself is called a label, a class, a category, or a decision. The output may also have a vector-value with elements being real numbers or categorical values [89].

In the quality estimation literature, which is the focus of this thesis, the type of output depends on the protocol used for rating the quality. In many protocols, the human subjects provide a discrete score, and for that case, the problem is considered as a classification problem. For example, the ACR protocol [2], allows for five discrete scores, and hence the problem becomes a multi-class classification problem. However, most quality assessments estimate the subjective mean opinion score (MOS) [47]. In this case, the problem converts to a regression problem. It is noted that in some other protocols, such as the MUSHRA protocol [3], the user essentially provides a continuous score and hence the problem is a regression problem by definition.

Neural networks Neural networks are an extensive area of research in machine learning, which is inspired by the biological brain. A neural network is a connected graph with input, output and hidden neurons that are connected with weighted edges. The weights affect how much the input propagates forward through the network to the output. The neurons typically use a scalar-to-scalar function called an *activation function* to transform their input to a value called the neuron's activation. The weights are

updated during the learning process that is called training.

Training Training is a learning process in which sample data is used to estimate the parameters of the model. In general, learning comprises an iterative forward and backward propagations that adjust the weights in order to optimise the objective function with regards to the training data.

Objective function The objective function is the function to be maximised or minimised in an optimisation problem. In the machine learning context, the objective provides a formal specification of the problem. The objective function (sometimes called cost function or loss function in the machine learning context) is determined before the training begins. Mean Squared Error (MSE) is an example of a simple objective function that is commonly used for regression problems. It is defined as:

$$L = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2, \tag{3.1}$$

where y_i and $\hat{y}_i$ are actual and predicted data points respectively, and N is the number of data points available for training. MSE is useful for regression problems, in which the intention is to reduce the cost based on the difference between the actual data points and the predicted regression line. Sections 3.3 and 3.4, reviews machine learning methods in the literature with regards to the varieties in architectures and objective functions.

Gradient descent Gradient descent optimisation methods are the most common techniques used for training neural networks and estimating their parameters [91]. Gradient descent is a way to find the model's parameters, $\theta \in R^d$, that minimise the objective function $L(\theta)$. Gradient descent methods are based on iterative updates of the parameters in the opposite direction of $\nabla_\theta L(\theta)$. Here $\nabla_\theta L(\theta)$, which is the gradient of the objective function with regards to the model's parameters, determines the direction of the slope of the surface created by the objective function. In short,

gradient descent algorithms follow the direction of the slope of the objective function downhill until it reaches its (local) minimum value [91]. The learning rate, η, determines the size of the steps to take towards the valley.

Basis functions and kernel functions Linear regression or classification is simple to implement but is not effective if a linear function cannot approximate the relationship between input and output. A key concept to overcome this problem in kernel-based methods is to use a set of basis functions for mapping data to space with a much higher dimensionality where a linear regression or classification is appropriate by applying hyperplanes.

As noted in Aizerman [92], inner products in this high-dimensionality space can be expressed with kernel functions in the original space. Therefore, using kernel functions enables us to carry out computations implicitly in the high dimensional space, without actually performing a transformation to the high dimensionality space. This concept is referred to as the *kernel trick*, which leads to computational savings.

Shallow and deep architecture Machine learning algorithms can be categorised into two types according to the level of their hierarchical abstraction from data [93]: shallow learning and deep learning. In this context, shallow and deep architecture refer to two different ways to model the problem when the complexity of problems increases. In conventional (i.e., shallow) machine learning methods, there is only one hidden layer, and the capacity of neural networks is controlled by varying their width (i.e., the number of neurons in the hidden layer). On the other hand, deep neural networks allow multiple hidden layers and their significant power is achieved through the depth of network. It is shown [94] that functions that can be implemented by a deep network of polynomial size, require exponential size in order to be approximated by a shallow network. Deep learning methods have gained popularity because they often outperform

shallow machine learning methods. However, they require an extensive amount of data for training. Due to the unavailability of a large scale labelled database, the supervised method proposed in Chapter 4 has a shallow architecture. On the other hand, the unsupervised method proposed in Chapter 5 is based on deep learning. The proposed method does not require labelled data, and for training the deep network, it utilises large scale unlabelled data that is publicly available. The next two sections provide a literature review on machine learning methods having shallow and deep architectures.

3.3 Machine learning with shallow architecture

The supervised method proposed in Chapter 4 has a shallow architecture. In this section, we briefly explain this model and review the Relevance Vector Machine (RVM) and the Correlated Nystrom Views (XNV), which are employed in Chapter 4.

In conventional machine learning problems, we have a model with a relatively small number of parameters and aim to optimise them so that the prediction error (or in general, the cost function) has its smallest value. In order to solve a machine learning problem, we execute code on the computer to use the training data and find the optimised parameters of that model.

For example, in a regression problem defined as:

$$y = f(x) + \epsilon, \tag{3.2}$$

the output is assumed to be the sum of a deterministic function of the input and random noise, where $f(x)$ is the unknown function. Machine learning algorithms aim to approximate the parameters of $f(x)$ based on sample data. Since data comes from a process that is not completely known to us, the outcome is defined as a random variable Y. Here the outcome Y is drawn from a probability distribution $P(Y = y)$, which specifies the

process. It should be noted that the process might be deterministic in reality, but because the complete knowledge about that is not accessible, it is modelled as random and it is analysed by applying probability theory [84].

The core task of machine learning in the regression problem above is modelling this problem by inference from the samples in training data. Programming computers to make an inference from sample data is a combination of statistics and computer science. Statistics provides the mathematical framework for inference, and computer science provides efficient implementation of the inference methods.

One approach to estimate the model parameters in machine learning problems is Maximum Likelihood Estimation (MLE). Maximum likelihood estimation attempts to find parameters θ for the model so that the observations $D = (d_1, ..., d_n)$ in training data are most likely to occur. In MLE, the objective function to be maximised is called the likelihood function. The likelihood function is defined as:

$$L(\theta|D) = f(D|\theta), \tag{3.3}$$

where $f(D|\theta)$ is the probability of observing samples D from a model that has properties defined by θ. Consequently, the maximum likelihood estimator of θ is defined as:

$$\hat{\theta} = \arg\max_{\theta}[L(\theta|D)], \tag{3.4}$$

which is typically computed through gradient descent. In MLE, the optimised parameters, $\hat{\theta}$, are found so that the likelihood (or equivalent log likelihood) of the training data is maximised. Optimisation with MLE is prone to overfit[1] to the training data when the training set is small. Furthermore, in the maximum likelihood approach, a parameter is treated as an unknown constant and hence no *statistical* information is provided about the results.

[1]Overfitting (or overlearning) means that the system learns aspects of the database that are not representative and that do not generalise to other data.

Alternatively, *Bayesian* estimation treats a parameter as a random variable, which takes any prior information into account by using a prior probability distribution. Bayesian estimation is used when some prior information about the parameter is available. It is called the prior because it is the knowledge we have before looking at the samples. Prior beliefs are particularly important when the number of available samples is small. Bayesian estimation combines the information that can be learnt from data with the prior information, and therefore it is less prone to overlearning the samples.

Bayesian estimation use Bayes' rule to combine the prior and the value calculated from the training set. It calculates the posterior probability after having seen the observation as:

$$posterior = \frac{prior \times likelihood}{evidence}. \tag{3.5}$$

This can be written as:

$$p(\theta|D) = \frac{p(\theta) \times p(D|\theta)}{p(D)}, \tag{3.6}$$

where θ denotes the unknown parameters, and D is a training dataset. $P(D)$ is a normaliser to guarantee that the posterior integrates to one.

Maximum A Posteriori (MAP) estimate is a Bayesian approach that is often applied when a probabilistic model suffers from overfitting. In MAP estimation, the posterior density is reduced to a single point [95] and the prediction is computed as:

$$p(y_*|x_*, D) = p(y_*|x_*, \theta_{MAP}), \tag{3.7}$$

where x_* is the test datapoint, y_* is its predicted output and θ_{MAP} are parameters that maximised the posterior. On the other hand, in the full Bayesian approach we use the complete posterior by averaging the output of all possible predictions, using all values for parameters, weighted by their probabilities:

$$p(y_*|x_*, D) = \int p(y_*|x_*, \theta)p(\theta|D)d\theta. \tag{3.8}$$

Except for certain simple cases where the posterior has a nice form, the integral is analytically intractable and is not easily evaluated unless approximation methods such as sampling are applied [95]. Fortunately, this operation is relatively simple if a Gaussian prior is considered over the joint probability distribution of the observations. Consequently, *Gaussian Processes for Machine Learning* [96] have become one of the important Bayesian machine learning approaches.

The fundamental idea in Gaussian processes is the closer data points are, the more correlated to each other their corresponding output will be. This idea is modelled below:

Model 1. *Let x_1 and x_2 be two input vectors where d represents the distance between them, and y_1 and y_2 are their corresponding outputs. We assume y_1 and y_2 have a Gaussian distribution with covariance matrix* $\begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix}$*:*

- *If d is small then y_1 and y_2 correlate strongly and hence c_{12} has a large value.*
- *If d is large then y_1 and y_2 become independent and hence c_{12} tends to zero.*

Gaussian processes formed a useful background for many well-known machine learning methods with shallow architecture. The Relevance Vector Machine (RVM) introduced by Tipping [97] is a special case of a Gaussian process. In general, the performance of the RVM is good for both regression and the binary classification. However, the original RVM suffers from a number of drawbacks. For example, it does not perform well for multi-class classification. Moreover, the order of its training effort is high. Major efforts have been made to reduce these drawbacks. This has led to the accelerated training algorithm described in [98], which does not use all data for a start. Furthermore, a multi-class training procedure with good performance was presented in [99], which uses the fast procedure described in [98].

The success of the RVM was because of both its effectiveness and its sparseness. The model is called sparse when it contains relatively few non-zero parameters. Sparsity makes the model interpretation simpler and generally generate models with improved productivity [100]. The sparseness in RVM was achieved by specifying a prior distribution for the vector of weights. The sparseness in RVM relies on hyperparameters that govern the prior distribution of the weights. The original RVM of Tipping [97] provides point estimates of the hyperparameters, which is known as type II maximum likelihood.

The RVM method defined in [101] applies a true Bayesian approach as an alternative to point estimates for the hyperparameters. This fully Bayesian RVM is based on the variational approximation and does not seem to have any practical advantages over the original RVM, but is more elegant. While closely related to the RVM, this alternative sparse method is sometimes referred to as a Sparse Bayesian Learning (SBL).

Fast methods for SBL have been developed in [102] for regression problems, which appears to make the variational-approximation based methods more attractive than the original RVM approach. In Chapter 4, we consider using these fast methods as the back-end regressor of the quality assessment system. By using this statistical method, the objective of the quality assessment turns into estimating the *predictive distribution of the quality* of an utterance. In this case, results can be used to obtain both the prediction and the precision. For example, when the output has a Gaussian distribution, the predicted value is its mean, and its precision is the inverse of its variance.

As will be explained in Chapter 4, we also employed Correlated Nystrom Views (XNV) [103] for quality assessment. XNV is a fast semi-supervised algorithm with shallow architecture for both regression and classification. XNV is shown to outperform other methods by improving predictive performance and reducing the variability of performance whilst also reducing runtime by orders of magnitude [103]. The method builds on two

main ideas. The first idea is based on the Nyström method that generates random features [104] for constructing views on a given data set. The second is multiview regression, using Canonical Correlation Analysis (CCA) [105], which biases the regression towards useful features.

In brief, XNV first applies two equally useful but sufficiently different views on the data. Then it uses the canonical norm to penalise features that are uncorrelated across the views. XNV substantially reduces the variance with a minimal increase in bias [103]. This makes it a good option for the quality assessment systems in Chapter 4 that are based on the statistical analysis methods.

The supervised and semi-supervised methods reviewed in this section have a shallow architecture, which is the base for the quality assessment system proposed in Chapter 4. The use of machine learning methods with deep architecture has appeared as a promising area of research in statistical machine learning. In the next section, we review deep machine learning, which is the introductory to the contents of Chapter 5.

3.4 Deep neural networks

In the previous section, we briefly reviewed machine learning methods with shallow architectures. The focus of this section is machine learning methods with deep architectures.

There have been numerous studies demonstrating the effectiveness of deep learning methods in a variety of application domains [106, 107]. Deep networks have been mainly applied to visual classification databases such as handwritten digits [108], image classification [21], object detection [109], face detection [110, 111] or pedestrian detection [19], and also to acoustic signals to perform speech recognition [112], acoustic modeling [113] and audio classification [114]. In Section 3.4.1 we present an introduction to deep neural networks in general and review the relevant methods.

Deep neural networks were traditionally used for *discriminative* mod-

els, which focus on predicting labels of the data. Deep *generative* models, which focus on modeling the actual distribution of data, have also achieved considerable success. The unsupervised quality assessment method proposed in this thesis is based on deep generative models. In Section 3.4.2 we present an introduction to deep generative neural networks. In Section 3.4.3 we explain Generative Adversarial Networks (GANs), which we implemented for developing the quality assessment proposed in Chapter 5.

3.4.1 Introduction

Deep learning has appeared as a promising area of research in statistical machine learning [110, 115, 116, 117, 118, 119, 120, 121]. Learning algorithms for deep architectures are organised in a hierarchy with multiple levels. This concept takes its inspiration from the mammalian visual cortex, which consists of a chain of processing elements, each of which is associated with a different representation of the raw visual input [88]. Accordingly, deep learning is centred on the learning of representations of data which are useful for the task at hand.

In deep learning, it has been hypothesised that learning a hierarchy of features makes it easier and more practical to develop useful representations. Such representations are beneficial because although they are tailored to a specific task, they borrow statistical strength from data that originates from other related tasks. Furthermore, learning the feature representation usually leads to higher-level features that are more robust to unforeseen sources of variance that exist in real data [88]. However, until 2006, it was not clear how to train such deep networks. This was because gradient-based optimisation often appears to get stuck in poor solutions when the algorithm starts from a random initialisation of the parameters [119].

In 2006 Hinton et al. [115] proposed a greedy layer-wise unsupervised learning procedure named *Deep Belief Networks* (DBN). DBNs are com-

posed of several layers of Restricted Boltzmann Machines (RBM). Boltzmann machines are statistical models that are characterised by the joint probability distributions of the states of their nodes, a property shared with graphical models [122]. The learning aims to make the weights such that in "free-running" mode the network gives for the visible nodes the same distribution as the environment does. The hidden units are trained to capture higher-order data correlations that are observed at the visible units [106]. The principle is that each layer starting from the bottom is trained to represent its input (the output of the previous layer). After this unsupervised initialisation, the stack of layers can be converted into a deep supervised feedforward neural network and fine-tuned by stochastic gradient descent [88].

Shortly after Hinton introduced DBNs, alternative algorithms were proposed based on auto-encoders. Examples are ordinary auto-encoders [116], sparse auto-encoders [117], denoising auto-encoders [123], variational autoencoders [90], and adversarial autoencoders [124]. The auto-encoder neural network is an unsupervised learning algorithm that sets the target values to be equal to the inputs and applies backpropagation to learn the weights. These deep auto-encoder algorithms can be seen as learning to transform one representation (the output of the previous stage) into another, at each step disentangling the factors of variations underlying the data [125].

The methods above either 1) perform a greedy-layer-wise pre-training of weights, using unlabeled data alone followed by supervised fine-tuning (as introduced by Hinton using RBMs), or 2) learn unsupervised encodings at multiple levels of the architecture jointly with a supervised signal [126]. The aim is that the unsupervised method improves the accuracy of the task at hand. It has been observed that once a good representation has been found at each level, it can be used to initialise and successfully train a deep neural network by supervised gradient-based optimisation [125].

Comparative experimental results have shown that deep networks can

outperform shallow architectures. However, when a deep neural network is trained on a small training set, it typically tends to suffer from overfitting. Accordingly, due to the limitation of the available data, deep architecture does not appear to be advantageous for the supervised quality assessment in Chapter 4.

Dropout is an algorithm introduced by Hinton et al. [127] to prevent neural networks from overfitting. Dropout can be interpreted as a form of regularisation by adding noise to the fully connected neural network layers. Each element in these layers is kept with probability p, otherwise is set to 0 with probability $(1-p)$. Dropout improves the network's generalisation ability, bringing improved performance on test datasets.

Drop-connect generalises Dropout [128] by randomly dropping the weights rather than the activations. Each hidden unit in such neural networks must learn to work with a randomly chosen sample of other units [128]. This should make each hidden unit more robust and drive it towards creating useful features on its own without relying on other hidden units to correct its mistakes [128]. In Chapter 4, we implement a supervised quality assessment with a deep architecture regularised by drop-connect. However, the system did not perform better than the one with shallow architecture. This can be justified based on the size of data available in comparison with the size of the neural network.

Unlike the supervised method in Chapter 4, the unsupervised method proposed in Chapter 5 is not dependant on labelled data and the availability of data is not a limitation any more. Hence, in Chapter 5 we benefit from deep learning and train the system with speech signal files that are freely available.

One other significant difference between the methods proposed in Chapters 4 and 5 is the size of the input. In Chapter 4, due to limitation of available data, we handpicked the useful features for the system and utilised a front-end module to extract an enhanced feature set from raw signals. However, in Chapter 5, spectrograms of the raw speech signals are used

for input because deep architectures allow large input vectors.

Standard neural networks receive a single vector as input and training is not efficient on multi-dimensional inputs. A convolutional neural network (CNN) [129] is a special kind of a neural network designed to cope with the variability of 2D shapes. They ensure some level of shift, scale and deformation invariance by combining local feature fields, shared weights, and utilising spatial subsampling [111]. Deep convolution neural networks (DCNNs) [21, 130, 131, 132, 133, 134] have shown excellent performance in processing images and spectrograms. In Chapter 5 spectrograms of the speech signals are used as the input into our quality assessment system and convolutional networks are beneficial.

The architecture of the regular deep neural networks and CNNs inherently relies on the assumption that samples are generated independently. Recurrent neural networks (RNNs) are feed-forward networks with a specific structure, based on the notion of time layering, which enables them to model data with temporal or sequential structure and different length inputs and outputs. RNNs became a powerful learning tool [135] for sequential inputs like speech and language processing where data points are related in time. In Chapter 6, we discuss using RNNs as the future line of this work, which enables our system to use raw signals as input instead of spectrograms.

In this section, we briefly reviewed deep neural networks and how it is advantageous for unsupervised quality assessment. In the next section, we review deep generative neural networks, which are applied in the unsupervised quality assessment method proposed in Chapter 5.

3.4.2 Deep generative networks

Generative models are powerful tools for learning data distributions in unsupervised learning. As will be explained in Chapter 5, the unsupervised method proposed in this thesis employs a deep generative model. In this section, we review deep generative methods and justify why we

selected Generative Adversarial Networks (GANs) to implement for the quality assessment proposed in Chapter 5.

The original application of generative models is sampling from a learned distribution and synthesising new instances. However, the applications of generative models have extended and they are considered an excellent tool for representation learning. Representation learning is learning a meaningful and interpretable latent representation typically called the latent space. It has a vast area of applications, not limited to, but including visualisation by projecting data onto two or three dimensions, data compression, and detection of abnormal patterns.

To generate data, generative models typically sample from a simple distribution and map that into a data point in the learned distribution. Hence good generative models are hopefully able to learn a good representation [136] automatically, where that simple distribution (that is easy to sample from) is considered to be the latent space.

Early methods of representation learning are based on restricted Boltzmann machines [115] and deep autoencoders [116]. Deep Convolutional Generative Adversarial Networks (DCGANs) [22] that are based on GANs, are shown to be a successful tool for representation learning. DCGANs benefit from convolutional networks with adding certain constraints on the architecture of the networks and the connections between the neurons. The variational autoencoder [90] and the adversarial autoencoder [124] are also shown to learn representations well by imposing a prior distribution to the latent space. Variational Recurrent [137] and Adversarial Symmetric Variational Auto-Encoders (AS-VAE)[138] are other examples of generative models with a similar concept.

It has been proposed [139] that a good representation is one that disentangles the underlying factors of variation. InfoGAN [136] successfully learns this in an unsupervised way by introducing the information-theoretic regularisation term that forces high mutual information between latent codes and generator distribution. In [140] the authors derived vari-

ational lower and upper bounds on the mutual information between the input and the latent variable. They used these bounds and derived a rate-distortion curve that characterises the tradeoff between the compression and the reconstruction accuracy of the samples in the training set.

In Chapter 5, we go out of the domain of the rate-distortion curve and study the case where samples come from a distribution that is different from the training set. We demonstrate that the latent variables become correlated when the distribution of input is different from the training data. We benefit from this novel idea and hypothesise that the correlation between the latent variables will be a good measure of quality when we compare the samples under test with the training data.

WaveNets [141], Generative Adversarial networks(GANs) [20] and variational Auto-encoders (VAE) [90] are the state-of-the-art generative models. The latent space does not have an explicit definition in WaveNets. Considering the criteria defined in Chapter 5 is for quality to be measured in the latent space, we do not pursue WaveNets. In VAE, the latent space is regularised using the KL-divergence (which is explained in Section 3.5) between encoded samples and the prior. This means that the distribution of the encoded samples is expected to be an approximation of the prior distribution. Hence the distribution of the prior is naturally slightly different from that produced by the trained encoding network. The approach we propose is based on the assumption that the the distribution of the prior is same as that produced by the trained encoding network. Therefore VAE is not a suitable candidate for our purpose either. In contrast, GANs use a simple continuous input noise vector z, which reflects the latent space and imposes no restrictions on how the generator may use this noise. Therefore the distribution of the latent space in GANs is same as the distribution defined as the prior. Furthermore GANs are more successful in modelling signals in comparison with VAE. Hence, GANs suit our purpose better and we prefer GANs as the basis of our system for our work. In the next section we review GANs and the approaches proposed

to improve its training.

3.4.3 Generative adversarial networks

Generative Adversarial Networks (GANs) [20] are learning techniques for both semisupervised and unsupervised problems, which have gained much attention since 2014 [22, 142, 143, 144, 145, 146]. As explained in the previous section, the unsupervised quality assessment proposed in Chapter 5 is based on GANs. The explosion of interest in GANs is driven not only by "their potential to learn deep, highly nonlinear mappings from a latent space into a data space and back", but also by "their potential for deep representation learning that can be used in a variety of applications" [147] (including image synthesis, semantic image editing, style transfer, image super resolution, and classification [147]). Image processing was the primary domain of GANs, but the idea was soon employed in other areas such as video synthesis [148, 149] and language processing [146, 150, 151, 152]. In this section, we first describe GANs. Then we review other GAN-based methods that improve the training and explain why we selected the original GAN to implement for the unsupervised quality assessment in Chapter 5.

GANs use a two player min-max game and learn a generator network G that generates samples playing against a discriminator network D [20]. G learns to generate samples by transforming a random input z drawn from a simple probability distribution $P_{\text{noise}}(z)$ into a sample $G(z)$ from distribution of data P_{data}. D learns to differentiate between samples from P_{data} and the generated distribution P_G [20]. In other words, D and G play a two-player minimax game as:

$$\min_G \max_D V(D,G) = \mathbb{E}_{x\sim P_{\text{data}}(x)}[\log D(x)] + \mathbb{E}_{z\sim P_{\text{noise}}(z)}[\log(1 - D(G(z)))], \tag{3.9}$$

where $V(D,G)$ is the value function of the game and $D(x)$ represents the probability that x came from the P_{data} rather than P_G [20]. If G and D have

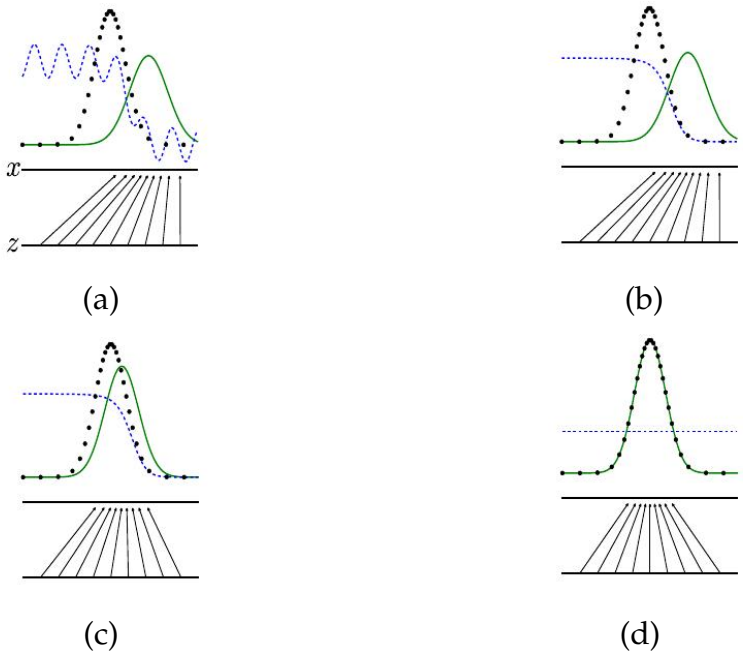


Figure 3.1: Guidance to understand the min-max problem in GANs (taken from [20]): In order to learn the distribution P_{data} (black dotted line), Generative Adversarial Nets iteratively update generative distribution P_g (green solid line) and discriminative distribution D (blue dashed line). The lower and higher horizontal lines are the domain from which z is sampled and the domain of x respectively (The arrows show $x = G(z)$ maps the uniform distrbution z to P_g). (a) P_g is similar to P_{data} and D is a partially accurate classifier [20]. (b) D is trained to discriminate generated samples from data [20]. (c) $G(z)$ moved to those regions that are more likely to be classified as real data [20]. (d) After several steps of training $P_g = P_{\text{data}}$ and $D(X) = \frac{1}{2}$ cannot differentiate between the two distributions [20].

enough capacity, they reach the point that P_G matches P_{data}. Figure (3.1) from [20] is a pedagogical explanation of the approach.

The original GAN models may suffer from different problems such as mode collapse, diminished gradient, and non-convergence. A number of works have been proposed to improve training GANs and they can be cat-

egorised into modifying the network design, updating the cost function, or using a different optimization technique.

Deep Convolutonal GAN (DCGAN) [22] is one of the most popular network design for GANs. GANs are known to be unstable to train as they often result in generators that produce nonsensical outputs. In DCGANs, a family of architectures is proposed that results in stable training across a range of datasets. It also allows for training higher resolution and deeper generative models.

The conditional GAN (CGAN) [144] is a conditional version of a generative adversarial network, which can generate data conditioned on the class labels. Stacked Generative Adversarial Networks (SGAN) [153] consists of a top-down stack of GANs, each trained to generate plausible lower-level representations conditioned on higher-level representations. SGAN is able to generate images of much higher quality than GANs without stacking.

In Minibatch discrimination [143] a new penalty term is added to the cost function, which prevents mode collapse. Mode collapse is when the generator collapses to a parameter setting where it always generates the same point. Feature matching [143] proposes a new cost function for the generator with the new objective that the generated data has to match the statistics of the real data.

Vanishing gradients is another problem that the original GAN suffers from. In gradient based learning methods, the parameters of the neural network are updated in each iteration of training proportional to the partial derivative of the error function with respect to that parameter. In such methods, the vanishing gradient prevents the weight from being effectively updated. In order to overcome the problem of vanishing gradients in GANs, Least Squares Generative Adversarial Networks (LSGANs) [145] adopts the least squares loss function for the discriminator. Wasserstein GAN (WGAN) [154] overcomes this problem and improves the stability of learning by proposing a new cost function using Wasserstein dis-

tance that provides a reliable gradient everywhere. WGAN enforces a Lipschitz constraint on the critic by using weight clipping. The authors of [155] found that weight clipping sometimes leads WGANs to generate only poor samples or fails to converge. They introduced gradient penalty (WGAN-GP) [155], which proposes an alternative to clipping weights by penalizing the norm of the gradient of the critic with respect to its input.

Energy-based Generative Adversarial Networks (EBGAN) [156] views the discriminator as an energy function that attributes low energies to the regions near the data manifold and higher energies to other regions. EBGAN allows the use of a wide variety of architectures and loss functions [156]. The work in [156] replaced the discriminator with an auto-encoder, where the energy is the reconstruction error. The EBGAN with the proposed architecture exhibits more stable behavior than regular GANs during training. Boundary Equilibrium Generative Adversarial Networks (BEGAN) [157] is based on the autoencoder for the discriminator as for the EBGAN, but using a loss derived from the Wasserstein distance. It uses a typical GAN objective with the addition of an equilibrium term which balances the discriminator and the generator [157].

In the standard GAN the generator is trained to increase the probability that fake data is real. Relativistic GANs (RGANs) [158] suggest the probability of real data being real should also decrease during the training. RGANs use a *relativistic discriminator*, which estimates the probability that the given real data is more realistic than a randomly sampled fake data and adopts the cost function relatively [159]. GANs with a relativistic discriminator are more stable and produce data of higher quality while its training is faster.

In this thesis, we use the generator module of GANs to learn the model of good quality speech signals. To assess the quality of speech we propose to compare the distribution of speech signals in the latent space. For this, we have to implement an inverted generator that maps the speech signals back into the latent space. To implement the inverted generator, a simple

architecture in the forward generator is desirable. Hence, in Chapter 5 we utilise the original GAN by Goodfellow [20] for learning the distribution of the spectrograms of speech signals. We use the same architecture used for CIFAR database in [20], and the algorithm successfully converged on our database. Hence we did not require the application of the techniques introduced above to overcome such problems.

As explained above, the GAN-based quality assessment proposed in Chapter 5, assesses the quality of speech by comparing the distribution of speech signals in the latent space. In the next section, we study different divergence metrics and how to utilise them to compare the distributions.

3.5 Divergence measures

In this section, we briefly review divergence between probability distributions, which is essential in order to read Chapter 5. Divergence is a measure to quantify the difference or discrepancy between two probability distributions $P(x)$ and $Q(x)$. It is a weaker notion than the distance as it does not have to be symmetric. Approximating a divergence between two distributions is used for various purposes in the statistics, information theory, and machine learning communities. In the following, we review the most popular divergence and distance measures that are used for training neural networks.

Kullback–Leibler divergence (KL) [160] is one of the most popular divergence measures in statistics and machine learning that has been used for decades in a wide range of inference problems. It is defined as:

$$\mathrm{KL}(P\|Q) = \int Q(x) \log\left(\frac{P(x)}{Q(x)}\right) dx. \tag{3.10}$$

One can define the scoring function $f_Q(x) = -\log Q(x)$ [161] and rewrite Equation (3.10) as

$$d_f(P,Q) = \mathbb{E}_{x\sim P} f_Q(x) - \mathbb{E}_{x\sim P} f_P(x) \tag{3.11}$$

Jensen Shannon divergence (JS) introduced by Rao [162] and generalised by Lin [163] is a symmetrised version of the Kullback-Leibler divergence. It is defined as:

$$\mathrm{JS}(P\|Q) = H\left(\frac{P+Q}{2}\right) - \frac{1}{2}H(P) - \frac{1}{2}H(Q), \tag{3.12}$$

where H is the Shannon entropy. Equation (3.12) can be expressed in terms of the Regarding Kullback-Leibler divergence as:

$$\mathrm{JS}(P\|Q) = KL(P\|\frac{P+Q}{2}) + KL(Q\|\frac{P+Q}{2}). \tag{3.13}$$

Integral probability metric (IPM) [164] is another popular family of distance measures defined as:

$$\mathrm{IPM}(P\|Q) = \sup_{f\in\mathcal{F}} |\mathbb{E}_{x\sim P} f(x) - \mathbb{E}_{x\sim Q} f(x)| \tag{3.14}$$

where $\mathcal{F}$ is a class of real-valued bounded measurable functions. The supremum in Equation (3.14) finds the function f, whose average value over P is most different from its average over Q. Wasserstein distance [165], and maximum mean discrepancy (MMD) [166] are two famous IPM metrics. In the following, we explain these two metrics in more detail and explain they are different only in the function class $\mathcal{F}$.

Maximum mean discrepancy (MMD) has been adopted in a variety of modern applications in machine learning and statistics. It considers the unit ball in a universal reproducing kernel Hilbert space (RKHS) [167] as the function class and defines the divergence as:

$$\mathrm{MMD}^2(P,Q) = \sup_{\|f\|_{\mathcal{H}_k}=1} \left(\mathbb{E}_{x\sim P} f(x) - \mathbb{E}_{x\sim Q} f(x)\right)^2, \tag{3.15}$$

where $\mathcal{H}$ represents an RKHS with k as its reproducing kernel. The choice of the unit ball in RKHS for MMD has two reasons: 1) it is *rich* enough that the MMD decreases to zero if and only if $P = Q$, and 2) it is *restrictive* enough for to converge quickly to its expectation when the sample size increases [166]. In terms of mean embeddings, MMD can be written as:

$$\mathrm{MMD}^2(P,Q) = \|\mu_P - \mu_Q\|^2_{\mathcal{H}_k=1}, \tag{3.16}$$

where $\mu_P = \mathbb{E}_{x\sim P}k(\cdot, x)$ and $\mu_Q = \mathbb{E}_{x\sim Q}k(\cdot, x)$. This indicates the MMD metric is analogous to the Euclidean distance between the mean elements of the two distributions in the Hilbert space [161].

Wasserstein distance or earth mover (EM) distance is defined as:

$$\mathrm{W}(P, Q) = \inf_{\pi\in\Pi(P,Q)} \mathbb{E}_{(x,y)\sim\pi}(\|x - y\|), \tag{3.17}$$

where $\Pi(P, Q)$ is the set of all joint distributions $\pi(x, y)$ with marginal P and Q. Intuitively the EM distance is defined as the optimal cost of transporting "probability" mass from x to y in order to transform the distribution P into the distribution Q. Under mild assumptions,$\mathrm{W}(P, Q)$ is continuous everywhere and differentiable almost everywhere [154]. Hence when optimised, it often behaves better than the KL and JS. This makes it more desirable to be used as the cost function for the generative models that are based on a relatively low-dimensional latent space. However, finding the infimum in (3.17) is not trivial.

The famous Kantorovich-Rubinstein theorem [168] shows that L1-Wasserstein distance is a particular case of Kantorovich metric and can be written as:

$$\mathrm{W}(P, Q) = \sup_{\|f\|_L\leq 1} \left(\mathbb{E}_{x\sim P}f(x) - \mathbb{E}_{x\sim Q}f(x)\right)^2, \tag{3.18}$$

where $\|f\|_L$ is the Lipschitz semi-norm of a bounded continuous real-valued function f. The final Equation (3.18) for Wasserstein distance is similar to Equation (3.15) at the starting point of MMD.

To solve the maximisation problem in Equation (3.18), WGAN considers function f to be a neural network and roughly approximates that using backpropagation. The supremum in Equation (3.18) is over all the 1-Lipschitz functions f. To enforce this in an approximate manner, WGAN uses weight clipping and clamps the weights to a fixed range after each gradient update.

In Chapter 5 we utilise the definition of divergence between the distributions to assess the quality of speech signals. We propose to project the

test signals into the latent space and rate them based on the divergence between the distribution of the latent variables and the prior model distribution defined for good quality signals. In Section 5.4, we study different divergence metrics for quality assessment and select the one that suits our purpose for rating the quality of speech.

3.6 Summary

In Chapter 2, we studied different types of speech quality assessment and reviewed the advantage of machine learning based quality assessment systems over conventional standards. In this chapter, we presented an introduction to machine learning and reviewed contemporary algorithms with shallow and deep architectures.

As explained in Chapter 2, several non-intrusive quality assessment methods have recently been proposed based on supervised machine learning models. The overall goal in Chapter 4 is to improve the estimation of the objective score of a speech utterance, so that in comparison with existing standards and methods, it has a high correlation with the scores obtained from human subjects.

In this chapter, we showed reasons why deep learning methods appear to be suitable for our purpose as they often outperform shallow machine learning methods. However, due to the unavailability of a large scale, labelled database, the supervised method proposed in Chapter 4 has a shallow architecture, and its performance is improved by enhancing the features. On the other hand, the unsupervised method proposed in Chapter 5 is based on deep learning as it does not require labelled data.

The overall goal in Chapter 5 is to implement the first unsupervised quality estimation system. In this method, we mimic the high dimensional functionality that exists in the brain and enables us to rate the quality. We aim to define a new criterion for quality, and the overall goal is that this criterion correlates with the scores from subject tests. In the following two

chapters, we explain these supervised and unsupervised quality assessment systems, respectively.

4

Supervised quality assessment

4.1 Introduction

Several non-intrusive quality assessment methods have recently been proposed based on supervised machine learning models. In these methods, the machine learning algorithms learn to estimate the quality of speech signals, where quality is defined as the outcome of a particular subjective quality estimation protocol. In supervised quality-assessment problems, the overall goal is to improve the estimation of the objective score of a speech utterance so that it has a high correlation with the scores obtained from human subjects.

Supervised learning based non-intrusive quality estimation, can be described as a multi-class classification or a regression problem, where the input is a set of signal features, and the output is its quality score. Signal features can be generated by executing a pre-processing algorithm on the input utterances. In many protocols, the human subjects are asked to rate the quality score with a discrete score [2]. This means our target variable

is a discrete score. Many protocols typically allow for five discrete scores [2]. In this case, the problem will be a multi-class classification problem. However, in the quality estimation literature, the mean score for an utterance is generally used, which is a continuous score computed based on the arithmetic mean value of subjective judgments [47]. For this case, the problem becomes a regression problem.

The ACR method [2] is the most commonly used subjective test procedure in telecommunication. In ACR, the human subjects are paid to listen to speech utterances under controlled conditions and rate them using a five-level impairment scale (1: bad, 2: poor, 3: fair, 4: good, 5: excellent). The subjective listening quality mean-opinion-score (MOS) is then computed for each speech file by averaging over the rating scores for all the subjects. In the supervised non-intrusive quality assessment, the main goal generally is to develop a regressor that predicts MOS values that are highly correlated with the MOS of subjective tests.

Some supervised learning algorithms, such as Bayesian methods, go further and use statistics to find a full predictive distribution instead of predicting a constant value. As explained in section 3.3, by using statistical methods in quality assessment, the objective of the quality assessment is to provide the predictive distribution for the quality of an utterance. Hence, results can be used to obtain both the *prediction* and the *precision*. For example, if the output has a Gaussian distribution, the predicted value is its mean, and its precision is the inverse of its variance.

We have implemented various supervised learning methods, including both Bayesian and non-statistical ones, and used them for quality assessment. The Bayesian algorithms we implemented are Variational Relevance Vector Machines (VRVM) [101], Fast Variational Sparse Bayesian Learning (SBL) [102], and Correlated Nystrom Views (XNV) [103], which are popular kernel-based methods in a nonlinear regression problem. The sparsity property of Variational RVM results in low computational cost and makes it useful for practical applications. The fast adaptive varia-

tional RVM decreases the complexity of the training procedure. Moreover, XNV reduces runtime by orders of magnitude compared to some other semi-supervised learning algorithms [103]. The non-statistical methods we used in our quality assessment are neural networks with drop-connect [169], stacked autoencoders [116], stacked denoising autoencoders [123], and generalized autoencoders [170]. To compare the performance of our machine learning based methods with each other and also with the P.563 standard, which is designed by humans, we tested them on the IITU-T Supplement 23 database. The evaluation metrics and test database are explained in sections 4.6.1 and 4.6.2 respectively. The results from our initial experiments with machine learning based methods that are listed above were the same, and none of them was better than the scores reported for the P.563 standard in [8].

Our experimental results from applying various machine learning models in non-intrusive quality assessment indicate that with the limited data we have, even rich machine learning algorithms do not enhance performance. From our results, and the results others reported in the literature, we concluded that the overall performance of speech quality assessment systems could be improved by either collecting more training data, which is usually expensive, or by enhancing the features [1]. The focus of the work explained in this chapter is to enhance the features that are input to the regressor. Our publication in 2017 IEEE International Conference on Acoustics, Speech and Signal Processing [1], presents the overall concept of this chapter.

To give a better understanding of the features, we first review the model architecture of the non-intrusive systems in section 4.2, and explain why enhancing the features is an important aspect of our work. Next we introduce two novel ideas that enhance the feature set.

The first idea is to augment the feature set with raw features that are presumably redundant [1]. We study the case where input features are noisy and illustrate that the proposed augmented feature set improves the

performance by reducing the effect of input noise [1]. We provide a more detailed analysis of this performance gain and its mathematical model in section 4.3.

The second idea we present in this chapter is the pre-processing method we apply to the data. This method redistributes the data to obtain pre-distorted features that facilitate the training. Section 4.4 explains the pre-processing method in more detail. Section 4.5 explains how these feature enhancement ideas are applied to quality assessment and discusses the aspects of implementation.

To demonstrate the effectiveness of our system, we evaluated it on the ITU-T Supplement 23 database, which is widely used as benchmark for comparing the performance of non-intrusive systems [14, 23, 26, 28, 61, 62]. The experimental results showed that our method outperforms contemporary single-ended quality assessment systems and current popular standards. Section 4.6 provides more details on the experimental results. This is followed by a summary of this chapter in section 4.7.

4.2 Model architecture of non-intrusive QA

Supervised learning based non-intrusive quality estimation proposed in this chapter is expressed as a regression problem, where the input is a set of features that describes the attributes of speech utterance, and the output is its quality score. In this section, we study two different types of architecture for non-intrusive systems in which features are extracted at different levels.

The classic view in regression problems is that if the information is irrelevant or redundant, then knowledge discovery during training is difficult. Therefore, in conventional machine learning algorithms, selecting a proper set of features is an important aspect of the overall performance [171, 172, 173]. Selecting a proper set of features usually improves the prediction performance of the predictors. It also provides faster and more

cost-effective predictors. Furthermore, it provides a better understanding of the underlying process that generated the data [174].

Lately, deep learning methods [116, 115, 175] have become very popular, particularly for building hierarchical representations of data. Deep learning models remove the need for feature engineering as they automatically develop feature representations from unlabeled data. Lower layers in deep architectures attempt to detect simple features to feed into higher layers, which detect more complex features [121]. Such neural network models are well suited to domains where large datasets are available [169]. Deep neural networks significantly outperform shallow ones in many large and complex systems. However, deep neural networks do not apply to the quality assessment method proposed in this chapter due to the small size labelled database that is publicly available as opposed to a substantial amount of data needed for training deep neural networks.

A non-intrusive quality assessment system that operates directly at the waveform level requires a large amount of data for learning a large number of parameters at the input level. Therefore, the non-intrusive quality assessment systems presented in the literature contain a front-end module, which pre-processes the speech signal and extracts information from the waveform to construct a feature vector. The feature vector must include the attributes that represent different types of distortion. Likewise, the supervised model proposed in this chapter has a front-end module, which computes features that describe the audio and are the input to the quality predictor.

The front-end module in non-intrusive quality assessment systems, decomposes the degraded signal voice into time frames and computes the physical features of the individual frames. An aggregation function is required to provide one score that represents the quality of the whole utterance. This section focuses on aggregation and where it is performed. Based on the literature, we conclude that the aggregation module could be placed either between the front-end system and the quality predictor

or after the quality predictor. In the following subsections, we explain how these two structures are different from each other and which model we favour.

In section 4.2.2 we describe the structure model in which the aggregation is performed over the predicted distortion of the frames. In section 4.2.1 we explain the model in which the aggregation is performed over the features of the frames.

4.2.1 Aggregation over the features

In this section, we explain the structure in which the aggregation module is the central component. Figure (4.1) shows the overall structure of this model. The front-end module decomposes the degraded signal voice into time frames and computes the features of individual frames. The aggregation component aggregates the features over all the frames of one utterance and computes the overall features for the utterance. The predictor then takes the overall features of one utterance to estimate its distortion and maps it to a MOS score. In this architecture, the temporal structure of the features is maintained and can be used for quality estimation.

There are several non-intrusive methods that have this structure [14, 23, 8], applying various aggregation approaches. For example, the speech quality assessment introduced in [14] assumes that the speech quality can be estimated from statistical properties of the per-frame features. Thus it calculates the moments independently for each per-frame feature, which gives a set of features that globally describe one speech utterance.

Aggregation over the features allows us to use the temporal structure of the features for quality estimation. Thus, at least in theory, the method will perform better than the method that we will introduce in section 4.2.2. For this reason, we favour this approach. The enhanced feature set proposed in section 4.5 contains the raw features from P.563 that has this architecture.

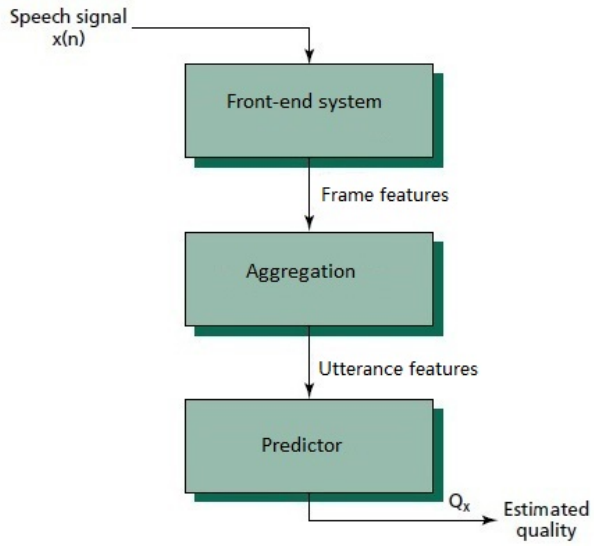


Figure 4.1: Architecture model for speech quality assessment system, where aggregation is performed over the features.

4.2.2 Aggregation over the predicted distortion

In this section, we explain the structure in which the aggregation module is placed after the predictor. Unlike the model explained in section 4.2.1, the distortion is predicted for individual frames and the aggregation is performed over the estimated distortion of all the frames in one utterance.

Figure (4.2) shows the overall structure of this model. The front-end module decomposes the degraded signal voice into time frames and computes the features of the individual frames. The predictor uses the features computed per-frame to estimate the distortion for individual frames. Importantly, this distortion is a scalar value for each frame. In the aggregation module, the individual distortions are aggregated to compute overall utterance distortion which will be later mapped to the quality score. ANIQUE+ [44] is an example of the speech quality assessment with such a structure.

Since the aggregation of the method of Figure (4.2) is performed on a

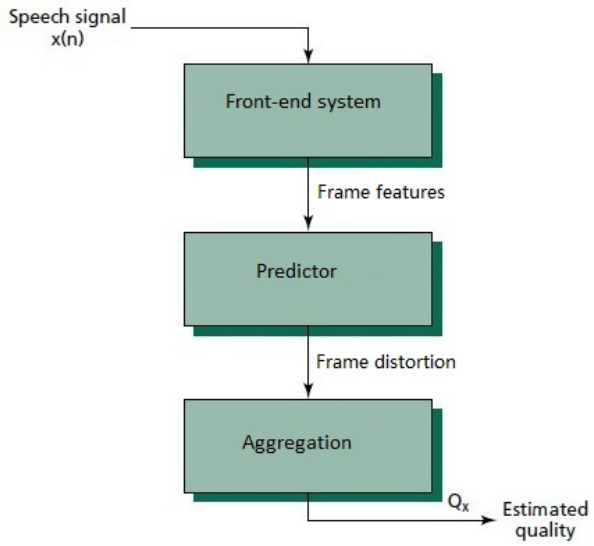


Figure 4.2: Architecture model for speech quality assessment system, where aggregation is performed over the predicted distortions.

per frame basis, the quality estimation cannot exploit the temporal structure of signal features other than the temporal structure in the scalar distortion. The enhanced feature set proposed in section 4.5 contains raw features from ANIQUE+, which has this structure. However, in order to exploit the temporal structure of signal features, we compute the moments of per-frame features. Section 4.5 explains this in more detail.

4.3 Feature set augmentation

In this section, we study the relationship between the number of features and the performance of machine learning algorithms. In the particular case where input features are noisy, we illustrate that the presence of redundant features in the input enables the machine learning systems to access more precise information and hence results in better performance. The enhanced feature set proposed in section 4.5 is an augmented fea-

ture set that benefits from redundant features and aims to improve performance by reducing the effect of input noise.

The term "curse of dimensionality" was first introduced by Bellman [176]. The term states that the convergence of predictors to the true value of a smooth function is very slow if the dimensionality of the input feature set is large [177]. This effect is justified due to the exponential growth of hypervolume as a function of dimensionality in Euclidean space [178]. The term also states that where the number of training data is fixed, having higher-dimensional features can make the predictor more prone to overfitting. Consequently, a technique called feature selection [172, 174, 179] is often an essential data processing step prior to applying conventional learning algorithms. Feature selection is the removal of irrelevant and redundant information, and it often improves the performance of traditional machine learning algorithms.

The results reported in [180] are one example of benefiting from high dimensional features based on collecting sufficient training data. The effectiveness of high dimensional features is also reported based on developing complex algorithms. For example, the complex methods in [181, 182, 183] use multiple feature combination and boosting algorithms that enable the system to achieve higher performance by managing high dimensional features properly. Deep learning approaches, including models such as CNN [121], RNN [135, 184], GAN [20, 185], are other examples that benefit from high dimensional features where sufficient training data is available. Deep neural networks are computationally tractable even when applied to high dimensional inputs and have gained significant interest as they outperform shallow neural networks.

To conclude, as stated in [174] "including presumably redundant variables might result in a performance gain". Although this statement is well-known and a large number of papers (e.g., [179, 186, 187]) refer to it, the explanation in [174] is qualitative, and a detailed analysis of the performance gain does not exist [1]. In this section, we study the scenario in

which redundant features represent the same information, but contain independent noise [1]. We analyse how enlarging the feature dimensionality improves the performance of linear machine learning models. In general this is not feasible to analyse, so an analysis of the linear case is presented. The experimental results suggest that the performance gain from redundant features can also be generalised into nonlinear learning problems.

In the following subsections, we study the relationship between the number of features and the performance of the system for two different scenarios. In the first scenario, the ground truth model has few features, and we increase the dimensionality of the feature set by adding redundant features. In the second scenario, the ground truth model has many features, and we enlarge the feature set by observing the features that contain new information [1]. Although in both scenarios, adding more features results in improved performance, the performance gain has different behaviours. This is because in the first scenario, we include redundant features to decrease the effect of noise, whereas, in the second scenario, we are adding new information by including the missing features [1]. The underlying model for a linear quality estimation is first presented in section 4.3.1. Then the different behaviours of the two models above are studied in Sections 4.3.2 and 4.3.3.

4.3.1 Underlying model for linear quality estimation

We aim to develop a MOS estimator based on a set of observed features. In this section, we explain how to model this with a linear regressor. $\mathbf{x}$ and $\mathbf{y}$ are underlying and observed feature vectors respectively. We represent the associated random variables with capital letters X and Y.

Let us assume X is distributed normally with a zero mean and are independent:

$$X \sim \mathcal{N}(0, R_X). \tag{4.1}$$

Without loss of generality, we assume that the covariance matrix of X is

constant along the diagonal, $R_X = I\sigma_X^2$. The MOS is computed as

$$MOS = a^T \mathbf{x}. \tag{4.2}$$

We estimate MOS based on a set of observed features $\mathbf{y}$ as

$$\hat{MOS} = b^T \mathbf{y}. \tag{4.3}$$

The random variable V is the prediction error [1] and defined as

$$V = a^T X - b^T Y. \tag{4.4}$$

Vector b must be estimated from Y aiming to minimize the prediction error on the training data [1]. In the following two subsections, we define two different families of the features and model the relationship between the number of observed features and the prediction error.

4.3.2 Model behaviour for redundant features

In this section, we assume the observed features in $\mathbf{y}$ are redundant and contain the same information, but have independent noise [1]. In the following, we model the relationship between the number of observed features and the performance of the linear quality estimator.

Let us assume the random underlying feature vector X contains the independent features that represent the data. In practice, X cannot be observed and hence is unknown. In contrast, the random observed feature vector Y represents a feature set observable in the real world. Y includes noise and is likely to contain features that are correlated. We define two different types of noise: 1) intrinsic noise and 2) observation noise. Intrinsic noise is considered to be the noise that is naturally part of the underlying features and is inevitable. Hence, it always is joined to X. Observation noise occurs due to errors in computation and measurement and is included in the observed features. Consequently, the random observed

feature vector Y is a transformation of the random underlying feature vector X into another space with a higher dimension [1]. Hence, Y contains redundant information and is of the form

$$Y = C(X + U) + W. \tag{4.5}$$

C is a transformation matrix, and U and W are random noise vectors called *intrinsic noise* and *observation noise* respectively [1].

The prediction error from 4.4 will be

$$V = (a^T - b^T C)X - b^T CU - b^T W. \tag{4.6}$$

X, U, and W are independent and it follows that:

$$\begin{aligned} \sigma_V^2 = \mathrm{E}[X^T(a^T - b^T C)^T(a^T - b^T C)X \\ + U^T C^T b b^T C U + W^T b b^T W]. \end{aligned} \tag{4.7}$$

σ_V^2 is scalar and we can write $\sigma_V^2 = \mathrm{tr}[\sigma_V^2]$. Exchanging the linear operators, the expectation and the trace, and using the cyclic property of the trace [1], we can write

$$\begin{aligned} \sigma_V^2 = b^T(CR_X C^T + CR_U C^T + R_W)b \\ - 2a^T R_X C^T b + a^T R_X a. \end{aligned} \tag{4.8}$$

The optimal b^* must satisfy

$$2(CR_X C^T + CR_U C^T + R_W)b^* - 2CR_X a = 0, \tag{4.9}$$

and it follows that

$$b^* = (CR_X C^T + CR_U C^T + R_W)^{-1} CR_X a. \tag{4.10}$$

Substituting (4.10) back into (4.8) gives

$$\begin{aligned}
\sigma_E^2 &= a^T R_X C^T (C R_X C^T + C R_U C^T + R_W)^{-1} C R_X a \\
&\quad - 2a^T R_X C^T (C R_X C^T + C R_U C^T + R_W)^{-1} C R_X a + a^T R_X a \\
&= a^T R_X a - a^T R_X C^T (C R_X C^T + C R_U C^T + R_W)^{-1} C R_X a && (4.11) \\
&= a^T R_X C^T C^{T\sharp} a \\
&\quad - a^T R_X C^T (C R_X C^T + C R_U C^T + R_W)^{-1} C R_X C^T C^{T\sharp} a && (4.12) \\
&= a^T R_X C^T (I - (C R_X C^T + C R_U C^T + R_W)^{-1} C R_X C^T) C^{T\sharp} a && (4.13) \\
&= a^T R_X C^T (C R_X C^T + C R_U C^T + R_W)^{-1} (C R_U C^T + R_W) C^{T\sharp} a && (4.14)
\end{aligned}$$

The underlying random features in X are independent. Without loss of generality, we assume $R_X = I_{d\times d}$ (so it sets the scale), $R_U = hI_{d\times d}$, and $R_W = gI_{t\times t}$, where d and t are the dimensionality of X and Y respectively, and g and h are small [1]. These assumptions led to

$$\sigma_V^2 = a^T (C^T C + hC^T C + gI)^{-1} (hC^T C + gI) a, \tag{4.15}$$

where we reduced the dimensionality from t to d. In the following we study the relation between σ_V^2 and the number of features, and analyse its behaviour by enlarging the feature set.

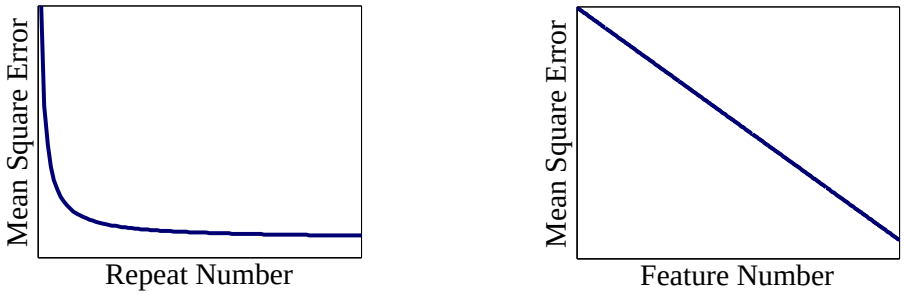
For simplicity we initially consider the "repeat" case, where C is a tall matrix of stacked identity matrices [1]. Let us repeat each feature n times. Then $C^T C = nI$ and we get

$$\begin{aligned}
\sigma_V^2 &= a^T (nI + hnI + gI)^{-1} (hnI + gI) a && (4.16) \\
&= \frac{g + nh}{g + n(h+1)} a^T a. && (4.17)
\end{aligned}$$

Figure (4.3.a) models this behaviour. This is clearer [1] if we do not have intrinsic noise and set $h = 0$:

$$\sigma_V^2 = \frac{g}{g+n} a^T a, \tag{4.18}$$

in which σ_V^2 goes to zero if n is very large. However, because of the presence of intrinsic noise, the floor for (4.17) is $\sigma_V^2 > \frac{h}{h+1} a^T a$.



(a) Model with redundant features (b) Model with subset of features

Figure 4.3: The abstract behaviour of two different models for gaining performance by enlarging the feature set.

Now consider the more general case where instead of repeating the features, C is a tall $n \times d$ matrix, where n is the dimensionality of the observed features and d is the dimensionality of underlying features [1]. We assume elements of C are i.i.d and have normal distribution

$$C \sim \mathcal{N}_N(0, \Sigma_c), \tag{4.19}$$

where Σ_c is a diagonal matrix with diagonal elements equal to σ_c [1]. We aim to estimate the behaviour of σ_V^2 by finding the expectation of Equation (4.15) over C [1]. Since the underlying behaviour of the model is not analytically tractable, we limit the case [1] and study the behaviour of this model where $h = 0$. We analyse the main aspect of the model behaviour by re-writing Equation (4.15) as

$$\sigma_V^2 = g a^T (C^T C + gI)^{-1} a, \tag{4.20}$$

$$E_C[\sigma_V^2] = g a^T E_C[(C^T C + gI)^{-1}] a, \tag{4.21}$$

Since the elements of C are i.i.d and have normal distribution, $C^T C \sim \mathcal{W}_N(\Sigma_c, n)$ has a Wishart distribution [188] with the mean value

$$E[(C^T C)_{ij}] = \begin{cases} n\sigma_c & i = j \\ 0 & i \neq j \end{cases}. \tag{4.22}$$

g is small compared to n. Hence we approximate $E_C[(C^TC + gI)^{-1}]$ with $E_C[(C^TC)^{-1}]$ and it follows that

$$E_C[\sigma_V^2] \sim ga^T E[(C^TC)^{-1}]a. \tag{4.23}$$

Since C is a tall matrix with normal distribution, $(C^TC)^{-1} \sim \mathcal{W}^{-1}(\Sigma_c^{-1}, n)$ has an Inverse Wishart distribution [188]. With the assumption $\sigma_c = 1$ we [1] have [1]

$$E[(C^TC)_{ij}^{-1}] = \begin{cases} \frac{1}{n-d-1} & i = j \\ 0 & i \neq j \end{cases}, \tag{4.24}$$

Using this in (4.21) we can estimate the mean of σ_V^2 as

$$E_C[\sigma_V^2] \sim \frac{g}{n - N - 1} a^T a, \tag{4.25}$$

where n is the dimensionality of observed features [1]. Again as expected, the variance of the error is decreasing by enlarging n. Hence if the model of (4.5) is correct, it motivates the augmented feature set with redundant features for higher performance [1].

4.3.3 Model behaviour for insufficient features

In this section, we [1] assume the dimensionality of the observed feature vector **y** is smaller than the dimensionality of underlying feature set **x** and develop a linear MOS estimator. In the following, we model the relationship between the number of observed features and the performance of the linear quality estimator [1].

Let us assume the random observed feature vector Y is a subset of the random underlying feature set X and is of the form:

$$Y = SX + W, \tag{4.26}$$

where W is random observation noise and

$$S = [I_{n\times n} \quad 0_{(N-n)\times(N-n)}]. \tag{4.27}$$

n and N are the number of selected features and full features respectively [1]. Accordingly, the prediction error is:

$$V = (a^T - b^T S)X - b^T W, \tag{4.28}$$

and we [1] aim to minimise its variance:

$$\sigma_V^2 = a^T R_X a + b^T (S R_X S^T + R_W) b - 2a^T R_X S^T b. \tag{4.29}$$

The optimal b^* must satisfy

$$2(S R_X S^T + R_W) b^* - 2S R_X a = 0, \tag{4.30}$$

and it follows that

$$b^* = (S R_X S^T + R_W)^{-1} S R_X a. \tag{4.31}$$

Let us assume $R_X = I_{N\times N}$ (to set the scale) and $R_W = g I_{n\times n}$. Using b^* in (4.29) we obtain

$$\sigma_V^2 = a^T [I_{N\times N} - S^T (I_{n\times n} + g I_{n\times n})^{-1} S] a \tag{4.32}$$

$$= \sum_{i=1}^{N} \lambda_i a_i^2, \tag{4.33}$$

where $\lambda_i = 1$ if $i > n$ and $\lambda_i = \frac{g}{g+1}$ if $i \leq n$. We [1] assume $a_i \sim N(E(a_i), \sigma_i^2)$ and so

$$E(\sigma_V^2) = \frac{N - \frac{n}{g+1}}{N} \sum_{i=1}^{N} E(a_i^2). \tag{4.34}$$

Equation (4.34) indicates that the variance of the estimation error and the number of selected features have a linear relationship, whereas in the model behaviour for redundant features is nonlinear [1]. Figure (4.3) illustrates the abstract behaviour of this model, and the model with redundant features from the previous section. Because of the observation noise, the floor for (4.34) is not zero and has the value of $(1 - \frac{1}{g+1}) \sum_{i=1}^{N} E(a_i^2)$.

4.4 Pre-processing features

In the previous section, we illustrated that an augmented feature set is beneficial for better performance of machine learning algorithms. The focus of this section is to enhance the feature set by pre-processing the features and redistributing them to have a smooth distribution.

It has been shown [189, 190] that preprocessing of the data can often have a significant impact on the performance of a machine learning algorithm. Consequently, modern quality estimation systems that are based on machine learning generally exhibit sensitivity to the distribution of data and how the data are presented. Standardisation is a preprocessing type that is important for many neural networks. In this section, we first briefly review the two most common techniques that are used for standardising the input of a neural network and then introduce a standardisation method to transform data to have a smooth and light-tailed distribution, which leads to a better predictor.

In the machine learning area, standardising usually refers to a transformation that is performed on the input data to scale them into an acceptable range for the network, or adjust its distribution to either meet the assumptions or facilitate the training. Standardizing the inputs of neural networks often receives little attention in the literature, mainly because insufficient prior information is available about the data or, if such information exists, it is too application-specific. Hence it is common in many algorithms (e.g., k-means [191], k-nearest neighbors [192], Ridge Regression [193], Gaussian Radial Basis Function Networks [194], Support Vector Machine (SVM) [195]) to simply standardise each feature to either the same range or the same standard deviation using one of the following methods:

- Min-max normalisation [196]: This method of normalisation will linearly scale input data into the appropriate range, which is typically the range of $[-1, 1]$ or $[0, 1]$. A linear scaling requires that the minimum and maximum values associated with the features be found or

estimated by an expert person in a given domain.

- Z-score normalization [196]: This method normalises the input data to have zero mean and unit variance. This can be done by subtracting the mean from each feature, then dividing the values of each feature by its standard deviation.

The two methods above are crucial and play an important role in the success of machine learning methods such as RBF [197] and pattern classification tasks based on PCA [198]. They are also important for getting good results from weight decay [199].

A number of machine learning models have been built with the implicit assumption that the distribution is normal or uniform, and hence cannot perform as designed if the input data is not appropriately distributed [200]. Even those models that can cope with irregular distributions will be assisted if the distributions are comparatively regularised [200]. Hence a standardisation method that adjusts the distributions appropriately is desirable.

Although the two common methods explained above adjust the variance of the data, they do not change the shape of the distribution of data and hence do not facilitate a smooth distribution. There are other methods that aim to increase the uniformity of data based on clipping off the ends of the distribution. These methods utilise statistical measurement to remove the outliers and spread out the distribution of the data [201]. However, in such methods, the threshold value for outliers is computed heuristically. These methods also make an assumption that the features have a normal distribution.

Other approaches [202, 203] exist that assume the data is non-normal and aim to increase the normality of the data. However, they assume that the data has a smooth distribution that is pushed to one side. Hence they transform data to remove the skew exclusively. Nevertheless, features with non-smooth distributions (e.g. multimodal data or data with outliers)

are likely to occur in many machine learning application areas, including speech quality estimation. As proposed in the previous section, using a large number of features is beneficial for better performance. The usage of a large number of features naturally leads to the inclusion of features that have poor behaviour [1].

The novelty we introduce in this section is to perform a pre-distortion operation to obtain pre-distorted features with a smooth and light-tailed distribution for the specific goal of facilitating the learning of the mapping from the features to speech quality. Section 4.4.1 presents a method to pre-distort the features. Section 4.4.2 discusses how to implement the pre-distortion operation.

4.4.1 Pre-processing method description

Our proposed method is to pre-distort at least a subset of the features individually, so that the pre-distorted features have a smooth and light-tailed distribution. In this section, we discuss the pre-distortion operation in more detail. The approach selected is a standard procedure for mapping a random variable with a particular distribution to a new random variable with another distribution.

To create a mapping from an observed feature y to a pre-distorted feature, it is convenient at a conceptual level to first map to a pre-distorted feature with a uniform probability distribution v. If we observe a realization y of the random variable Y with the cumulative distribution $F_Y(y)$, then the corresponding realization of V is $v = F_Y(y)$, which has a uniform distribution on the interval $[0, 1]$ [204]. The flow chart for this method is presented in Figure (4.4).

The random variable V does not have outliers and is an extreme example of a light-tailed (or zero-tail) distribution. Hence it does not have regions for which it is difficult to learn the relationship between input and output. The information of the relationship is not lost as the mapping between Y and V is one-to-one. As explained in the next section, the estimate

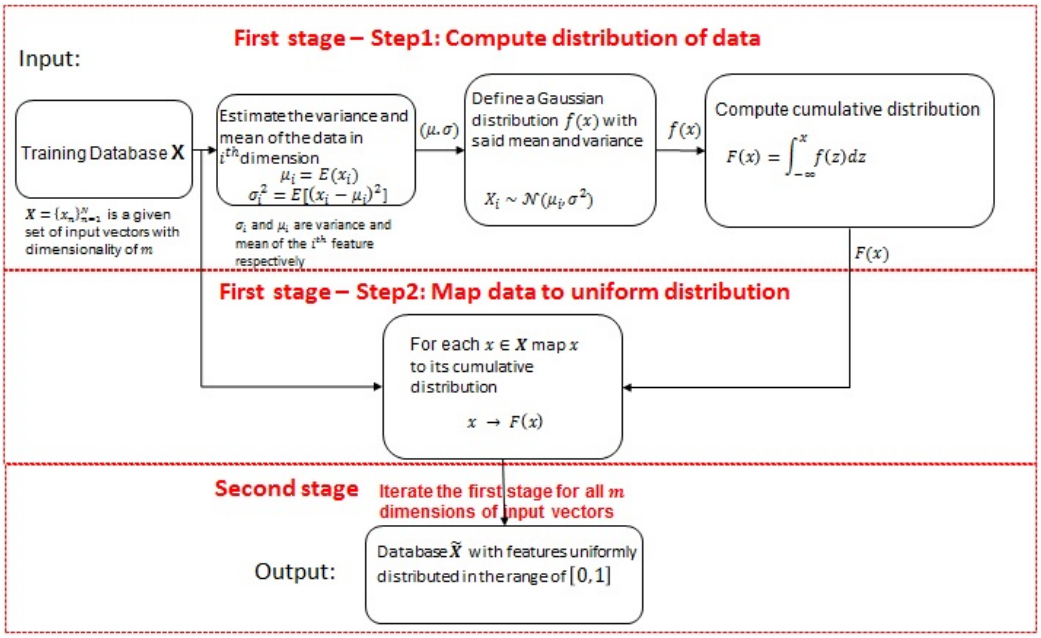


Figure 4.4: Flow chart of the proposed method for standardising the input of neural network to uniform distribution.

of F_Y is more accurate when more data is available. Thus when the training database is large, there is no significant drawback using V rather than Y as the input for the quality estimation system, but it is advantageous because of its smooth distribution.

If a (non-zero) light-tailed distribution $f_W(w)$ is desired, it is possible to map the feature to a new random variable W. To obtain the pre-distorted feature, W, we apply the mapping $w = F_W^{-1}(F_Y(y))$ to each observed feature realization y, where $F_W^{-1}(\cdot)$ is an inverse mapping.

4.4.2 Pre-processing method implementation

In the proposed method explained in the previous section, the cumulative distribution F_W of the desired feature must be designed by the user. However, the estimation of the cumulative distribution F_Y of the observed feature is required. This section explains the methods to estimate F_Y and then discusses the implementation aspects for mapping F_W^{-1}.

Existing methods can be used for the estimation of F_Y. A first illustrative method to estimate the cumulative distribution of Y is to utilise histograms [205]. A second illustrative method is based on Gaussian mixture distributions. Established methods, such as expectation maximisation (EM) [206] can be used to estimate the parameters of the order-Q Gaussian mixture distribution of Y from a given set of data $D = y_1, y_2, ..., y_N$. Note that a larger cardinality N of D facilitates a larger order Q. This leads to approximate any probability distribution to the desired precision for sufficiently high Q. Hence, we utilise this second method and employ the fitgmdist function in Matlab 9.9.0 library to estimate the parameters of the distribution of the features that are presumed to have order-3 Gaussian mixture distribution. In the following, we discuss the mapping F_W^{-1}.

If $W = V$ (a uniform distribution on [0,1]) then the mapping F_W^{-1} is trivial. It is noted that Gaussians are light-tailed and may work well as the input to a machine-learning system. If W is desired to be a Gaussian random variable, then F_W^{-1} is to be an inverse mapping of the cumulative

function of a Gaussian. With the assumption that the desired pre-coded feature W has unity variance, the cumulative distribution function [207] is defined as

$$F_W(w) = \frac{1}{2}[1 + \mathrm{erf}(-\frac{w}{\sqrt{2}})]. \tag{4.35}$$

Thus for this case

$$F_W^{-1}(F_Y(y)) = \sqrt{2}\mathrm{erf}^{-1}(2F_Y(y) - 1). \tag{4.36}$$

The erf and inverse functions are available in many platforms, hence F_W^{-1} can be computed for any value $F_Y(y)$.

4.5 Enhanced feature set for quality estimation

The supervised non-intrusive quality assessment system proposed in this chapter is based on the extraction of features that capture the information from a speech signal and represent different types of distortion. This section describes the proposed enhanced feature set that leads to improved prediction accuracy of the single-ended quality assessment.

As discussed in Section 2.2, the non-intrusive quality estimation P.563 and ANIQUE+ are the two existing standards and naturally form an excellent reference for our work. Therefore, we [1] built our input vector so that it contains both the features extracted from P.563 and ANIQUE+. Since P.563 and ANIQUE+ are designed for for narrowband speech, our system requires to downsample the speech files to 8 kHz if they are wideband.

The feature sets from both standards P.563 and ANIQUE+ are expected to represent similar information about the quality of the speech [1]. Hence our input vector is likely to hold redundant features [1]. However, as illustrated in Section 4.3, we hypothesise that the quality assessment system benefits from this redundancy as it results in reducing the impact of input

noise [1]. In the following, we describe the procedure we performed to build our feature set from P.563 and ANIQUE+.

As explained in Section 2.3, P.563 analyses the speech signals using several modules and determines a set of characterising signal parameters. The algorithm then uses a restricted set of the key parameters to determine a distortion class. Furthermore, the assigned distortion class and the key parameters are then used to predict the speech quality. Naturally the 43 characterising signal parameters form an informative global feature set for the quality assessment platforms [1].

In ANIQUE+, as explained in Section 2.4, the *Articulation Analysis Block* decomposes the incoming speech signal into successive time frames that are classified into *active speech* or *audible background noise* frames [1]. The algorithm then computes the local feature vector for each frame, which has the dimensionality 69. The per-frame features are used to predict one scalar value that represents the distortion of each frame, which is then aggregated over the duration of the signal to estimate its perceptual distortion.

In ANIQUE+, only one scalar value represents the distortion of each frame, and hence, it does not exploit the temporal structure of the signal features. In this work, we aim to improve the predictive accuracy of the quality assessment by considering the influence of the temporal statistics on the perception of the quality of an utterance. Hence, we [1] use the method suggested in [14] and converted the 69 per-frame features generated with ANIQUE+ to per-utterance features by computing the first four moments of the features over the active speech frames of the signal. In this approach, we [1] hypothesise that the speech quality can be estimated from statistical attributes of the per-frame features, and their probability distributions are described with their mean, variance, skewness, and kurtosis [14]. Thus for each per-frame feature Φ_i, we obtain four global features, $\mu_{\Phi_i}, \sigma_{\Phi_i}, s_{\Phi_i}, k_{\Phi_i}$, that describe the speech utterance. We define the global feature set Ψ, which contains 276 per-utterance features that are

computed by aggregation over the 69 ANIQUE+ per-frame features:

$$\Psi = \{\mu_{\Phi_i}, \sigma_{\Phi_i}, s_{\Phi_i}, k_{\Phi_i}\}_{i=1}^{69}. \tag{4.37}$$

$\mu_{\Phi_i}, \sigma_{\Phi_i}, s_{\Phi_i}$, and k_{Φ_i} are mean, variance, skewness, and kurtosis of the feature Φ_i that is computed over the speech active frames of the signal.

The feature sets generated from ANIQUE+ and P.563 have a different nature. However, they both represent the same information about the perceived quality of speech [1]. As illustrated in section 4.3, the quality assessment system benefits from redundancy. Hence, we built the augmented feature set Σ with the dimensionality 319 by accumulating 43 features computed from P.563 and 276 features extracted from ANIQUE+:

$$\Sigma = \{\Xi, \Psi\}, \tag{4.38}$$

where Ξ and Ψ contain global features from P.563 and ANIQUE+, respectively.

Finally, to facilitate the training, we standardise the features using the method proposed in section 4.4. We [1] standardise each feature in Σ to obtain a pre-distorted feature x_i with a uniform probability distribution and built our final enhanced feature set $X = \{x_i\}_{i=1}^{319}$. The experimental results reported in the next section demonstrate the effectiveness of our proposed feature set.

4.6 Experimental results

Section 4.6.1 explains the evaluation metrics and is followed by section 4.6.2 that describes the database we used in our experiments. The experimental results in section 4.6.3 validate that the quality assessment system benefits from redundant features. Finally, section 4.6.4 presents the experimental results from the quality assessment systems we developed with the enhanced feature set, and compares its performance with the existing methods.

4.6.1 Evaluation Metrics

To analyse our system and compare it with other non-intrusive models, we computed the Root Mean Squared Error (RMSE) and Pearson Correlation Coefficient (PCC), which are commonly used in this field. We report both per-file and per-condition results for RMSE and PCC. The per-condition score is generally better than the per-file score as it reduces material dependence [208].

RMSE measures the closeness of predicted scores to subjective MOS based on the mean square of the residual errors:

$$RMSE = \sqrt{\frac{\sum(x_i - y_i)^2}{N}}. \tag{4.39}$$

x_i and y_i are the subjective and predicted MOS, respectively. Equation 4.39 is used for both per-file and per-condition RMSE. In calculating per-file RMSE, x_i and y_i are the subjective and predicted MOS of utterance i. In calculating the per-condition RMSE, we first average the predicted and the subjective MOS over the conditions. In this case, x_i and y_i are subjective and predicted MOS averaged over the utterances that are degraded under condition i. Consequently, N is the number of utterances for computing per-file RMSE, or the number of conditions for computing per-condition RMSE.

Likewise, we reported both per-file and per-condition PCC. PCC is based on Pearson's formula and gives an alternative view of the closeness of the fit between predicted MOS and the subjective MOS:

$$PCC = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum(x_i - \bar{x})^2 \sum(y_i - \bar{y})^2}}. \tag{4.40}$$

$\bar{x}$ and $\bar{y}$ are the average over x_i and y_i.

Per-condition PCC is the most commonly reported score [12]. A non-intrusive quality assessment is considered to have a high performance if condition-averaged predicted MOS have a high correlation with condition subjective MOS.

In order to make a direct comparison between objective and subjective scores, it is common practice [7] to perform a third-order polynomial regression before computing per-condition PCC. The regression is monotonic so that information is preserved, but it eliminates the influence of irrelevant factors in subjective votes such as the preferences of individual subjects, or the context of the experiment. As it is common practice, we also perform a third-order polynomial regression to map the condition-averaged predicted scores onto the condition subjective MOS. Hence, y_i in Equation 4.39 and Equation 4.40 is the mapped condition-averaged predicted score for condition i.

In the third-order polynomial regression, the subjective scores are regressed as a function of objective scores. We employ the Polyfit function in the Matlab 9.9.0 library to estimate the parameters of the polynomial regressor. Then we employ the polyval function in the Matlab 9.9.0 library to map the objective scores to new scores that are expected to be more correlated to the subjective MOS.

4.6.2 Database

To evaluate our method with real data, we employed the ITU-T coded-speech data set, Supplement 23 [34], which is the only labelled database that is publicly available and is commonly used for the evaluation of objective speech quality systems. ITU-T Suppl23 provides speech material, and related subjective test plans and scores. Since our experiment was limited to Supplement 23, we used k-fold cross-validation techniques [209] to assure our system was not overfitting or giving a misleading result on the given database.

ITU-T Suppl23 was originally designed to characterize the subjective performance of the 8 kbit/s codec that has been proposed for adoption as per ITU–T Recommendation G.729 [210]. Supplement 23 contains three experiments. Experiment two uses the comparative category rating (CCR), which needs reference signals. Our non-intrusive system is independent

of the reference signal, hence, experiment two is not suitable for our purpose. We used experiments one and three that are scored based on the absolute category rating (ACR).

Experiment one examines the performance of G.729 codec interworking with other ITU–T speech coding standards (G.711, G.726 , and G.728) and regional speech coding standards used in digital cellular applications (full-rate GSM, RPE-LTP:GSM–FR, North American VSELP: IS–54, half-rate Japanese digital cellular: RCR Std 27C). The conditions defined in experiment one include encodings by each of these single codecs (including G.729), and also the encodings by the combination of two or three codecs. The combination conditions have been defined to represent the combinations of codecs that are likely to occur in real applications, including connections involving a mobile link, where G.729 codec is used for a wireless system, or where wireless mobile terminals are connected over trunks with the G.729 codec as the network codec [211]. The full specifications of the 44 conditions in experiment one are given in Appendix A.

Experiment three examines the effect of channel degradations on the G.729 codec. This experiment includes conditions that evaluate G.729 codec under detected frame erasure, and random bit error channel degradation conditions. Table A.2 in the Appendix gives the full specifications of 50 conditions in experiment three.

The coded-speech data set Supplement 23 is delivered on three CD-ROMs; each CD-ROM is allocated to one experiment. The database in experiments one and three has seven datasets, which came from different organizations with different languages: CNET (France), CSELT (Italy), Nortel (formerly BNR, Canada) and NTT (Japan). The seven data sets contain a total of 1328 speech files. Speech samples for these experiments consist of two short sentences and are approximately 8 seconds long. Speech occupies 80 to 90 percent of this time, and the remaining 10 to 20 percent is inter-sentence pauses. All the speech materials are recorded in 16-bit linear PCM (binary) files with a PC-format (low-byte first) [34]. Each speech

file in Supplement 23 is scored by 24 subjects; the average of these scores form the mean opinion score (MOS).

Supplement 23 has limited data samples and is considered to be a small database. Hence, cross-validation is commonly used to evaluate non-intrusive quality systems with Supplement 23. Cross-validation [209] is a resampling procedure, which reduces problems like overfitting or selection bias. One procedure, called k-fold cross-validation, divides data into k groups (or folds) of approximately equal size. K-fold cross-validation involves k iterations. At each iteration, one fold is treated as a test set, and the system is trained with the remaining $k - 1$ folds [209].

The voice files in experiments one and three from Supplement 23 come in seven datasets. Hence it is common [14, 23, 26, 28, 8, 61, 62] to validate the system applying 7-fold cross-validation, leaving one dataset out in each iteration. The seven datasets in Supplement 23 came from different labs with different languages and conditions. Hence, the distribution of the data set that is used for the test might be very different from the other six data sets used for training. Consequently, the test results are expected to vary from one iteration to another depending on how similar the test samples are to the samples used for training. Some methods such as [24, 29, 46, 66] used a different type of 7-fold cross-validation on supplement 23 in order to distribute data evenly into test and training groups. For this, they pooled all data sets together and randomly divided the voice files into seven groups. Applying cross-validation on the mixed data increases the similarity of test data and training data and hence increases the scores. However, there is no real gain in the performance of the system. In this thesis, we use both types of cross-validation for different purposes.

In our experiment with the features in section 4.6.3, we pooled all data sets together and randomly divided the speech files into seven groups. Dividing the data into the groups with a similar distribution results in stable behaviour at different iterations of the cross-validation. An identical distribution of samples in test and training sets permitted us to analyse

the effect of redundant features on the performance of the system independent of the effect of a variety of conditions in the test voice files. This enabled us to study the behaviour of our system and validate the relationship between the number of features and the performance of the system.

The experiment in section 4.6.4 is to evaluate the performance of our final system. To have a fair comparison with the scores reported in the literature, we used datasets of Supplement 23 as folds and apply 7-fold cross-validation procedure, leaving one data set out in each iteration. That is, six data sets of Supplement 23 are used for training, and the remaining data set is used for the test. Hence the voice files in the test set come from a laboratory that is different from the laboratories that generated voice files in the training set, and the test speech might be in a different language. Consequently, the scores reported in section 4.6.4 are expected to be lower than the scores in section 4.6.3.

4.6.3 Experiment with redundant features

This section explains the experiments that we performed on the ITU-T Suppl23 database to verify speech quality assessment benefits from redundant features. As explained in section 4.6.2, we pooled all data sets together and randomly divided the 1396 speech files into seven groups to apply seven-fold cross-validation. We evaluated the relationship between the number of features and the performance of the quality predictor. We demonstrated that our experimental results fitted the behaviour of the model with redundant features better.

To study the relationship between the number of the features and the performance of the quality predictor, we initially used a simple linear regressor. We evaluated its performance on the subset of features, X_n, which is randomly selected from the enhanced feature set X that was explained in section 4.5. The parameter $n = 10, 20, 30, ..., 310$ is the cardinality of X_n. To observe the statistical behaviour of the model, we repeated the experiments with 20 random subsets of features for each value of n. The results

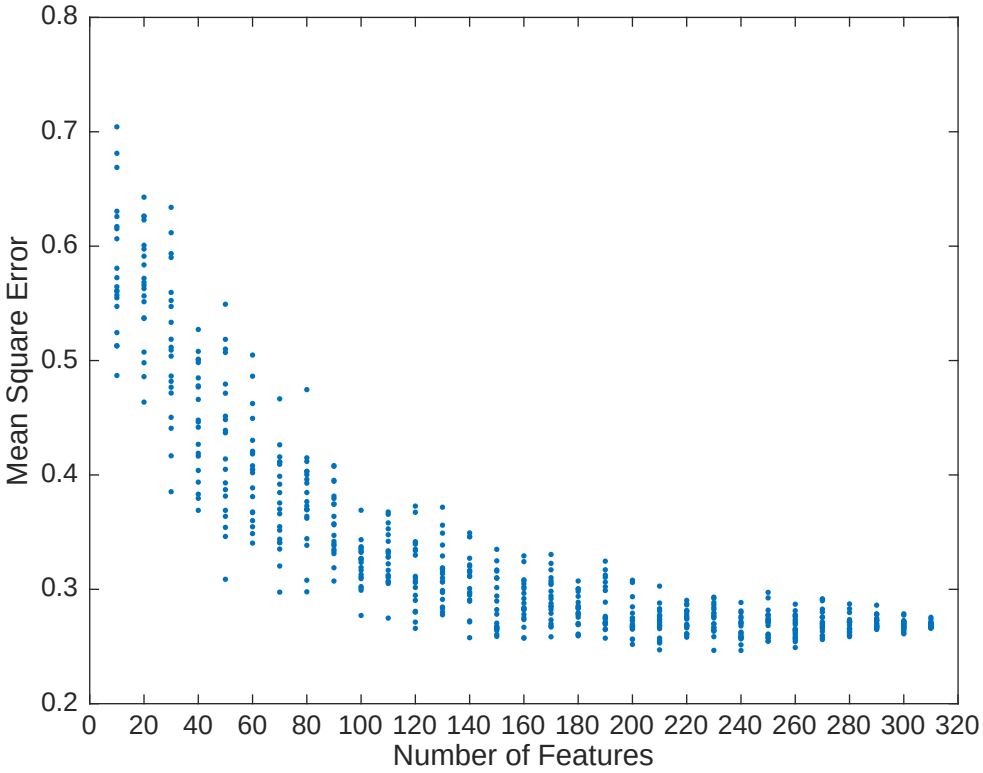
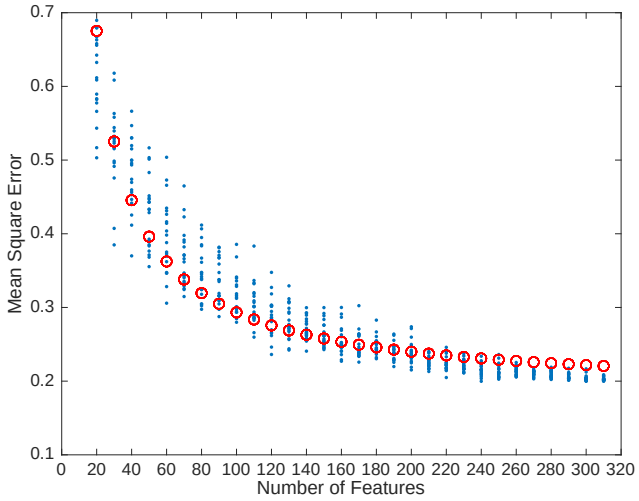


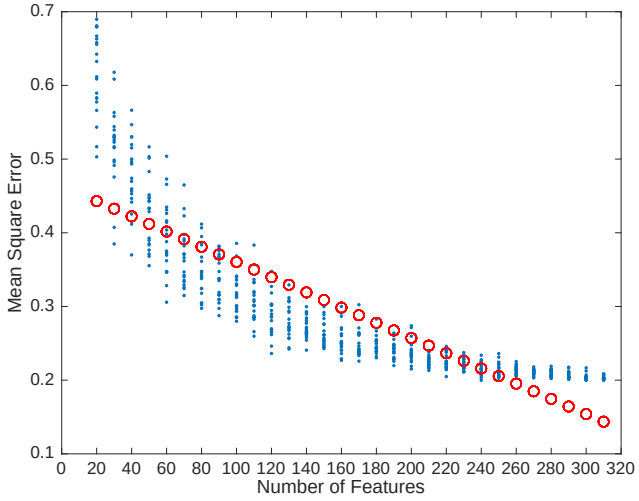
Figure 4.5: The relationship between the number of the features used for training our linear regressor and the Mean Squared Error, which represents the performance of our linear quality predictor.

are shown in Figure (4.5), which suggest the error asymptotically goes down as the number of features increases. This indicates the behaviour of our system is similar to the behaviour of the model with redundant features shown in Figure (4.3.a), rather than the linear behaviour of the model for insufficient features in Figure (4.3.b).

Next, we show that the quality assessment follows the same behaviour for non-linear predictors. For that, we [1] repeated our experiment with a neural network that had one hidden layer with five nodes using a sigmoid activation function, followed by a linear regressor. The combination



(a) Redundant feature model



(b) Subset of features model

Figure 4.6: The relationship between the number of features and the Mean Squared Error, which represents the performance of a non-linear quality predictor. The blue points represent the experimental results that verify the error is decreasing by increasing the number of features. The red circles in (a) and (b) represent the best line that fits the blue points based on two different models proposed in section 4.3.

of sigmoid activation function in the hidden layer and linear regressor as output is a very common basic choice here. As discussed in section 4.1, larger neural networks with more complexity did not improve the performance of the system and we concluded that this simple small neural network chosen here was sufficient for this task.

Figure (4.6) shows the fit of our experimental results to the two models with redundant and insufficient features. The blue points represent the experimental results. The red circles in Figure (4.6.a) and (4.6.b) are two candidate models fit to the data from redundant features and insufficient features respectively. To find the candidate models for Equations (4.25) and (4.34) that best fit to our data, we employed the fminsearch function in the Matlab 9.9.0 library. The fminsearch function finds minimum value of multivaraible functions, which here is our cost function that is defined as squared error between data points and the output from Equations (4.25) and (4.34) respectively. Based on Akaike's information criterion [212] the model with redundant features fits our data better than the model with insufficient features and its evidence ratio is 2.5×10^{31}. The evidence ratio indicates that for given data from our experimental results, the model with redundant features is 2.5×10^{31} more likely than the model with insufficient features to be the best fit.

In both experiments explained in this section, it was observed that the variance of the performance of the system decreased with increasing the value of n. This was expected as the different random X_n's are more likely to include the same features for larger n, when the overall number of the features in X is fixed [1]. Hence, the variance of the performance of the system is small for larger values of n [1].

4.6.4 Experiment with quality assessment

This section explains the final experiment with our proposed nonintrusive system using the enhanced feature set proposed in section 4.5. In the following, the effect of the proposed standardisation method is analysed, and

the effect of redundant features is explained in more detail.

The enhanced feature set proposed in section 4.5 is based on extracting the features of speech files using P.563 and ANIQUE+, which are standard methods for narrowband speech. Hence we downsampled the 16 kHz speech files in the database to 8 kHz before we could build our augmented feature set. Generating the enhanced feature set from ITU-T Supplement 23 database took 12 minutes on an Intel(R) Core(TM) i5-8265U processor.

As explained in section 4.4, the augmented feature set includes many features that have multimodal distributions. We standardised those features with the method proposed in section 4.4.2 and mapped them into features with a uniform probability distribution on the interval $[0, 1]$. The standardisation of the remaining features was not advantageous because of their smooth distribution. Thus the remaining features were mapped to the range $[0, 1]$ using min-max normalisation. Standardisation of the features extracted from ITU-T Supplement 23 database took 20 minutes on an Intel(R) Core(TM) i5-8265U processor.

We performed experiments with different regressors in our system: RVM, XNV, and neural network of different sizes. Preliminary experimental results suggested that the performance of all the systems is similar, independent of the regressor type and the neural network size. However, the high-dimensional enhanced feature set made the training of RVM and XNV slow. Although the RVM based non-intrusive system is slow and takes almost one day to complete, it is considered an excellent prediction tool where extra information about the precision of the predicted score is required. In the following, we report the results based on configuring our system to use a neural network. Like the previous experiment in section 4.6.3, the neural network has one hidden layer with five nodes using a sigmoid activation function, followed by a linear regressor [1]. Training this neural network with ITU-T Supplement 23 database takes around 15 minutes on cpu.

To evaluate the effect of our proposed enhanced feature set, we [1] per-

Table 4.1: RMSE and PCC computed per-file and per-condition for different types of feature sets. The scores are computed based on seven-fold cross-validation on ITU-T Suppl23. The per-condition PCC score is computed after applying the 3rd-order polynomial regression.

		RMSE		PCC	
Input Feature set		Per-File	Per-Cond	Per-File	Per-Cond
Min-max Normalisation	P.563	0.640	0.548	0.75	0.87
	ANIQUE+	0.632	0.529	0.73	0.87
	P.563 and ANIQUE+	0.566	0.469	0.81	0.91
Proposed standardisation	P.563	0.632	0.538	0.75	0.87
	ANIQUE+	0.616	0.510	0.75	0.89
	P.563 and ANIQUE+	0.548	0.458	0.82	0.92

formed experiments with six different feature sets:

- P.563 features that are standardised with min-max normalisation
- ANIQUE+ features that are standardised with min-max normalisation
- augmented feature set from ANIQUE+ and P.563 that are standardised with min-max normalisation
- P.563 features that are standardised with our proposed method
- ANIQUE+ features that are standardised with our proposed method
- augmented feature set from ANIQUE+ and P.563 that are standardised with our proposed method

We first analyse the effect of the augmented feature set. As illustrated in Table 4.1, combining ANIQUE+ features with P.563 features increased the performance of the system from 0.87 to 0.91 for the ITU-T Suppl23 database (see the first three rows in which the min-max normalisation is applied). Likewise, the performance is increased to 0.92, where our proposed standardisation method is applied (see the last three rows). This improvement in the performance from combining the feature sets from ANIQUE+ and P.563 is consistent with what we proposed in section 4.3, as both feature sets represent the quality of speech well and most probably contain the same information, but include independent noise. This implies that using the augmented feature set that includes features from both standards reduced the effect of input noise and improved the performance of our non-intrusive quality assessment for this database [1].

We also use Table 4.1 to evaluate the effect of our proposed standardisation method. Comparing the results in the last three rows of Table 4.1 with the first three rows indicates that our proposed standardisation method improved the performance of our system for the ITU-T Suppl23 database. Our standardisation method increased the PCC score from 0.87 to 0.89

when the ANIQUE+ features are used. However, it was not advantageous for P.563 features alone, and the PCC score stayed as 0.87 independent of the type of standardisation applied. This was expected as the number of non-smooth features in P.563 was not large, and hence, our proposed standardisation method was not beneficial for this case.

The last row in Table 4.1 illustrates that the per-condition PCC score is 0.92 when we use the enhanced feature set proposed in section 4.5 as the input to our system. In the following, we study this result in more detail and compare it with other methods in the literature followed by the statistical significance test at the end of this section.

Table 4.2 reports the details of the scores from our system using the enhanced feature set [1]. Columns one and two present the per-file and the per-condition RMS error on unseen data. Per-file and the per-condition PCC results are given in columns three and four. The score is reported for each iteration of the cross-validation procedure with the name of the database used as a test set. Given that different databases come from different laboratories and languages (BNR: English, CNET: French, CSELT: Italian, and NTT:Japanese) it was expected that the PCC value would vary from one database to another depending on the similarity between the distribution of the test and training speech signals.

Our experimental results in Table 4.2 show the mean value of the PCC score from 7-fold cross-validation on experiment one and three from supplement 23 database is 0.92. The scatter plot in Figure (4.7) visually represents the correlation coefficient with the value 0.92 between the subjective condition scores and the predicted condition scores. Each data point in the plot represents one of the test conditions from the 332 different conditions in experiment one and three. As explained in section 4.6.2, the test conditions in experiment one are related to codec distortions, and the test conditions in experiment three are the effect of channel degradation. We

[1]The authors of [1] made a mistake and reported MSE score instead of RMSE. Hence the scores in [1] are slightly smaller than the actual RMSE scores reported here.

Table 4.2: RMSE and PCC computed per-file and per-condition for our proposed system, using a combination of ANIQUE+ features with P.563. The performance measures are computed after applying a third-order polynomial regression to the model predictions . Each database in ITU-T Suppl23 corresponds to a lab and an experiment, where X1 and X3 indicate experiments one and three respectively.

	RMSE		PCC	
Database	Per-File	Per-Cond	Per-File	Per-Cond
BNR-X1	0.421	0.248	0.858	0.946
BNR-X3	0.406	0.235	0.849	0.949
CNET-X1	0.429	0.288	0.835	0.923
CNET-X3	0.431	0.303	0.807	0.893
CSELT-X3	0.529	0.391	0.806	0.884
NTT-X1	0.404	0.259	0.816	0.912
NTT-X3	0.385	0.242	0.853	0.932
Mean	0.432	0.285	0.832	0.919

computed the mean value of the PCC score from 7-fold cross-validation for each experiment individually and that is 0.93 and 0.90 for experiment one and three respectively. The reason for the lower score on experiment three is that it contains a test database in Italian language, where none of training data is Italian. If we remove the Italian database from testing, the mean value of the PCC score for experiment three increases into 0.92. This suggests that our system performance for both experiments are very similar to each other, although the distortion types are very different in these two experiments. Hence, we conclude that the performance of our system is not dependant on the type of distortions in experiments one and three.

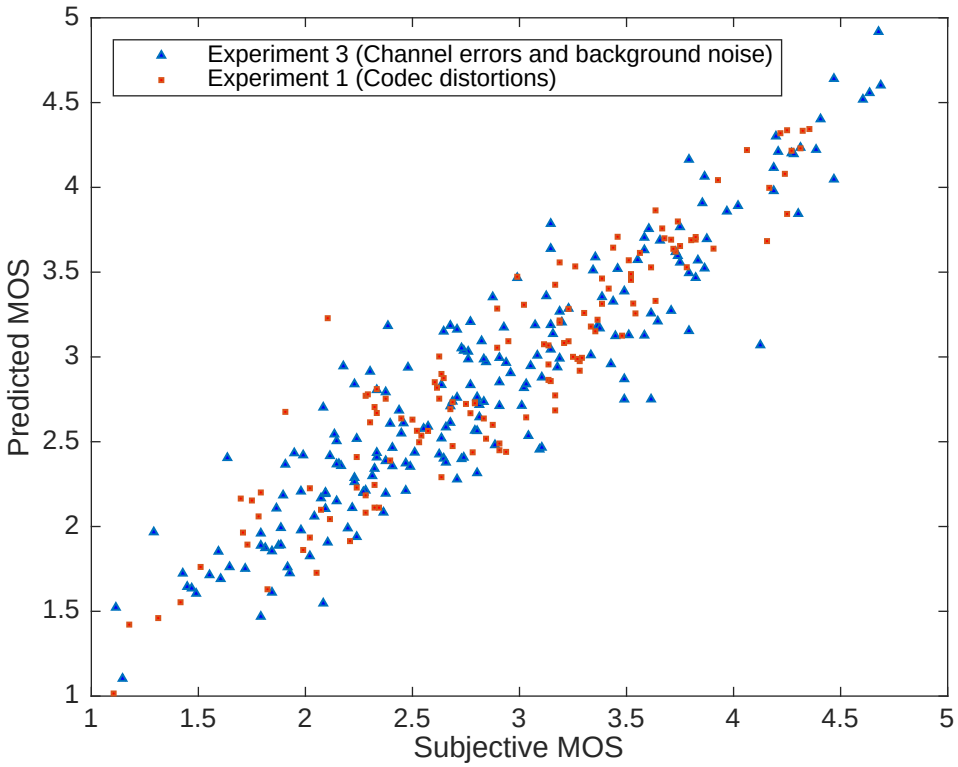


Figure 4.7: Scatter plots for condition scores on experiment one and three from ITU-T Supplement 23 databases [34]. These two experiments contain 332 conditions testing coder distortions, channel errors, and noise for English, French, Japanese, and Italian languages [8]. The results include a third-order monotonic fit between experiments.

In the following, we compare the performance of our system with other methods. We first briefly compare the performance of our system with two current ITU-T standards (for intrusive and non-intrusive methods) by simply comparing the scores as it is common practice. This forms a baseline comparison for us to start. Then we compare our results with the latest non-intrusive methods in the literature in more detail and finally provide a statistically significance test.

Table 4.3 shows the distribution of the absolute error for the ITU-T

Suppl23 database and compares our method with ITU-T standards P.563 and P.862.1. The P.563 standard is the current standard for non-intrusive quality assessment. The results in Table 4.3 indicate that our non-intrusive model is more accurate than P.563 for this database. As shown in the table, 93.67% of test signals had an error smaller than 0.5, which is higher than the 86.14% reported for P.563.

Table 4.3: Distribution of absolute errors for the Supplement 23 database after a third-order monotonic mapping (ITU-T P.862.1 is intrusive and expected to have higher performance than other methods in the table that are non-intrusive).

Absolute error	<0.25	<0.50	<0.75	<1.0	<1.25
ITU-T P.862.1	72.89%	95.18%	99.10%	100.0%	100.0%
ITU-T P.563	57.23%	86.14%	97.29%	99.70%	100.0%
Our Method	64.76%	93.67%	97.89%	99.40%	100.0%

ITU-T P.862.1 is an intrusive standard and has access to the original speech when assessing the quality of test speech. Hence P.862.1 was expected to be more accurate than our model. As reported in Table 4.3, 95.18% of test data in P.862 had an error smaller than 0.5, which was slightly higher than the 93.67% we reported. The difference between the accuracy of P.862.1 and our model became larger for a smaller error range. In P.862.1, 72.89% of estimated MOS had an error smaller than 0.25. However, in our model, only 64.76% of the estimated MOS had an error smaller than 0.25.

In the following, we evaluate the performance of our system in comparison with other methods in the literature. As it is a common practice, we focus on the mean value of PCC scores computed from 7-fold cross-validation on Suppl23 database and compare it with the results others reported in their work.

Table 4.4 reviews the PCC scores reported in the literature [14, 23, 26,

28, 61, 62] related to the non-intrusive quality assessment of speech that is built based on machine learning methods. The per-condition PCC scores are reported from performing cross-validation on ITU-T Suppl23. As explained in section 4.6.2, the standard practice is to do cross-validation by leaving one data set out in each iteration. However, the scores with † are reported from pooling all datasets together. Doing cross-validation on the mixed data is expected to result in a higher score. Other methods such as [15, 16, 24, 29, 45, 46, 58, 66, 213] are not shown in this table as they are trained and tested with databases that are either not provided for public usage or their subjective scores are not available. Thus we were unable to make a fair comparison.

As shown in Table 4.4, ANIQUE+ has a very high score of 0.98. This is to be expected as ITU-T Suppl23 was included in the training databases [215]. The authors of [14] and [28] report the next highest scores, 0.94 and 0.92 respectively. However, for both systems, additional databases were used for training and a higher score is expected. The score reported in [26] is 0.91. However, the author acknowledged an implementation error and the correct score is 0.88 [1].

Reviewing the results in Table 4.4 indicates that our model is competing with the model introduced in [62]. In the following, we provide a more detailed comparison of these competitive results for each iteration of the cross-validation and this will be statistically significance tested at the end of the section. 3

Table 4.5 compares our results with 1) the result reported by [62], 2) non-intrusive standard P.563 and 3) intrusive standard P.862.1. The reason we chose the method in [62] for this comparison is that it has the highest score in Table 4.4, where the training and test data is same as data we used in our system.

Table 4.5 indicates that the performance of our system is higher than the P.563 standard on five data sets, and it is almost equal to P.563 on the other two data sets. Intrusive standard P.862.1 has access to the original

Table 4.4: Review of the scores in the literature related to machine learning methods for assessing the quality of speech, from performing cross-validation on ITU-T Suppl23. The scores with an asterisk are from systems that used additional databases for training. The scores with † are based on experiments with different cross-validation methods. The score with $\times$ is not correct and is the result of an implementation error

Method	PCC Score
ANIQUE+ [44]	0.98*
Low Complexity, Non-Intrusive Speech Quality ... [14]	0.94*
Non-intrusive speech Quality Assessment Using ... [28]	0.92*
Our model [1]	0.92
A Hierarchical Bayesian Approach to Modeling ... [26]	$0.91^{\times}$
Probabilistic Non-Intrusive Quality Assessment ... [62]	0.91
A Bayesian Estimator for Non-intrusive Speech ... [214]	0.90†
A Bayesian Approach to Non-Intrusive Quality ... [61]	0.89
Nonintrusive Speech Quality Evaluation Using ... [59]	0.88†
A Bayesian Hierarchical Mixture of Experts [23]	0.88

Table 4.5: Comparison of PCC computed per-condition for Supplement 23 database after a third-order monotonic mapping. Each database in ITU-T Suppl23 corresponds to a lab and an experiment, where X1 and X3 indicate experiments one and three, respectively. ITU-T P.862.1 is intrusive and expected to have higher performance than other methods in the table that are non-intrusive.

Database	P.862.1	P.563	method in [62]	Our method
BNR-X1 (English)	0.968	0.902	0.926	0.949
BNR-X3 (English)	0.934	0.916	0.944	0.949
CNET-X1 (French)	0.947	0.885	0.912	0.923
CNET-X3 (French)	0.904	0.886	0.888	0.886
CSELT-X3 (Italian)	0.964	0.854	0.847	0.884
NTT-X1 (Japanese)	0.957	0.842	0.940	0.914
NTT-X3 (Japanese)	0.943	0.929	0.887	0.927
Mean	0.945	0.888	0.906	0.919

signal when scoring the test signal and its performance is naturally higher than our model on six data sets. P.862.1 reported a PCC score of 0.934 on the remaining dataset, which is slightly lower than the PCC score, 0.949 in our model.

As illustrated in Table 4.5, the result from our model is very competitive with the results in [62]. Analysing the PCC score for each iteration (i.e, each row in the table) indicates that our performance is on average 0.0126 higher in correlation than the performance of the method in [62]. The mean PCC in our method is 0.919, which is 0.013 higher than the mean PCC, 0.906, reported in [62]. In the following, we analyse the reliability of our results and statistically compare them with the results from [62].

In order to measure the reliability of the experimental results, it is a common practice to compute the confidence interval. The confidence interval is a measure of the belief that an unknown parameter value lies in a specific interval. Most commonly, the 95% confidence level is used [216]. 95% confidence is an estimated interval that we have 95% confidence the unknown parameter value lies in that interval.

For a random variable X with a normal distribution, 0.95% confidence interval is

$$\bar{X} \pm 1.96\frac{s}{\sqrt{n}}, \tag{4.41}$$

where n is the number of observed samples, $\bar{X}$ is the sample mean, and s is the sample standard deviation. However, the distribution of PCC in our experiments is not symmetric and is negatively skewed. The reason for the skew is that correlation cannot be greater than 1.0, and the distribution does not extend in the positive direction as it does in the negative direction. In the following, we explain how we computed the 0.95% confidence interval for our experimental results in Table 4.5.

The PCC we reported in Table 4.5 is based on our experiment and is the sample correlation that is called r. In our experiment with ITU-T Suppl23 r is 0.919, which is an estimate of the population correlation ρ. We used the sample correlation r to construct a confidence interval for the population

correlation ρ (Later in this section, we also use r to perform a hypothesis test on ρ and compare our method with [62]).

We computed the confidence interval on ρ based on the following steps from [217]:

- Let r be the sample correlation of the n points. Then the quantity

$$W = \frac{1}{2} \ln \frac{1+r}{1-r} \tag{4.42}$$

  is approximately normally distributed, with mean given by

$$\mu_W = \frac{1}{2} \ln \frac{1+\rho}{1-\rho}, \tag{4.43}$$

  and variance given by

$$\sigma_W^2 = \frac{1}{n-3}. \tag{4.44}$$

- Since W is normally distributed (with known standard deviaton $\sigma_W = \frac{1}{\sqrt{n-3}}$), equation (4.41) can be used to compute a 95% confidence interval for μ_W.

- The 95% confidence interval for ρ is obtained by inverting the equation (4.43) and mapping the interval for μ_W into the confidence interval for ρ as

$$\rho = \frac{e^{2\mu_W} - 1}{e^{2\mu_W} + 1}. \tag{4.45}$$

We computed the 95% confidence interval for the results we reported in Table 4.5 based on our experiment with ITU-T Suppl23. The 95% confidence interval is $[0.911,\ 0.942]$. This means we are 95% confident that the correlation coefficient between the subject scores and the scores predicted by our method is between 0.911 and 0.942.

Although 95% confidence interval is a common approach to check the reliability of our experiment, it does not compare our results with others and does not provide any information about how confident we are that our method performs better than others. The hypothesis is "The performance

of our method is better than the performance of the method in [62]". To determine how certain we are that this hypothesis is true, we must perform a hypothesis test. A hypothesis test produces a number between zero and one that measures the degree of certainty we may have in the truth of a hypothesis [217].

We constructed a hypothesis test to compare our method with [62]. There are two possible interpretations from comparing the results we observed in our experiment and the results reported in [62]:

- H0: $\rho_1 - \rho_2 \geq 0$.
- H1: $\rho_1 - \rho_2 < 0$.

The standard name for H0 is the null hypothesis, and H1 is called the alternate hypothesis. The interpretation in H0 is that the population correlation coefficient ρ_1 in [62] is actually greater than or equal to the population correlation coefficient ρ_1 in our method. Hence the sample correlation coefficient $r_1 = 0.0906$ (reported in [62]) is lower than sample correlation coefficient $r_2 = 0.0919$ (that we reported) only because of the possible variation from the population mean.

The interpretation in H1 is that the population correlation coefficient ρ_1 is actually less than population correlation coefficient ρ_2, and the sample correlation coefficients $r_2 = 0.0919 > r_1 = 0.0906$ represents a real difference that is expected to be seen if a new experiment is performed.

A hypothesis test assigns a quantitative measure to the plausibility of the null hypothesis, which is called P-value [217]. The P-value is a number between zero and one that measures the strength of the disagreement between observed samples and H0. When the P-value is small, the evidence against H0 is stronger.

Hypothesis tests are closely related to confidence intervals and can be performed based on the quantity of W computed in equation (4.42). We compute the P-value based on the following steps from [217]:

- We assume H0 is true and compute a null distribution under the assumption that H0 is true:

$$W_1 - W2 \sim \mathcal{N}(0, \sigma^2_{W_1} + \sigma^2_{W_2}) = \mathcal{N}(0, 0.077968118). \tag{4.46}$$

- We compute a z-score, which is a number representing how many standard deviations above or below zero $W_1 - W_2$ is

$$z = \frac{W_1 - W_2 - 0}{\sqrt{(1/(n_1 - 3) + 1/(n_2 - 3)}} = 1.00040892. \tag{4.47}$$

- From the z table, the P-value is 0.15, which is the area to the left of z under the normal curve in (4.46).

To early statisticians [218], p-value equals 0.05 is considered to be an appropriate choice for being significant evidence rejecting H0, and it has continued to be conventionally used in statistical analysis. Consequently, p-value equals to 0.1 explains that it is not unlikely for the observed data to be random. However, the point at which the rejection of H0 occurs depends largely on the degree of discrepancy and how it is interpreted by each individual [219]. This means the choice of significance level at which H0 is rejected is arbitrary and p-value that is not significant at the 0.05 level can be significant at the 0.1 level. Since the P-value for the results we reported based on our experiment with ITU-T Suppl23 is 0.15, we fail to reject H0 at the 0.05 level. Nevertheless, the P-value tells us that if H0 were true, the probability of observing the scores reported in Table 4.5 (that shows the performance of our method is higher than [62]) is only 15%.

4.7 Summary

We have proposed a new non-intrusive speech quality assessment based on a neural network and demonstrated the performance gain from the enhanced feature set. To achieve the higher performance we introduced

two novel enhancement procedures to the feature set: 1) Augmentation, 2) Standardization.

We hypothesised that an augmented feature set with redundant features reduces the effect of input noise and improves the performance of a non-intrusive speech quality assessment. We evaluated the relationship between the performance of the linear regressors and the number of the redundant features and derived equations that show the variance of the error asymptotically decreases with enlarging the feature set. Based on our experimental results with the linear and non-linear regressors, we conclude that a machine learning based non-intrusive system benefits from redundant features that represent the same information but include independent noise.

Our second hypothesis was that pre-distorted features with a smooth distribution facilitate the training of a machine learning based speech quality assessment. This was confirmed by the experimental results from applying our proposed standardisation methods on our augmented feature set. The effect of standardisation is more evident when a larger fraction of features have a non-smooth distribution.

The final experimental results with the ITU-T Supplement 23 database confirm the performance gain from the enhanced feature set. To demonstrate the proposed system performs well, we analysed the reliability of our results and statistically compared them with the current methods.

5

Unsupervised quality assessment

The brain is an excellent source of inspiration for machine learning. We postulate that the human brain has a model of auditory signals it receives and hence can rate the quality of these signals as good or bad if they are similar to, or different from that model postulated in the brain. In this chapter, we use generative models to simulate the existing models in the brain. We define a new criterion for quality and aim to mimic the high-dimensional functionality in the brain, which enables people to rate the quality of speech.

5.1 Introduction

In conventional non-intrusive algorithms, the specialists in the field use their knowledge to design complex algorithms to model the interaction of the features and their contribution to the overall quality [1]. Machine learning methods do not explicitly design a model, and the system relies on statistical learning from the training data. Both types use the knowl-

edge of experts to either design the system or carefully craft the features required for training. The quality assessment systems introduced in [37, 38, 39] use unsupervised feature learning, so they rely less on hand-crafted features designed based on prior knowledge. However, data used for training the regressor at the end is still to be labelled by the subjects. In this chapter, we explain how we built an entirely unsupervised quality assessment system, which does not need supervision of any kind.

Non-intrusive quality assessment methods in the literature imitate the behaviour of subjects who rate the quality of speech by learning statistics from the training data. Such systems perform well on speech corrupted under known conditions. However, they fail to reflect other types of impairment if they are not included in the training data. The quality degradation caused by the WaveNet coder [220] is an example where the quality assessment methods existing at the time failed to identify in their evaluation process and hence were not reflected in the quality score. Hence a reliable quality assessment method is desirable which is not limited to specific types of corruption or training data. To implement such a method, we look at the factors that an individual implicitly takes into account when rating the quality of speech. These factors are likely based on how far the samples are from their expectation or how much they are used to those types of corruptions.

To simulate the model existing in the human brain for the perception of good quality speech, we adopt generative models and train them with speech files that sound reasonable and natural and that humans are typically used to hearing. Then for measuring the quality of speech, we apply a new criterion that is based on divergence metrics explained in section 3.5.

The remainder of this chapter is organised as below. In section 5.2, we review the literature that inspired the approach developed in this chapter and compare our proposed approach with the methods in the literature that adopt a similar insight.

In section 5.3, we state the problem we solved in this chapter by studying generative models from a different perspective, and we show how they can be employed for measuring the quality of speech. We start with a generator G that is trained to map the latent variable Z to sample X_1 from $P_1(X_1)$. We note that using G to generate data X_2 from $P_2(X_2)$ effects the distribution of Z, unless $P_2(X_2) = P_1(X_1)$. We analyse this hypothesis in section 5.4 in more detail and explain how to benefit from this in measuring the quality of samples that are outside the training domain.

In section 5.5, we explain our proposed method and implement an inverted generator that projects sample X to its latent variable Z. In section 5.6, we perform experiments to assess the proposed idea and evaluate the high-level performance of the proposed quality assessment system. We observed that the new proposed criterion positively correlates with the objective scores estimated by PESQ. This correlation is promising for investment on this novel measure.

The focus of this work is mainly on *audio* quality estimation. However, the theory and experiment results can accurately be extended to other domains, such as the domain of the *image*. In section 5.7, we summarise the overall findings in this chapter and discuss the limitations and shortcomings.

5.2 Related works

Many successful machine learning models have been implemented with the intention to copy the schemes observed in the biological brain. For example, deep-learning models attempt to mimic the activity in layers of neurons in the neocortex [221]. Convolutional neural network models [129] that are successfully used for pattern recognition tasks are another example that are heavily inspired by the study of cats' brains in the 1950s and 1960s by Hubel and Wiesel [222, 223]. In their research, they explored how neurons in the brain are organised to produce visual perception and

suggested a new model that specifies how mammals perceive the world visually. In the following, we briefly review the findings with regards to vision and study how the concept presented there can be extended into speech and employed in quality assessment. Then we highlight the key differences between our proposed method and the other methods in the literature that are based on a similar concept.

Hubel and Wiesel [222, 223] verified that although vision starts from the eyes, the actual interpretation of visual inputs is in the primary visual cortex in the brain. Their experiments [224] showed that if a kitten is prevented from having a visual experience during a critical period at the start of its life, its vision will be severely affected for the rest of its lifetime. They showed that the effect was less when the prevention of visual experience and the eye closure was delayed. Furthermore, they confirmed that there were no effects on the vision from the closure of eyes in an adult cat [225]. These experimental results hold that learning the model of vision in the brain has to occur during a critical period in infancy. It was shown that the same phenomenon also exists in primates [226].

The critical period hypothesis was also popularised in linguistics by Lenneberg in 1967 [227]. This hypothesis holds that primary language acquisition must occur in childhood before cerebral lateralisation is complete. The requirement for hearing and practising during a critical period is apparent in the studies of language acquisition in feral children who have minimal exposure to language, and likewise in the studies that involve congenitally deaf children [228]. Research on the brain regarding second language acquisition [229, 230] also illustrates that variation in age of exposure to second languages results in different neural representations.

Other interesting findings that are closely related to the critical period hypothesis are the studies associated with the phonetic structure of language. The structure of the phonetic sounds that people hear during their early life shapes both their perception and production of speech [228].

Studies showed that very young infants do not have any bias towards the phonemes characteristic of any particular language. Hence young infants perceive the speech sounds universally similar to each other. However, this will change when they grow. For example, adult Japanese speakers cannot reliably distinguish between the /r/ and /l/ sounds in English [231], whereas 4-month-old Japanese infants can make this discrimination as reliably as 4-month-old English infants. From this insight, one possibility is that adults' responses to the stimulation of the brain is based on the circuits they retain from the input they are exposed to during childhood [228].

Based on these findings, we propose that a person rates the quality of input signals based on the pre-existing structure in their brain that is formed during their childhood. Quality does not have an explicit definition. However, an individual implicitly rates the quality, where high quality for them corresponds to normal and typically formed based on their listening habits. In other words, given that an individual has been exposed to a variety of speech since they were born, they have a model of good quality speech in their heads, which is linked to the speech that they typically hear. In this study, we adopt this hypothesis, and as will be explained in section 5.3, define a new criterion for measuring the quality by employing generative models. We now provide a brief overview of the fundamental differences between our proposed system and the existing non-intrusive quality assessment systems in the literature. Then we review methods in the literature that seem to apply an approach equivalent to our proposed approach of applying deep generative networks. We further state how our work is distinct from them.

Recently, many quality estimation systems have been introduced based on machine learning methods [28, 29, 30, 31]. Our work is distinct from them in two significant aspects:

1. In this study, data used for training are standard audio signals and are not required to be labelled by human subjects. This is because

we mimic the behaviour of the human brain that is not trained with degraded signals.

2. In this study, quality has a new definition, which is based on how different the input is from the pre-existing model trained on natural inputs. Therefore, the method is not dependent on the specific types of corruptions in the training data.

Anomaly detection methods such as [232, 233, 234, 235], which try to find anomalies in the data without supervision, seem to be conceptually similar to what we propose in our quality assessment. However, our work differs from them in two key ways. The first key difference is their application. Our system is designed to rate the quality, rather than simply indicating the anomaly. As will be explained in this chapter, our system allows rating because it explores the informative knowledge about the distribution of the test samples. In contrast, the anomaly detection systems define general criteria for anomalies based only on the distribution of training data and seek no information about the distribution of test samples.

The second main difference is the approach applied in our system, which makes this rating possible. Unsupervised anomaly detection methods [236, 237] are often based on neural networks that are trained to reconstruct training samples. In these models, it is assumed that the network can reconstruct a new test sample with a small reconstruction loss only if it comes from the distribution of training data. Hence, a large reconstruction loss is an indicator of the anomalies for test samples. Such systems are optimised once for the entire training data so that they can reconstruct any new sample that is similar to training data. However, in our proposed approach, we first train our system with the training data but then at test time, we optimise the corresponding latent variable for each test sample individually. Following that, as will be explained in section 5.5, we use the latent variables of test samples to measure the difference between the distribution of test inputs and training data. Finally, we quantify this dif-

ference and adopt it as a measure of quality.

Similar to our approach, the authors of [238] suggest using deep generative models for speech quality assessment, and applied a WaveNet architecture in their work. However, they try to mimic PESQ, which is an *intrusive* algorithm. To our knowledge, no one has used the concept of deep generative learning for *non-intrusive* quality assessment.

Likewise, RankGAN [239] proposed a generative adversarial network that assesses and ranks the quality of samples. However, RankGAN is not designed for assessing the quality, and it intends to help GANs with learning a better generator. As opposed to performing a binary classification task in the original GAN, RankGAN learns by relatively ranking information. In RankGAN, the discriminator module is replaced with a ranker. The ranker is trained to rank the fake input lower than real input with respect to a reference signal. Since a reference signal is required in RankGAN, it is again an intrusive type of quality assessment.

The authors of [240] brought the concept of quality assessment into the field of generative models. However, their goal is opposite to ours. They use quality assessment methods for measuring the quality of samples generated from GANs intending to assess the quality of the GAN itself. Our work is one of the first to use GANs for quality assessment from this point of view.

In this chapter, we propose the novel idea of adopting generative models for non-intrusive quality assessment. For this, we study generative models from a different perspective. Instead of focusing on the generators for generating samples with good quality, we focus on the latent variables that are input for the generator that produces samples with varying quality. Subsequently, our contribution is to adapt a generative model and build a generic model of speech similar to the model in the brain for measuring the quality of new input based on the structure existing there. The next section outlines this statement in more detail.

5.3 Problem statement

As explained in the previous section, it is natural to think of the quality of voice as the distance between what people hear and the model existing in their brain (i.e., the model that is developed based on what they are used to hearing). Based on this notion, and how a person's opinion about the quality is affected by this factor of distance, we claim that the quality of an audio file is good for a person when it is close to their expectations built on their listening habits. Conversely, quality is deemed to be bad if it is far from their expectations. Generative models are a suitable method to be used in this context because they enable us to build a generic model of speech similar to the model that exists in the brain. In this section, we briefly state the notion of generative models for quality assessment and the relative criterion to be considered for that and leave the detailed analysis of our hypothesis and its practicality for the next section.

When generative models are trained on a dataset that is considered to be good quality data, these models learn a manifold of good quality data. The distribution of the learnt manifold is usually complex and not explicitly defined. To generate good quality data, the generator G samples the latent random variable, Z, from a simple distribution, P_Z, and maps that into the random variable, $X \sim P_X$, from the distribution of good quality data. At first sight, using the log-likelihood of the samples generated from the generators seems like a good measure to assess its quality. However, as stated in [241] log-likelihood and visual fidelity of samples seemed to be mostly independent of each other when the data is high-dimensional in deep generative models. Moreover, evaluating likelihoods is challenging when the model density is specified implicitly based on the prior density P_Z and the generator function G. Hence, in the following section, we seek to find a reliable criterion that measures quality correctly. Here we briefly analyse the log likelihood of data in generative models and briefly discuss how transforming data into the latent space effects the model density.

Let us consider $g = G^{-1}$ is a function representing the inverse of the neural network that projects back X into its latent variable Z:

$$Z = g(X). \tag{5.1}$$

For simplicity we assume g is a bijective function and the dimensionality of X and Z are identical. By applying the formula of change of variables, the probability density of the sample x transformed to z can be written as:

$$P_X(x) = P_Z(z) \left| \det \frac{\partial g(x)}{\partial x} \right|, \tag{5.2}$$

where the term $\frac{\partial g(x)}{\partial x}$ is the Jacobian of g at x. Although $P_Z(z)$ is easy to compute, the Jacobian of high-dimensional distributions can be computationally expensive [242]. In our scenario, $P_X(x)$ will remain unknown. However, the log likelihood of the test sample, $\bar{X}$, is computed as:

$$\mathbb{E}_{\bar{X}} \log P_X(\bar{X}) = \mathbb{E}_{\bar{X}} \log(P_Z(g(\bar{X}))J(\bar{X})) \tag{5.3}$$

$$= \mathbb{E}_{\bar{X}} \log(P_Z(g(\bar{X})) + \mathbb{E}_{\bar{X}} \log(J(\bar{X})), \tag{5.4}$$

where we defined $J(x) = \left| \det \frac{\partial g(x)}{\partial x} \right|$.

In Equation (5.4), the first term $\mathbb{E}_{\bar{X}} \log(P_Z(g(\bar{X}))$ is the cross entropy between the ground-truth and model distributions, and term $\mathbb{E}_{\bar{X}} \log J(\bar{X})$ provides the adjustment required for the conversion to the likelihood in the original domain. This equation illustrates that the likelihood value might be reduced or increased in the latent space depending on the Jacobian term, which accounts for the compression or expansion of the local volume in the mapping from X to Z. If we replace the log-likelihood with a general form of function $h : \mathbb{R}^N \to \mathbb{R}$, we have:

$$\mathbb{E}_X h(X) = \int P_X(x)h(x)\mathrm{d}x = \int P_Z(G^{-1}(x))J(x)h(x)\mathrm{d}x. \tag{5.5}$$

Hence, we note that other measures are also not the same when evaluated in the original domain or in the latent domain. However, if the distance measure between two distributions be zero in the original domain, evaluating them in the Z domain also would be the same.

This suggests that for comparing $\bar{X}$ with X, we should compare distributions rather than computing a likelihood or similar measures. This is because a comparison of distributions would always show the difference between random variables $\bar{X}$ and X. However, the likelihood and related measures only provide the probability of the samples given the model distribution of the training data. Consequently, they do not provide additional information about how samples are distributed and hence do not make it possible to compare other aspects of the distribution of samples with the distribution of training data.

In the following section, we propose that for measuring the quality of test sample, $\bar{X}$, we use the inverted generator, G^{-1} and map $\bar{X}$ into the latent space to find the relevant $\bar{Z}$ for that. Then we postulate that comparing the distribution of the latent variable, $\bar{Z}$, with the prior distribution, P_Z, is a good indication of the quality of $\bar{X}$. The following section analyses our proposed criteria in more detail and studies the general form of measuring divergence between the distributions for this purpose.

5.4 Analysis

We analyse the hypothesis stated in the previous section, which is the principle of the proposed unsupervised quality assessment developed in this chapter. We first analyse the utilisation of a generative model for simulating the model that hypothetically exists in the brain. Then we analyse the proposed criteria for comparing the distribution of the latent variables with the prior distribution in order to rate the quality of a new sample.

Let us assume generator G_1 acts as an operator that maps a normally distributed random vector $Z \sim \mathcal{N}(0, I)$ to another vector $X_1 \sim P_1$ as:

$$X_1 = G_1 Z. \tag{5.6}$$

Now consider a new test sample $X_2 \sim P_2$ that can be generated from an-

other operator G_2:

$$X_2 = G_2 Z. \tag{5.7}$$

We intend to compare X_2 with X_1 and rate its similarity to X_1. For this, we suggest finding the input $\bar{Z}$ for G_1 so that the output is X_2, and that is:

$$\bar{Z} = G_1^{-1} G_2 Z. \tag{5.8}$$

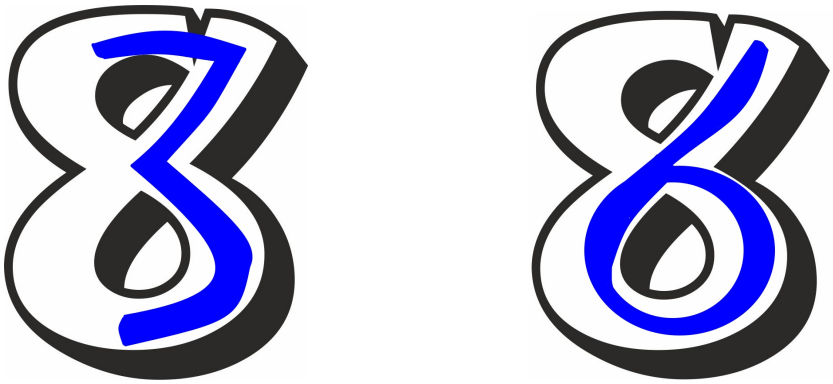
Accordingly, $G_1^{-1} G_2 = 1$ only if $G_1 = G_2$ and that results in $P_1 = P_2$. In this special case $G_1^{-1} G_2$ Maps Z to itself and hence, $\bar{Z}$ has a multivariate normal distribution with zero mean and the identity covariance matrix. In other scenarios the term $G_1^{-1} G_2$ in Equation (5.8) effects the distribution of $\bar{Z}$. Consequently, the distribution of $\bar{Z}$ will be different from $\mathcal{N}(0, I)$. We propose that divergence between the distribution of $\bar{Z}$ and Z is an indication of how much the test sample X_2 looks like the desired sample X_1. Furthermore, we propose that the operator G introduced above can be represented by generative neural networks that map the latent variable Z to sample X.

As illustrated in section 3.5, we can define a divergence between two distributions P_Z and $P_{\bar{Z}}$ in the general form of:

$$d_f(P_Z, P_{\bar{Z}}) = \mathbb{E}_{Z \sim P_{\bar{Z}}} f(Z) - \mathbb{E}_{Z \sim P_Z} f(Z), \tag{5.9}$$

where different choices of f results in different divergence metrics. For example, by choosing $f(z) = z$ or $f(z) = z^2$, Equation (5.9) measures the distance based on the mean or the variance of the distributions. IPM distance measures, such as MMD and Wasserstein distance do not explicitly define the function, and instead specify a class of functions $\mathcal{F}$. Then the supremum function f that has an average value over P_Z that is most different from its average over $P_{\bar{Z}}$, is used to measure the divergence (see section 3.5 for a more detailed explanation).

The divergence metrics used in statistics usually do not have any prior information about the distributions that are compared and therefore are



(a) Digit three. (b) Digit six.

Figure 5.1: Association of the pattern of digit eight to digit three and six.

very generic. In this work, we know that P_Z is a multivariate normal distribution with zero mean and the identity covariance matrix. Consequently, we have some intuition about the properties of the distribution that are more relevant to be compared for measuring how much that distribution diverges from $\mathcal{N}(0, I)$. In this work, we rely on this prior information and select a function f so that it reflects the properties that best represent the normal distribution.

To select a proper function and also to visually compare the distributions of variables, we made a pilot experiment in section 5.6.1. This pilot experiment represents the example latent variables in the two-dimensional space, which potentially leads us to possible choices for f. Here we only provide a brief overview of this pilot experiment and explain how we use it to verify our hypothesis. The detailed explanation of the experiment and analysis of the results are presented in section 5.6.1.

The pilot experiment employs the images of handwritten digits from the MNIST database (refer to section 5.6.1 for the description of the MNIST database) and investigates the earlier statement about rating the similarity of new samples with the desired samples by comparing their distributions in the latent space. As shown in Figure (5.1), by their nature the patterns

and curves in the shape of digits three and six are to some extent related to the pattern in the shape of digit eight. As will be explained in section 5.6.1, other digits can be more or less related to the digit eight too. Accordingly, we assume that the distribution of the image of the digit eight is the desired distribution, and we intend to compare the distributions of the images of other digits (including digit eight itself) with the desired distribution. Consequently, we replace G_1 in Equation (5.6) with a GAN that is trained with the images of the digit eight. G_1 maps $Z \sim \mathcal{N}(0, I)$ to X_1, which is a random variable with the dimensionality of 28×28 that represents the greyscale images of handwritten digit eight.

After training the GAN, we implement an inverted generator G^{-1}, which maps greyscale images of any new handwritten digit, $\bar{X}$, to its relevant 2-D latent variable $\bar{Z}$. We perform four tests, in which, $\bar{X}$ represents images of digits eight, three, six, and one respectively. Figure (5.6) in section 5.6 presents the two-dimensional latent space and visually compares the distribution of the latent variables, $\bar{Z}$, retrieved from these tests for the digits eight, three, six, and one.

Figure (5.6.a) presents the two-dimensional latent space relevant to the digit eight. This figure illustrates that the elements of the latent variable for generating the digit eight are approximately distributed as expected and it is comparable to a multivariate Gaussian distribution with zero mean and the identity covariance matrix. Furthermore, Figure (5.6.b) presents the two-dimensional latent space that visually compares the elements of the latent variables of the digits three and eight. This figure illustrates that the elements of the latent variable for the digit three are directional in the manifold. This again validates our hypothesis and also suggests that the covariance matrix or the correlation coefficient in the latent space might be a good choice for measuring the dissimilarity of the digit eight with other digits.

Ultimately, based on the entire results discussed in Section 5.6.1, we hypothesise that a criterion based on the correlation coefficients of the latent

variables is likely to be a relevant measure for quality assessment. In the following, we explain how to apply this criterion for quality assessment and later evaluate that in section 5.6.2.

For speech quality assessment, which is the focus of this thesis, we compare the test signal, $\bar{X}$, with the good quality training signal, X, based on the divergence between the distribution of their respective latent variables in the latent space. Hence, we utilise Equation (5.9) and rate the quality of $\bar{X}$ relative to X as:

$$R_X(\bar{X}) = d_f(P_Z, P_{\bar{Z}}) \tag{5.10}$$

$$= \mathbb{E}_{Z\sim P_{\bar{Z}}} f(Z) - \mathbb{E}_{Z\sim P_Z} f(Z), \tag{5.11}$$

where P_Z and $P_{\bar{Z}}$ are the distribution of the latent variables computed for X and $\bar{X}$ respectively. P_Z is a prior distribution defined in the setting of the GAN before it is trained with good quality speech. Since P_Z is pre-defined, the second term, $\mathbb{E}_{Z\sim P_Z} f(Z)$, in (5.11) is fixed. We remove the fixed term from the relative measure in 5.11 and define the absolute quality rating of test signal $\bar{X}$ as:

$$R(\bar{X}) = \mathbb{E}_{Z\sim P_{\bar{Z}}} f(Z). \tag{5.12}$$

In this new measure, the offset that is introduced into our performance measure is ignored.

The choice of $f(Z) = \log(P_{\bar{Z}}(Z))$ results in a single measure that rates the quality based on the log-likelihood of the random variables in the latent space. However, as discussed in the previous section, log-likelihood and related measures do not compare the distributions and hence are not desirable in our application. We hypothesize that $f(Z) = \|ZZ^T\|$ is a good choice as it provides a rating measure, R, based on the latent variable's covariance matrix and provides the desirable additional information regarding the distribution of the samples and its difference with the multivariate normal distribution. In order to compare the covariance matrices and assign a single value to them, we associated the covariance matrix to a scalar score by applying a matrix norm. The matrix norm is a function,

$||.|| : K^{m \times n} \rightarrow \mathbb{R}$, that maps all matrices of size $m \times n$ in the space of $K^{m \times n}$ into a scalar value in the set of real numbers under specific conditions.

Among many types of matrix norms, the Frobenius norm is the most simple one and seems to suit our purpose. The Frobenius norm of matrix A that is defined as:

$$||A||_F = \sqrt{\sum_{i=1}^{m} \sum_{j=1}^{n} |a_{ij}|^2}. \tag{5.13}$$

is the square root of the sum of squared magnitude of all entries. The Frobenius norm of a matrix can also be described as the Euclidean norm of the vectorised version of a matrix that is formed by concatenating all of its rows or columns. Accordingly, it is likely to be a good choice for our work. Employing the Frobenius norm in Equation (5.12), we propose to rate the speech signal $\bar{X}$ and assign a score to that as:

$$r(\bar{X}) = ||\Sigma_{\bar{Z}}||_F, \tag{5.14}$$

where $\Sigma_{\bar{z}}$ is the covariance matrix of the latent variable relevant to $\bar{X}$. Equation (5.14) is the main equation for this body of work, where the larger value for r represents a more significant difference between the distribution of $\bar{X}$ and high-quality data. Hence we expect that our proposed score increases as the quality of test signals decreases. Our experimental results in section 5.6.2 validate that our proposed rating based on the score defined in Equation (5.14) is a useful criteria for unsupervised quality assessment, which measures the degradation in the speech files.

To compute $\Sigma_{\bar{Z}}$ in Equation (5.14), our system requires an inverted generator, G^{-1} that maps each sample x to its latent variable z, so that $G(z) = x$. The following section explains the inverted generator in more detail and presents the implementation aspects of that.

5.5 Implementation of Method

The criterion defined in the previous section measures the correlation of data points in the latent space. Hence our method relies on having a valid inversion process that projects a given observation in the original domain to a vector in the latent space. An inversion process, known as inference, maps a data sample, x, to a latent variable, z, so that when z is passed through the generative network, it produces the original data x.

The inference model is accurate only if it is an injective function that maps distinct elements of the original domain into distinct elements in the latent space. If the inference model is non-injective, it is possible that the distinct data points, x_1 and x_2, be mapped into the same latent variable $\bar{z}$. Passing $\bar{z}$ through the generative network generates $\bar{x}$, which cannot be identical to both x_1 and x_2.

As discussed earlier in section 3.4.2, the generative model we chose for this work has a GAN training structure. All GANs have a generator that maps data from the latent space into the space that is to be learnt. However, the original formulation does not support inverse mapping. The independently proposed Adversarially Learned Inference (ALI) [243] and Bidirectional GANs (BiGAN) [244] jointly learn a generation network and an inference network using an adversarial process. However, it is shown in [147] that the fidelity of reconstructed data samples synthesised using an ALI/BiGAN are poor. In [245], the reconstruction is improved by introducing an additional adversarial cost. However, none of these inference models is defined as injective. Furthermore, such inference models only learn the training data, and hence samples with a different distribution are not expected to be reconstructed precisely. Therefore, these models are not good candidates for our application as test samples are likely to have a different distribution from the training data.

To verify our statement above with regards to the inverted inference models such as ALI and their fitness for our purpose, we tested ALI with

Figure 5.2: Examples of digits from MNIST on the left and their reconstruction with ALI on the right side.

several sample data from the MNIST database. Figure (5.2) shows random digits from the MNIST database and their reconstruction with ALI that is trained on the digit eight. The images on the left side are the sample digits from MNIST. The images on the right are the reconstructed samples that are generated with ALI from a 100-dimensional latent space. ALI reconstructs the digit eight with a small error. However, the other digits are not reconstructed well, and all the reconstruction images are a form of a digit eight similar to that digit.

Other techniques have been proposed to invert the generator of the pre-trained GANs [246, 247] for every single input under the test. These techniques have a different training approach that instead of modifying the network weights, modifies the input. In such methods, the weights from the pre-trained generator in the GAN network are frozen, and the reconstruction error is minimised individually for each sample by modifying the input. In such methods the reconstruction is precise when the dimensionality of the latent space is sufficiently large. Although additional time is required to train the inversed generator for each test sample, the benefit is that the reconstruction error is minimised for each sample. Such methods are desirable only if the reconstruction error has to be minimal.

In Equation (5.14), we assume that the inverted generator is absolute so that when the latent variable is passed through the pre-trained generator, it reconstructs the original data points under the test with zero error. Therefore, we choose an approach similar to the techniques above, where the weights of the pre-trained generator are frozen and the desired input

Figure 5.3: Examples of digits from MNIST on the left and their reconstruction with inverted GAN on the right side.

is found from the iterative steps of backpropagation. For qualitative evaluation of the inverted GAN, in Figure (5.3) we show pairs of data points and their reconstruction from a 100-dimensional latent space. This figure illustrates that although the reconstruction is not precise in our inverted generator, it results in smaller reconstruction error in comparison with ALI in Figure (5.2).

Flow-GAN [242] introduced recently is a generative adversarial network with an invertible generator. The generator in the flow-GAN is a sequence of invertible bijective transformations. Hence, the inference model is formed by inverting those invertible generators. Employing the bijective inference model in the flow-GAN results in zero reconstruction error for all data. Additionally, the inference model in flow-GAN does not have the overhead of training the network for individual test samples. However, Flow-GANs were not available when this research was completed.

In the next section, we employ our inverted generator to estimate the latent variables so that when they are passed through the generative network, it produces data points that are close to the original data. We perform experiments to assess our proposed criterion for measuring the quality, and evaluate the high-level performance of our proposed unsupervised quality assessment system.

5.6 Experiments

We initially design a pilot experiment to validate our proposed idea and then test our system on speech files to verify how this can be used for quality assessment. The next two sections explain these experiments in more detail.

5.6.1 Pilot experiment with MNIST

We perform a pilot experiment on the MNIST database of handwritten digits to visually verify the correctness of our proposed approach. First we describe the MNIST database. Next, we explain the experiment we design to represent our idea and the motivation behind our approach. Finally, we present the experimental results and analyse how they support our hypothesis with regards to the measure we define in section 5.4 in order to apply that in quality assessment.

MNIST databse

The Modified National Institute of Standards and Technology (MNIST) database [248] is a collection of handwritten digit images designed for testing the learning techniques and pattern recognition methods on real-world data. The MNIST database is publicly available and requires minimal preprocessing and formatting. Consequently, it is used extensively in machine learning research [20, 249, 250, 251], and it has become a standard for fast-testing machine learning algorithms and techniques [252, 253].

The MNIST (modified NIST) database is constructed from NIST's Special Database 3 and Special Database 1, which contain black and white images of handwritten digits. SD-3 and SD-1 in the NIST database are the training data set, and the test data set respectively. NIST's training data set is collected from American Census Bureau employees, whereas the test data set is collected from American high school students. Therefore, SD-3

is cleaner than SD-1, and the digits there are more easily recognised. Since the distribution of the training set and the test set in the NIST database is different, the original NIST is not desirable for machine learning experiments. Accordingly, the MNIST database was developed by remixing the samples from NIST's data sets, which is more appropriate for testing machine learning techniques [248].

The training set in the MNIST database is composed of 30,000 images from SD-1 and 30,000 images from SD-3. Overall the training set contains 60,000 images that are collected from approximately 250 writers. The test set in the MNIST database is composed of 5,000 images from SD-1 and 5,000 images from SD-3. The writers of 10,000 digits in the test set are different from the writers of the digits in the training set. The images in the training set and test set are labelled by values between zero and nine, which specify what digit they are.

In addition to remixing the samples, the black and white (bilevel) images from the NIST database are normalized in the MNIST database. In the modified database, the images from the original NIST are size normalized and centred in a fixed-size image so that the centre of gravity of the intensity lies at the centre of an image with 28×28 pixels [252]. The dimensionality of the resulting image sample vectors in MNIST is $28 \times 28 = 784$. The image sample vectors contain grey levels, and the pixel values are 0 to 255, where 0 is white, and 255 is black. The greyscale images are the result of the anti-aliasing technique used by the normalization algorithm [248]. Anti-aliasing is the smoothing of edges in digital fonts or images. It blends colors in a natural-looking way and makes edges appear less jagged. Figure (5.4), presents random samples from the MNIST database.

The MNIST database in machine learning is comparable to the TIMIT database [254] in the signal processing [252]. Similar to TIMIT phone classification and recognition tasks, which have been commonly used for developing and testing speech recognition algorithms [255, 256], MNIST as discussed earlier has been used as a benchmark for testing machine learn-

Figure 5.4: Random sample digits from the training set in the MNIST database.

ing techniques, and general classification tasks [252]. Hence, similar to TIMIT, which is familiar to many speech processing researchers, MNIST is a well-known database for machine learning researchers. In the following section, we describe how we design an experiment to test the correctness of our proposed machine learning technique with the MNIST database. Later in section 5.6.2, we use the TIMIT database and the NOIZEUS database to examine the proposed machine learning based quality assessment technique with speech.

Design and description of experiment

As explained above, the reason that MNIST is popular is its simplicity and its size, which allows deep learning researchers to quickly check and prototype their algorithms [253]. In this section, we explain our experiment employing the MNIST database for testing the correctness of our proposed machine learning technique and explain the motivation for conducting this experiment.

The seven-segment display shown in Figure (5.5) is a form of an electronic display device for displaying decimal numerals. The set of seven

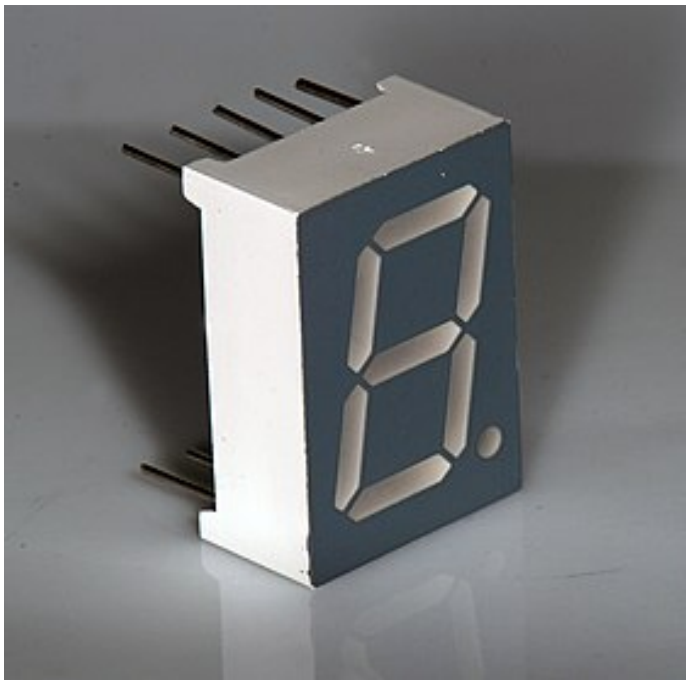


Figure 5.5: The seven-segment display from [257]

segments in these displays that form the number eight is the original motivation behind the pilot experiment we designed for checking our idea before we perform experiments with the speech files.

The seven elements of the display can be lit in different combinations to represent the numerical digits zero to nine, where digit eight is on display if all the elements are on. The fact that all the digital numbers are displayed via the digital number eight is an indication that all the digits are somewhat associated with digit eight. Considering that handwritten digits are to some extent similar to the digital display of numbers, we propose to extend this idea of an association between the digit eight and other digits into the pattern of handwritten digits. We design our experiment to measure the similarity between the digit eight and the other digits in the MNIST database and intend to quantify this similarity by applying the measure defined in section 5.4. We naturally expect to observe that digits

such as three and six, have more similarity to the digit eight than the other digits such as one or seven.

In this pilot experiment, the digit eight is the desired sample, and we intend to rate the similarity of other digits to that. In the next section, this experiment will be extended to speech quality assessment, in which speech files that have good quality are desired samples, and we intend to rate the quality of test speech files by assessing their similarity to the desired speech files. As explained in section 5.4, for rating the similarity of the samples to the desired sample, we require a generative neural network that maps the latent variables sampled from the latent space to the desired sample. Accordingly, we train a GAN with all the 600 images of digit eight from the MNIST training database.

We adopt the original formulation of GAN implemented for generating MNIST samples by Goodfellow et al. [20], which is written in Python. The neural network they used as the discriminator contains two hidden layers, each containing 240 neurons using a maxout activation function. The input size is $28 \times 28 = 784$, and the output has one neuron with the sigmoid activation, which determines whether samples are from the generator or the data distribution.

The generator neural net used by Goodfellow et al. in the GAN has two hidden layers, each containing 1200 neurons with rectifier linear activations. The output has $28 \times 28 = 784$ neurons with a sigmoid activation function. The input size in this original GAN is 100. However, we change the input size to be variable k. As will be explained at the end of this section, for the visual case, we first set k to be two, and then increase that to 100 for real comparison between the digits.

Furthermore, we train our GAN only with the images that contain the digit eight, whereas the original GAN in [20] is trained on the whole training data set that contain digits from zero to nine. Therefore, the pre-trained generator of our GAN maps the random variable $Z \sim \mathcal{N}(0, I)$ drawn from the latent space into the greyscale images of digit eight.

After training the GAN, we develop our inverted generator on top of the pre-trained generator as explained in section 5.5. The inverted generator, which is also written in Python, is an inversion process that projects the new sample image x into latent space and estimates its corresponding latent variable $\hat{z}$. The dimensionality of x is $28 \times 28 = 784$ and $\hat{z}$ is set to be k-dimensional.

To verify that our proposed statement in section 5.4 is correct, we use our inverted GAN and use the images of digits one to nine from the MNIST test data set and map them to their relevant latent variable in the latent space. We expect to see that the distribution of the latent random variable $\hat{Z}$ is similar to the prior distribution of latent random variable $Z \sim \mathcal{N}(0, I)$ for digits such as three that are similar to eight. On the other hand, we expect to see that $\hat{Z}$ diverges from $\mathcal{N}(0, I)$ when the digits under the test are more distinct. In the following, we describe the process of estimating the distribution of the latent variable for each digit.

To estimate the distribution of the latent variable for each digit $d \in [1, 9]$, we use 120 samples $\{x_{d,i}\}_{i=1}^{120}$, where $x_{d,i}$ is the i^{th} random sample from the MNIST training set that contains an image of handwritten digit d. For each of the 120 samples, we use the inverted GAN to find a k-dimensional latent variable, $\hat{z}_{d,i}$, that is relevant to $x_{d,i}$. We stack $\hat{z}_{d,i}$s and form the matrix $\bar{\mathbf{z}}_d$, in which each row contains one realisation of the random latent variable relevant to the digit d. The size of $\bar{\mathbf{z}}_d$ is $120 \times k$, and it contains 120 points in the latent space, where each point generates digit d keeping the handwriting style if used, as input to our GAN. At the end of this test, we have nine matrices that represent the estimated distribution of the latent variable for each digit from one to nine.

In order to visualise the distribution of the latent variables, we first set k to be two, which results in two-dimensional latent space. A two-dimensional latent space enables us to plot the realisations of latent variable. These plots present a high-level insight into the variation of the distribution of the latent variables of the different digits. However, two is not

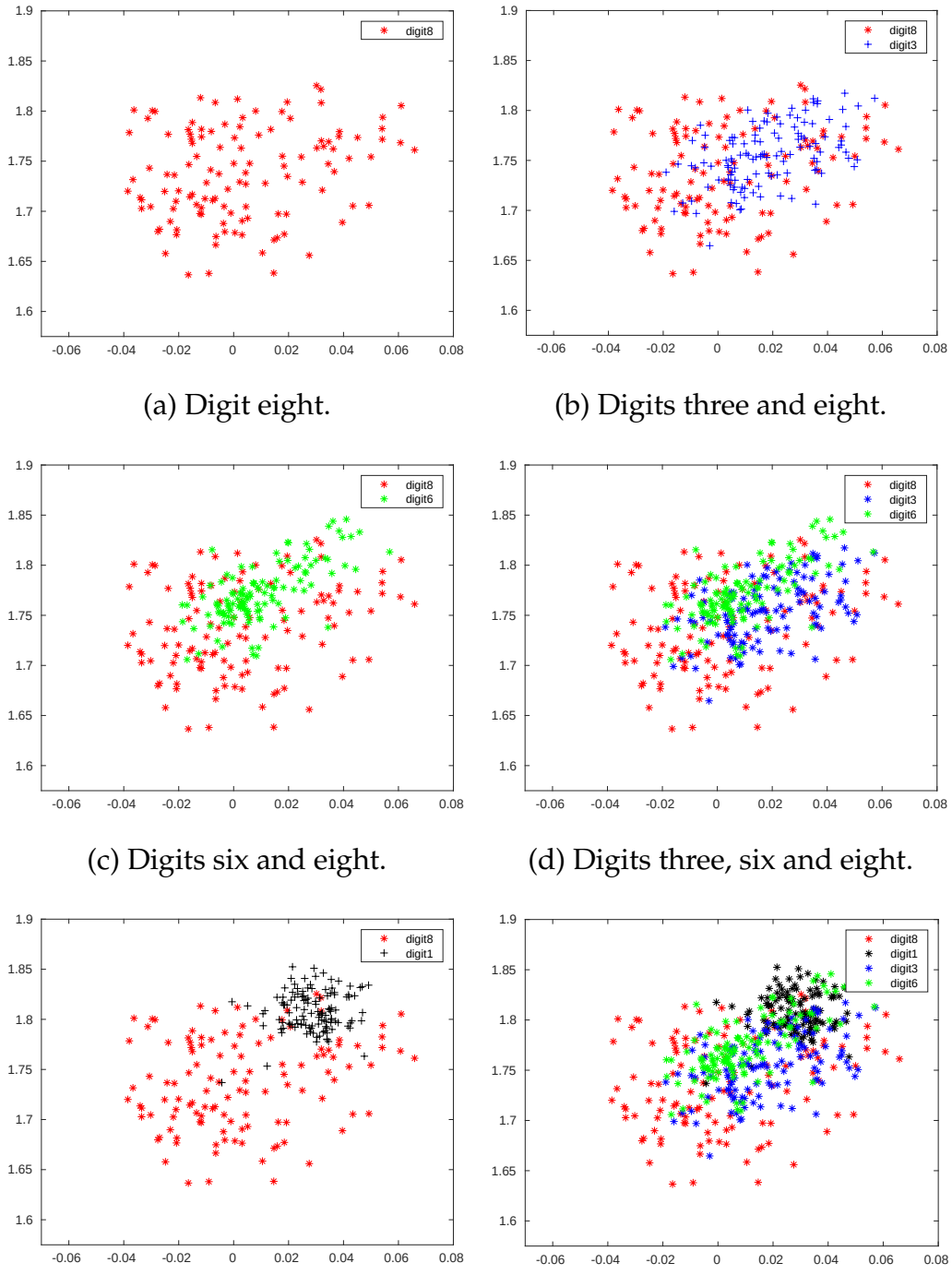
sufficient to reconstruct the digits properly with a small reconstruction error. Hence, the latent variables computed with our inverted generator are not a good estimate of the distribution of the latent variables. Moreover, the dimensionality of two is likely not adequate to represent the variation in the distributions. Hence, we repeat the test and increase the value of k to 100 to analyse the distribution of the latent variables of digits and compare them with each other in more detail. The next section presents the experimental results and their relevant analysis.

Results

Figure (5.6.a) shows the elements of latent variables found for 120 samples of digit eights from the MNIST test data set for $k = 2$. Since the GAN is trained to map $Z \sim \mathcal{N}(0, I)$ to the images of the digit eight in the MNIST training set, we expect the latent variable retrieved for the images of the digit eight in the test set to have a distribution that is close to a multivariate Gaussian distribution with a zero mean and a diagonal covariance matrix. Figure (5.6.a) illustrates that the elements of the latent variable for 120 test samples of the digit eight, which are retrieved from our inverted generator, are approximately distributed as expected. It is noted that the mean value of the latent variable is not zero on one axis. However, it is relative to a multivariate Gaussian distribution with an identity covariance matrix.

Figure (5.6.b), and (5.6.c) compares elements of the latent variable found for test images of the digit eight with elements of the latent variable found for 120 samples from the digits three, and six respectively. As expected, the latent variables for the digits three and six are directed in the manifold. Figure (5.6.d) intends to compare the distribution of the latent variable of the digit three with the digit six. We infer that the correlation between x-axis and y-axis for the digit six is larger than is the digit three as it is less similar to the digit eight.

We repeat the test above for 120 samples from the MNIST database that



(a) Digit eight.

(b) Digits three and eight.

(c) Digits six and eight.

(d) Digits three, six and eight.

(e) Digits one and eight.

(f) Digits one,three, six and eight.

Figure 5.6: Comparison of two-dimensional latent space for the digits. The latent space is based on the reconstruction of 120 handwritten samples from the MNIST database. The system is trained based on the latent space that has a Gaussian distribution with a unit covariance matrix.

contain the images of the digit one. Since the dimensionality of two is not sufficient for generating the digit one with a GAN that is trained with the digit eight, the elements of latent variable computed for the digit one are not a good estimation of its distribution. This poor estimation is the result of the large reconstruction error produced by the inverted generator, and the large reconstruction error is the result of low dimensionality of the latent space.

Figure (5.6.e) compares the elements of the latent variable found for test images of the digit eight with the elements of the latent variable found for 120 test samples from the images of digit one. Since the pattern in the model of the digit one is very different from the pattern in the model for the digit eight, the difference between the two distributions is more apparent, where both the mean and the covariance are different. As shown in Figure (5.6.f) the elements of the latent variable for the digit one has a very different distribution from the digit eight, whereas the elements of the latent variables for the digits three and six are more similar to the digit eight.

The distinct distribution of the latent variable for the digit one is also partly caused by the large reconstruction error from the inverted generator that has an input size of two. Hence, these initial plots are simply a way for visualising the high-level effect of similarity of the samples on the correlation between their relevant latent variables and are not particularly reliable.

To continue with our pilot experiment, but to achieve a more reliable estimation of the distribution of the latent variables, we enlarge the dimensionality of the latent space and set $k = 100$, which is the size of the input for the orignal GAN implemented in [20] for the MNIST databse. We repeat the test and compute $\{\hat{\mathbf{z}}_d\}_{d=1}^{9}$, where $\hat{\mathbf{z}}_d$ is a 120×100 matrix that represents the distribution of the latent variable for digits $d \in [1, 9]$.

We test the criterion we defined in section 5.4, and rate the similarity of each digit to the digit eight, using the measure defined in equation (5.14).

Table 5.1: Rating the similarity of digits to digit eight based on Equation (5.14). The rating is based on the examination of 120 handwritten samples from the MNIST database for each of the digits.

digit	8	3	5	6	2	9	7	4	1
R	.166	.167	.170	.171	.172	.179	.180	.182	.225

Table 5.1 shows the rating of the digits according to their similarities to the digit eight. The digits are ordered based on our proposed measure in increasing order. Hence, the digits on the left side of the table have the lowest value (i.e., are more similar to the digit eight), and the digits on the right side have the highest value (i.e., are less similar to the digit eight). By looking at the order of the digits in the table, we interpret that digits three and five are the most similar digits to digit eight, whereas digits four and one have the least similarity.

Since the shape of digits six and nine are rotationally symmetric, we expected them to have a similar rating. However, the results in Table 5.1 show that the digit six has a lower correlation than is the digit nine, which means the digit six is more similar to the digit eight compared to the digit nine. By looking at the shape of digits six and nine in Figure (5.7), we suggest that this disagreement with our expectation is probably because digit six is more likely to have a diagonal rather than vertical line, which creates similarity to the diagonal line in digit eight. On the other hand, as shown in Figure (5.7), the descending line on digit nine can be diagonal or vertical depending on the style of the handwriting.

Ultimately, from this reasonable rating in this pilot experiment, we infer that our proposed criterion and technique for measuring the similarity of the samples are reasonable, and we propose to employ this measure for quality assessment. The next section presents our experiments with audio files in this context.

Figure 5.7: Random samples from the MNIST database to display the vertical and diagonal lines in digits six, nine and eight. Some examples of the diagonal and vertical lines are marked in green and red in order to highlight the similarity and dissimilarity of those lines in digits six and nine to the diagonal line in digit eight.

5.6.2 Experiment with speech

In the previous section, we designed a pilot experiment to validate the idea behind our proposed unsupervised learning technique for assessing the quality of speech. We tested our technique with the MNIST database by training a generator on the desired sample data and applying an inverted generator for rating the similarity of new samples to the desired samples based on Equation (5.14). The results from the pilot experiment lead us to apply our technique to the real-world problem of quality assessment. In this experiment, we limit the degradation to the case of additive noise and examine how our proposed technique is beneficial in the field of non-intrusive speech quality assessment. This initial step towards implementing the first unsupervised quality assessment is promising and opens a new path for applying unsupervised learning into speech quality assessment.

To verify the application of our technique in non-intrusive speech quality assessment, we test our proposed system with the speech files. First we define the databases we use in this experiment and the pre-processing of

the speech files. Next, we provide a description of the experiment. Finally, we present the results and analyse them.

Databases

In this experiment, we use the speech files from two databases, TIMIT [254], and NOIZEUS [258]. Speech files in the TIMIT database are similar to how people are typically used to hearing speech. Consequently, as discussed in section 2.2, they are considered as being of good quality in this work. On the other hand, the NOIZEUS database contains noisy speech files and we use them for testing our quality assessment system. Here we provide an overview of these two databases.

The TIMIT acoustic-phonetic continuous speech corpus [254] is designed to provide speech data for the development and evaluation of automatic speech recognition systems. The TIMIT database design is the result of a joint effort among researchers at the Massachusetts Institute of Technology (MIT), SRI International(SRI), and Texas Instruments, Inc. (TI). The corpus is composed of 2342 distinct sentences from three collections [259]:

1. 2 calibration sentences, which are provided by SRI.

2. 450 phonetically compact sentences that are hand-designed by MIT.

3. 1890 randomly selected sentences chosen by TI.

The researchers at TI conducted the recording of the sentences with 630 speakers from eight dialectical regions of the United States. 439 of the speakers (Approximately 70%) are male, and 191 of them are female (30%). Each talker reads a total of ten sentences, which are the two calibration sentences, five of the phonetically-compact sentences, and three of the randomly selected sentences. Accordingly, the TIMIT database contains a total of 6300 speech files. The speech data are digitally recorded at 20 kHz in a relatively quiet environment where the peak signal to noise ratio is 29 dB. The recording is simultaneously on a pressure-sensitive microphone and

on a Sennheiser close-talking microphone. The speech files are filtered and downsampled to 16 kHz at the National Institute for Standards and Technology (NIST). In addition to the speech files in the waveform, the TIMIT database includes their relevant time-aligned orthographic, phonetic and word transcriptions. However, these are not relevant to our work and are not used in this thesis.

We consider the speech files in TIMIT as the desired samples and use them for training our system. Consequently, we test our system with the speech files in the NOIZEUS database. The publicly available speech database NOIZEUS is a noisy speech corpus developed for facilitating the comparison of speech enhancement algorithms among research groups [260]. The NOIZEUS database contains 30 sentences produced by three male and three female speakers (5 sentences per speaker). The 30 sentences are selected from the IEEE database [261] and include all the phonemes in the American English language. The sentences are recorded in a soundproof booth with Tucker Davis Technologies (TDT) recording equipment. The recordings are originally sampled at 25 kHz and then downsampled to 8 kHz. Subsequently, the recordings are corrupted by different real-world noises over a range of signal to noise ratios (SNRs).

Noise signals are selected from the AURORA database [262], which includes recordings from different places: car, exhibition hall, restaurant, street, airport, train station, train and a crowd of people (babble). The noise signals are added to the speech signals at four SNR levels of 0 dB, 5 dB, 10 dB and 15 dB.

To consider the realistic frequency characteristics of equipment in the telecommunication area, the speech and noise signals are filtered first. The filters are the modified Intermediate Reference System (IRS) filters used in ITU-T P.862, which simulate the receiving frequency characteristics of telephone handsets. The modified IRS filter is independently applied to both the clean and noise signals and then filtered noise is artificially added to the filtered speech signals. In order to add noise to the speech signals, the

active speech level of the filtered clean speech signal is first determined. Next, a noise segment that has the same length as the speech signal is randomly selected from the noise recordings and scaled to reach the desired SNR level. Ultimately, the noise segment is added to the filtered clean speech signal.

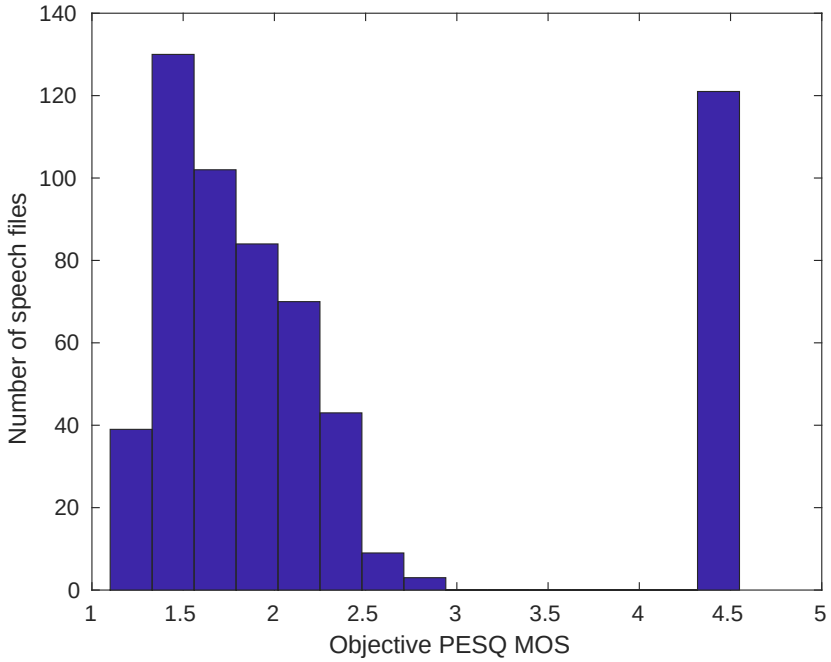


Figure 5.8: Distribution of objective MOS score computed for speech files in the NOIZEUS database by running the PESQ algoritm.

As will be explained in the description of the experiment, we use the speech files in the NOIZEUS database as input to our inverted generator and rate their quality based on the divergence of their distribution in the latent space from the prior distribution defined for our GAN that is trained with speech files in TIMIT. To evaluate our ratings and analyse their correlation with the MOS score, we run the full-reference PESQ algorithm [7] on both clean and degraded speech files.

The intrusive PESQ algorithm estimates the quality of degraded speech signals by comparing them with the clean reference speech signals. Figure (5.8) shows the distribution of MOS score computed for the NOIZEUS databsase. The PESQ score computed for noisy speech files in NOIZEUS is between one and three. For estimating the quality of clean speech files, the clean speech file is compared with itself as the reference. Since there is no difference between them, the PESQ score estimated for all of the clean speech signals is equal to 4.5.

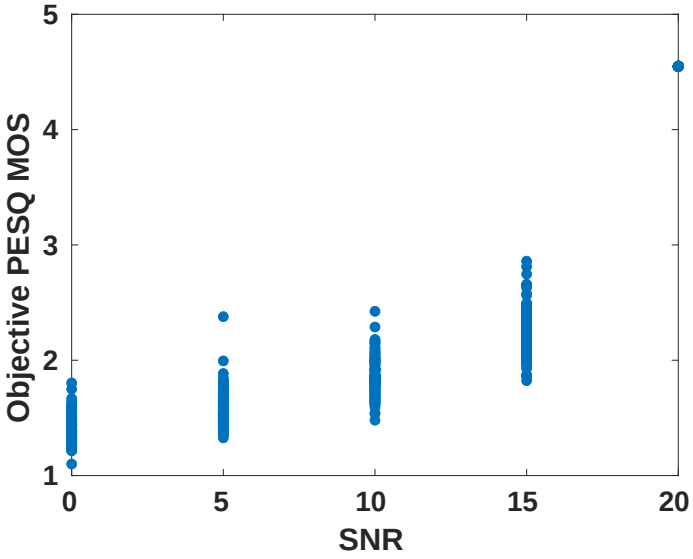


Figure 5.9: comparison between the objective MOS scores computed for speech files in the NOIZEUS database by running the PESQ algoritm and the SNR level.

Figure (5.9) shows the objective PESQ MOS scores computed for speech files in the NOIZEUS database and compares them with the SNR level of those files. As expected, the quality increases by as the SNR level increases. The scores reported in the results section is based on the PESQ objective MOS of the NOIZEUS files. In the following section, we explain how we

pre-process the speech files in the TIMIT and NOIZEUS databases to use them as input to our neural network.

Pre-processing

The speech files in the TIMIT and NOIZEUS database are represented as time series, where the y-axis denotes the amplitude of the waveform. The raw waveforms are high dimensional and may not necessarily provide explicit information about the quality of speech. On the other hand, the spectrograms of speech files, which provide a time-frequency representation of the signal, can reduce the dimensionality of raw audio and retain more information than most hand-crafted features traditionally used for audio analysis [263].

The spectrograms map raw audio data to a more structured representation that expresses essential signal properties more clearly [264]. Furthermore, spectrogram representations conveniently allow using neural network architectures that were originally designed for image processing, and image classification [265]. Hence, we choose to apply the spectrograms of the speech files from the TIMIT and NOIZEUS database as input to our neural network that is based on the GAN implementation provided in [20].

The time-frequency spectrograms are images that contain information with time and frequency along axes, and the strength of a frequency component at each time frame by the brightness or the colour. The Short-Time Fourier Transform (STFT) is a simple, easy-to-apply signal transformation method that transforms time-domain signals into time-frequency space. The spectrogram is created from the magnitude of the components in the frequency domain that are computed by STFT. We employ the Spectrogram function in the Matlab 9.9.0 library to extract the time-frequency information in the speech files of TIMIT and NOIZEUS. Here we provide details of pre-processing of the speech files in TIMIT and NOIZEUS and explain how we generate their spectrograms.

The GAN presented in [20] implemented different architectures for different databases. The architecture of the neural network we use in this experiment is based on the GAN implementation provided in [20], which is trained with the CIFAR-10 database [266]. Although we use the same architecture (i.e. same number of neurons in the hidden layers and same activation functions), we train our GAN with the TIMIT database. The GAN in [20] is trained with 50,000 images from the CIFAR-10 database. Hence, it is likely that we require tens of thousands of data points for training our GAN with the spectrograms of speech files in the TIMIT database. The TIMIT database contains a total of 6,300 sentences, a total of 5.4 hours. To increase the size of our training data into the range of tens of thousands, we split the 6,300 speech files in the TIMIT databse to smaller blocks and generate 38,880 speech segments that are half a second long. We randomly select 23,000 of the segments that contain speech for training. Similarly, we break the speech files in the NOIZEUS database into half a second blocks too. However, we select only one random block from each degraded speech file and use that one for the test. Furthermore, we downsample the speech files in the TIMIT database to 8 kHz, so that they match with the speech files in the NOIZEUS database.This preprocessing of the speech files took 20 minutes to complete on an Intel(R) Core(TM) i5-8265U processor.

We apply the short-time Fourier transform to the half-a-second-long speech signals acquired from the TIMIT and NOIZEUS. In computing a spectrogram, the STFT window size parameterises the trade-off between time and frequency resolution [267]. A large window size results in narrow-band spectrograms with high resolution in frequency, whereas small window size results in wide-band spectrograms with high resolution in time. If our machine learning model were sufficiently powerful, different choices of the spectrograms would not make a difference in the system's performance as both narrow-band and wide-band spectrograms hold the same amount of information. However, in practice, different choices might im-

pact the sensitivity of our approach for different types of degradation.

Due to the large computational demand of GAN and the limitation in the size of the input, we set the parameters to make the size of the spectrograms close to the size of image files in the CIFAR-10 database. Since the speech files are sampled at 8 kHz, the half a second speech segment contains 4000 samples. The window size is set to 128 (16 ms), which is slightly smaller than 20 ms window size that is commonly used in other speech applications. The overlap is set to 64 samples, which corresponds to the typical 50% overlap and the window function is a Hamming window. This . The number of frequency points used to calculate the discrete Fourier transforms is 128. Applying the STFT with these parameters generates spectrogram representations of the size 64×64 (frequency $\times$ time). Figure (5.10) displays random spectrograms generated from the TIMIT database. In the following section, we explain how to use these spectrograms of TIMIT and NOIZEUS for evaluating our system.

Design and description of experiment

As discussed earlier, in this experiment, we assume that the distribution of the spectrograms produced from speech files in the TIMIT database is the desired distribution for the spectrograms for their relevant speech files to be recognised as good quality speech. We intend to rate the quality of the speech files in the NOIZEUS database by projecting their relevant spectrograms back to the latent space and compare their distribution with the desired distribution. As explained in section 5.4, for this rating we require a generative neural network that maps the latent variables sampled from a simple distribution into the greyscale spectrograms with the distribution of speech files in TIMIT. Furthermore, we require an inverted generator that projects the greyscale spectrograms generated from speech files in NOIZEUS back into the latent space and estimates their relevant latent variable.

In this experiment, we set up the architecture of the GAN similar to

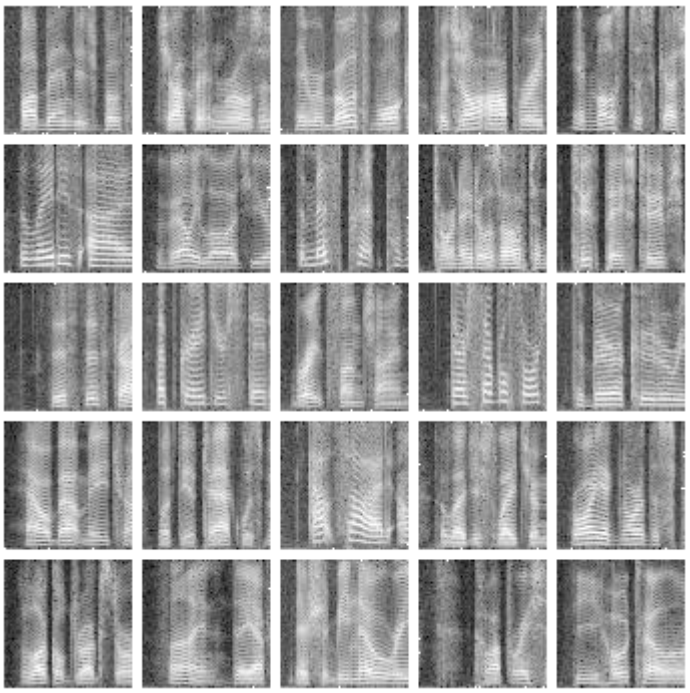


Figure 5.10: Examples of the spectrograms that are generated from a half a second of speech. The speech segments are randomly selected from the TIMIT database.

the original GAN used in [20] for the CIFAR-10 database [266]. However, instead of the CIFAR-10 database, we train our GAN with the TIMIT database and hence, adjust the architecture relatively to fit our purpose.

The generator neural net contains two hidden layers, each containing 8000 neurons with a rectifier linear and sigmoid activation function respectively. The discriminator neural net contains two hidden layers, each containing 1600 neurons using a maxout activation function. The output of the discriminator has one neuron with the sigmoid activation, which determines if samples are from the generator or the data distribution.

The CIFAR-10 database contains color images with the size of 32×32 in RGB format, where the red, green and blue pixels have integer values

from 0 to 255. Therefore the size of the generator output and discriminator input in the original GAN is $32 \times 32 \times 3 = 3072$. However, the size of the spectrograms in our work, is 64×64 and they are greyscale. To adjust the architecure of the GAN to the spectrograms in our work, we change the size of the generator output and discriminator input to $64 \times 64 \times 1 = 4096$.

We train our GAN with the 23,000 random spectrograms generated from the TIMIT speech files. Accordingly, the pre-trained generator of the GAN maps the random input $Z \sim \mathcal{N}(0, I)$ drawn from the latent space into the greyscale spectrograms with the distribution of the spectrograms of speech files in TIMIT. Training GAN on a cpu is very slow. Hence we trained our system with a gpu. Training our GAN with 23,000 training data took about 32 hours on Dual Intel Xeon processor. Figure (5.11) shows random samples generated with our GAN. These images are not real spectrograms and are not generated from a speech signal.

After training the GAN, we develop our inverted generator, G^{-1} on top of its pre-trained generator G. As explained in section 5.5, the inverted generator is an inversion process that projects the new sample image x to its corresponding latent variable z. Hence, G is expected to generate x if z is used as the input. The size of x and $\hat{z}$ is $64 \times 64 = 4096$ and 1000 respectively.

As discussed in section 5.5, the inversion process is not precise and hence G^{-1} maps x to $\hat{z}$, which is an estimate of z. Accordingly G maps $\bar{z}$ to $\bar{x}$, which is somewhat different from x. This reconstruction error is expected and does not directly impact our results as our proposed measure in the latent space does not consider the image difference in the spectrogram domain.

Figure (5.12) shows some random sample spectrograms generated from the noisy speech files in the NOIZEUS database and displays their reconstruction next to them on the right side. The spectrograms on the top two rows are generated from the noisy speech files, where SNR level is low. Therefore, those spectrograms are expected to be more noisy. In

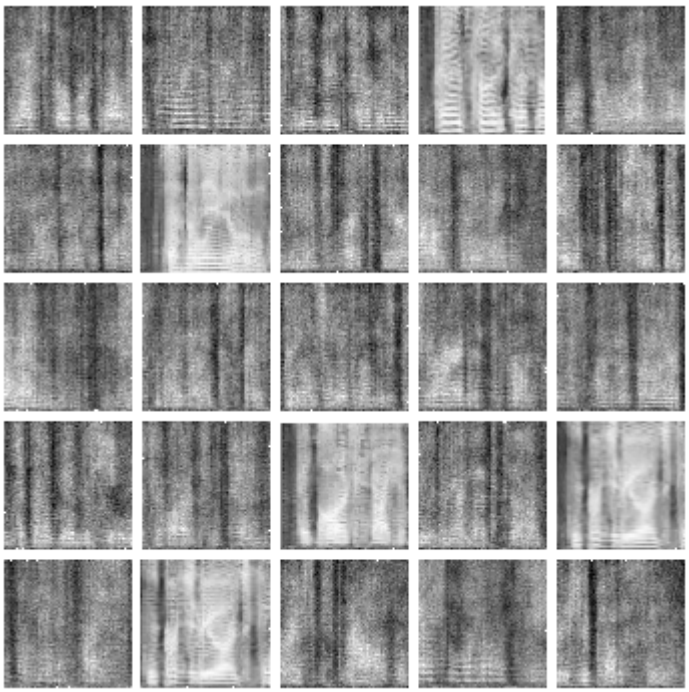


Figure 5.11: Random images generated from a GAN that is trained with the spectrograms of half a second of speech files from the TIMIT databse.

contrast, the spectrograms on the bottom rows are generated from speech files, where the SNR level is relatively high and they contain less noise.

To examine our proposed idea in section 5.4 for speech quality assessment, we use our inverted GAN and project the spectrograms of the speech files from the NOIZEUS database back into the latent space and estimate their relevant latent random variable $\hat{Z}$. We expect to see that the distribution of the estimated latent variable $\hat{Z}$ is similar to the prior distribution of the latent random variable $Z \sim \mathcal{N}(0, I)$ for the spectrograms of speech files that have small noise (and hence have large SNR). On the other hand, we expect to see that the estimated distribution of latent variable $\hat{Z}$ diverges from $\mathcal{N}(0, I)$ when the spectrograms under test are from the speech files that have a considerable noise (and hence small SNR). In the following section, we describe the process of estimating the

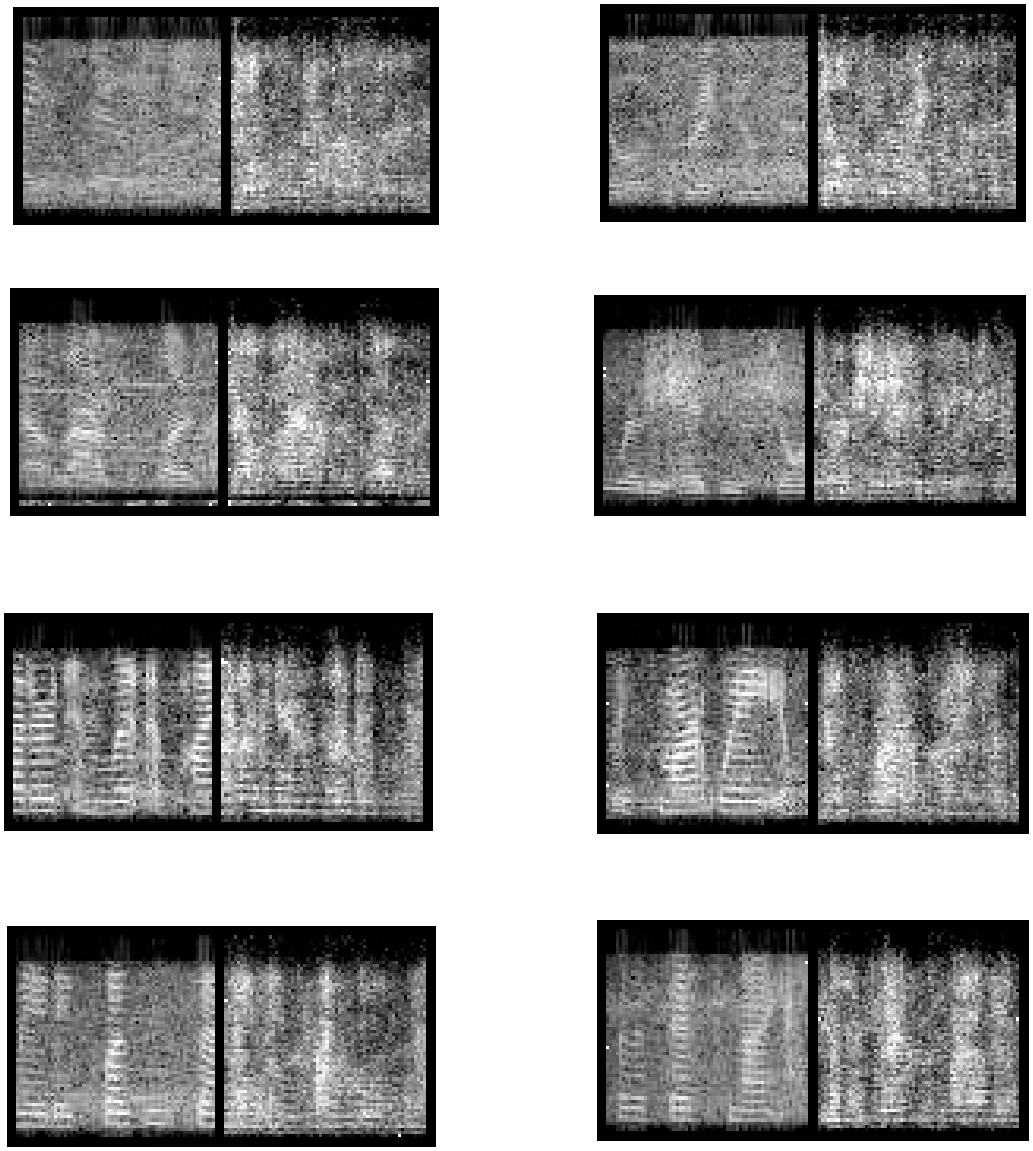


Figure 5.12: Examples of spectrograms from NOIZEUS on the left and their reconstruction with inverted GAN on the right side. The speech and noise signals in NOIZEUS are filtered by a modified Intermediate Reference System (IRS) [258]. Hence, the higher frequencies filtered by the modified IRS have a zero value and are displayed as black stripes at the top of each plot.

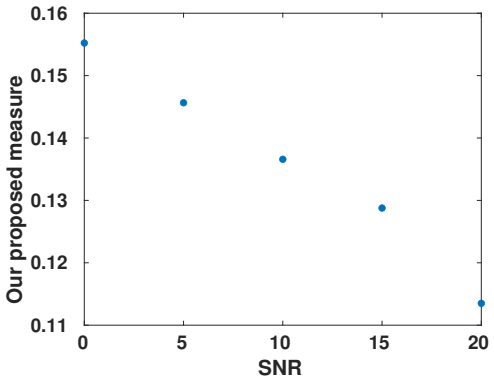


Figure 5.13: Comparison between the SNR level of noisy speech files and the Frobenius norm of the correlation coefficient of their latent variables.

distribution of the latent space for the test speech files and present the experimental results.

Setup and results

In this section, we present the experimental results and examine the correlation between the measure defined in Equation (5.14) and the SNR level and also the PESQ MOS.

We group the test files into five different conditions based on their SNR levels: 0 dB, 5 db, 10 dB, 15 dB and the clean files. We intend to rate the quality of each condition using the measure in Equation (5.14). To estimate the distribution of the latent variable for speech files that are corrupted with noise and have a SNR equal to $d \in [0, 5, 10, 15]$, we use 120 samples $\{x_{d,i}\}_{i=1}^{120}$, where $x_{d,i}$ is the spectrogram relevant to the i^{th} random speech file in the NOIZEUS database that has a SNR = d. We stack $x_{d,i}$s and form the matrix $\mathbf{x}_d$, which has the size of 120×4096. For each of the 120 spectrograms in $\mathbf{x}_d$, we train the inverted GAN on $x_{d,i}$ individually and find its relevant 100-dimensional latent variable, $\hat{z}_{d,i}$. This process takes two seconds on Dual Intel Xeon processor for each spectrogram (and hence four minutes for 120 samples).

We stack $\hat{z}_{d,i}$s and form matrix $\hat{\mathbf{z}}_d$, in which each row contains one realisation of the latent variable in the latent space of degraded speech files with the SNR value of d. The size of $\hat{\mathbf{z}}_d$ is 120×100. Accordingly, it contains 120 estimated points in the latent space that reconstructs the 120 spectrograms in $\mathbf{x}_d$ with a small error, if used as input to our GAN. We repeat this experiment for the clean files. At the end of this test, we have five matrices that represent the estimated distribution of the latent variable for the clean files and the degraded files with SNR values in $[0, 5, 10, 15]$. We use the Frobenius norm of the correlation coefficient of each matrix and rate each condition using Equation (5.14). Computing Frobenius norm of each matrix takes a millisecond by running a MATLAB code on a cpu.

Figure (5.13) determines that our proposed measure decreases as the SNR level increases. This is consistent with the analysis in Section 5.4, which states that our proposed measure is expected to decrease as the quality of test signals increases. Figure (5.13) shows a substantially linear relationship between the SNR level and our proposed measure, which suggests that it is a good measure for rating the quality.

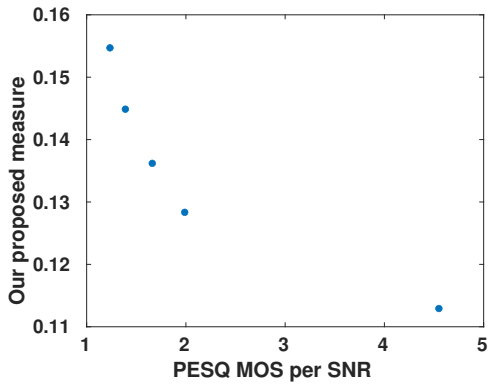


Figure 5.14: Comparison between the PESQ MOS score of speech files in NOIZEUS and the Frobenius norm of the correlation coefficient of their latent variables. The speech files are grouped into five conditions based on their SNR level.

We also study the correlation between our proposed measure and the per-condition MOS of the speech files in the NOIZEUS database. To compute per-condition MOS, we average over the PESQ MOS scores [7] of test files for each condition. Figure (5.14) shows the relationship between our proposed measure and the PESQ MOS score. Likewise, our proposed measure decreases as MOS increases. This is consistent with our hypothesis that expects a negative correlation between our measure and MOS.

In order to statistically verify the negative correlation between our measure and MOS, we randomly selected a smaller number of samples from the test signals and repeated our experiment. Figure (5.15) illustrates that negative correlation is evident despite changing the subset of samples.

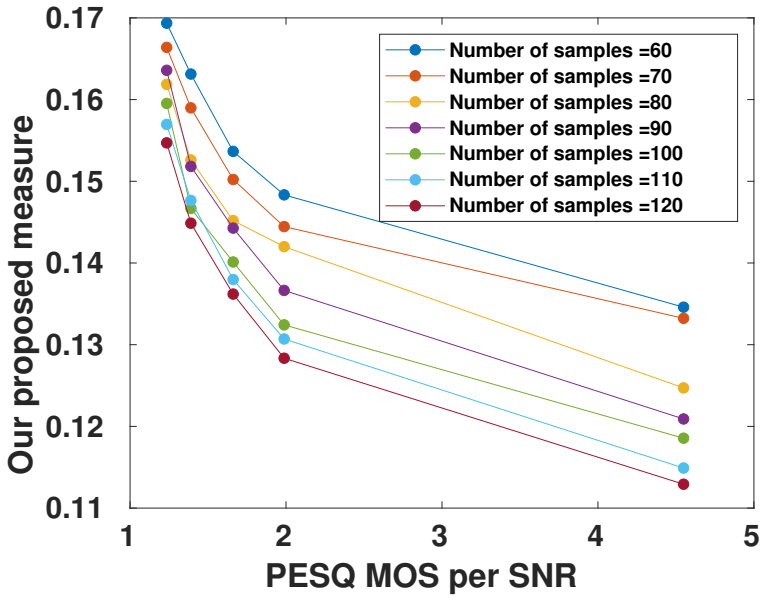


Figure 5.15: Comparison between the PESQ MOS score of speech files in NOIZEUS and the Frobenius norm of the correlation coefficient of their latent variables. The speech files are grouped into five conditions based on their SNR level. The experiment is repeated for different number of samples that are randomly chosen from original 120 test samples.

Furthermore, Figure (5.16) displays the PESQ MOS with regards to the log-likelihood of the samples projected into the latent space. Unlike the strong correlation between MOS and our measure, there is no clear relationship between MOS and the log-likelihood and log-likelihood is not as informative as our criteria when MOS score is smaller than two. This is consistent with the statements in [241, 242], where it is explained that the log-likelihood of the images generated by generator models and their relevant latent variables are not highly correlated with the quality of generated images, and therefore they are not a good measure for evaluating the generators in the generative models.

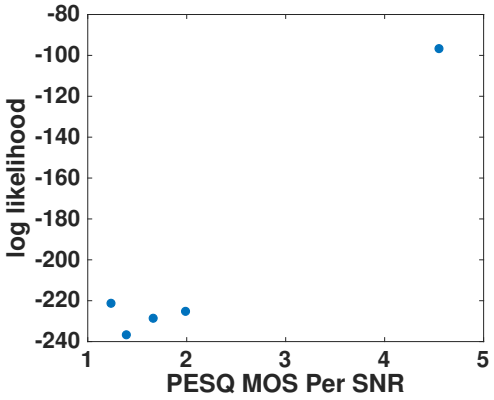


Figure 5.16: Comparison between the PESQ MOS score of speech files in NOIZEUS and the log-likelihood of their latent variables. The speech files are grouped into five conditions based on their SNR level.

In addition to the experiment with the different SNR levels, we further divide corruptions based on the different noise types: train, airport, street, and babble noise. We define 16 conditions where each condition is defined based on one of the four noise types, where its SNR level can be 0, 5, 10, or 15. The 17^{th} condition is no corruption, which includes the clean speech files. We randomly select 120 speech files from each condition and form 17 matrices, $\{\mathbf{x}_d\}_{i=1}^{17}$. The size of $\mathbf{x}_d$ is 120×4096, where each row in $\mathbf{x}_d$ contains one spectrogram relevant to a speech file randomly selected from the d^{th}

condition. For each of the 120 spectrograms in each of the 17 matrices, we train the inverted GAN individually and form 17 matrices, $\{\hat{\mathbf{z}}_d\}_{i=1}^{17}$. Each row in $\hat{\mathbf{z}}_d$ contains one realisation of the latent variable relevant to the spectrogram of one speech file from d^{th} condition. The size of $\hat{\mathbf{z}}_d$ is 120×100, and it contains 120 estimated data points in the latent space that reconstructs the spectrograms in $\mathbf{x}_d$ with a minimum error. Again, we use the Frobenius norm of the correlation coeffiecient of each matrix and rate each condition using Equation (5.14). Figure (5.17) illustrates the PESQ MOS value for each condition along the Frobenius norm of the correlation coeffiecient of each matrix. The results show a modest correlation between our measure and MOS, where the Pearson Correlation Coefficient between them is 0.9795.

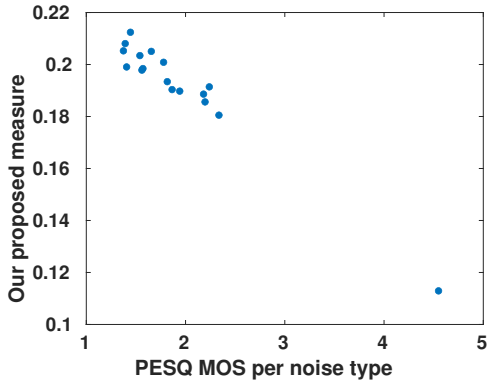


Figure 5.17: MOS score of NOIZEUS samples corrupted with the same noise type to the average correlation coefficient in the latent space.

The experiments above are designed as the proof of concept to investigate the idea of applying unsupervised machine learning into the quality assessment with the intention of removing the requirement of labelled data. The original idea in this work was to have a generic model for speech where it should not matter what representation is used for speech. However, in practice, different choices for the parameters of the spectrograms might make a difference in the system's performance. For example, in this

work we used linear frequency scale, whereas MEL spectrograms most probably make the task easier for the neural network.

Although the scoring function and the parameters of the spectrograms are not optimised in these experiments, the results from speech files that are degraded by background noise are promising. These experimental results, which show high correlation between our proposed measure and MOS, confirm that our proposed measure is valuable. Furthermore, they prove that generative models are beneficial in speech quality assessment and open doors to implementing the unsupervised methods in this field.

5.7 Summary and discussion

In this section, we summarise the GAN-based quality assessment approach proposed in this chapter. We address its shortcomings and discuss the potential solutions by applying other generative models for unsupervised speech quality assessment.

Recently many non-intrusive quality estimation systems have been introduced based on machine learning methods. However, they are all supervised or semi-supervised. The innovative GAN-based quality assessment presented in this chapter is the first attempt at an entirely unsupervised method in this context.

We introduced a novel application for GAN and built our quality assessment based on the correlation of data points in the latent space. The core ability of the GAN is learning generative models that map simple latent distributions into data. In this work, we splitted the speech files into segments to be shorter in length and used their low-resolution spectrogram images to assess their quality. Experimental results show a strong exponential relationship between their quality and the correlation of spectrograms projected back into the latent space.

Due to the large computational demand of a GAN for generating large size images, we splitted speech signals into short segments and used their

low dimensional spectrograms as input features to our GAN. However, with the improvement of deep learning, an end-to-end assessment that operates directly at the waveform level is more valuable as feature engineering approaches might not retain all the information. For example, by dividing the speech signals into short segments, we assumed that the short segment of the speech represents its overall quality. Furthermore, spectrograms do not hold all the information if they are under sampled. Moreover, using the spectrograms of the speech files causes an additional overhead to the system. The pre-processing module that divides the speech signal to the fixed time frames and computes their discrete Fourier transforms is not expensive. However, an end-to-end assessment that operates directly at the waveform level is still more valuable for the in-service systems because feature extraction will be part of the overall system. In this context, the output of the first hidden layer of the neural network is usually considered as the features and they are expected to be more informative about the quality.

A natural future line of work is to develop a speech quality assessment system that uses the raw audio waveform as input by replacing the GAN with a generative model that operates on high dimensional data. WaveNet is a generative model for raw audio and is suitable for employment in an end-to-end unsupervised non-intrusive quality assessment. We used a GAN in this work as WaveNet does not appear to have an explicit definition of latent space in the literature, whereas the basis of our proposed criteria is to measure the divergence of distributions in the latent space. WaveNet and whether it has the potential to be adapted in this application will be discussed in future works in section 6.2. In the next chapter, we discuss the contributions of the research in this thesis and provide a conclusion.

6

Conclusions

6.1 Summary

In this section, we summarise the two distinctive speech quality assessment methods developed in this thesis. We discuss the advantages of these two methods and their benefits for speech quality assessment.

The first speech quality assessment proposed in this thesis is based on supervised learning. The existing supervised learning based quality assessments usually aim to improve the estimation of the objective score of a speech utterance in a way that the scores estimated correlate with the scores obtained from human subjects. The existing methods improve on their performance mainly by using additional data. However, in this thesis, we concluded the performance of the supervised learning based speech quality assessment systems could be improved by enhancing the features that are used as the input to the regressor. In this thesis, we proposed an enhanced feature set for quality assessment based on two ideas: augmentation and standardisation:

- The novelty in *augmentation* is to enhance the feature set with raw features that are presumably redundant. We hypothesised that an augmented feature set with redundant features reduces the effect of input noise and improves the performance of a non-intrusive speech quality assessment. We evaluated the relationship between the performance of the linear regressors and the number of the redundant features and derived equations that show the variance of the error asymptotically decreases with enlarging the feature set. Based on our experimental results with the linear and non-linear regressors, we concluded that supervised-learning-based non-intrusive systems benefit from the redundant features that represent the same information but contain independent observation noise.

- With respect to *standardisation*, we concluded that pre-distorted features with smooth distribution facilitate the training of machine learning based speech quality assessment. The novelty introduced here was the pre-processing method we applied to the data, which redistributes them to achieve higher performance. The effect of standardisation is more evident when a substantial fraction of features has a non-smooth distribution.

The anticipated benefit of our novel enhanced feature set for quality assessment was confirmed with the experimental results reported in Chapter 4.

The second non-intrusive quality assessment method that we introduced in this thesis is unsupervised and is inspired by the functioning of the human brain. Our proposed unsupervised based non-intrusive assessment system is distinct from other machine learning based non-intrusive systems in two significant aspects:

- Data used for its training consists of standard speech signals and is not required to be labelled by subjects. Therefore the data required

for its training is not costly and consequently is more readily available.

- Quality has a new definition based on how different the input is from the pre-existing model trained on the natural inputs. Hence training is not based on imitating statistics of degraded speech files.

We concluded that for implementing a system that is not trained on labelled data and is not even required to be exposed to degraded speech signals, we have to simulate two functionalities similar to what exists in the human brain:

- the unsupervised functionality that learns a model of natural speech.
- the functionality that compares the observed speech with the pre-existing model, and rates the quality respectively.

To mimic the first function, we used a GAN training structure and trained that on natural speech to learn a generative model of good quality data. To mimic the second functionality, we first projected the test speech back into the latent space. Then we rated the quality based on a distance between the test signal and the distribution of good quality speech in the latent space. The experimental results show that our proposed criterion based on the distance in the latent space is highly correlated with quality scores.

While we do not claim that the proposed unsupervised system performs better than other supervised and semi-supervised methods in the literature, we believe that this primary step towards implementing the first unsupervised quality assessment. This highlights the potential of applying the generative models for quality assessment, which initiates a new era in this context.

The focus of this thesis was the *speech* quality estimation. However, the theory can be extended and applied in quality assessment for other domains, such as for images. In practice, we implicitly showed in this

thesis that our proposed method is an excellent potential image quality assessment system. This is because our experimental results are based on the spectrograms of the speech files, which form a particular set of images. The following section provides the future path of this research.

6.2 Future works

The unsupervised quality assessment proposed in this thesis operates based on the original formulation of GAN developed by Goodfellow et al. [20]. At the time this research has been conducted, an invertible GAN that can be used for inference was not established. Therefore, we implemented our own inference model that estimates the optimum latent variable by minimizing the reconstruction error. Recently, the authors of [242] introduced Flow-GAN, which consists of a pair of generator-discriminator networks with the generator specified as a normalizing flow model [268]. In a normalizing flow model, the generator transformation G is invertible, where G maps the latent variables z to the observed variables x such that $x = G(z)$ and G^{-1} exists. Using the inference model, G^{-1} results in zero reconstruction error for all data. Therefore, the unsupervised approach proposed in this thesis is expected to be more precise if GAN is replaced by the Flow-GAN.

Furthermore, as discussed in Chapter 5, an assessment that operates directly at the waveform level is desired. WaveNet is a deep generative model of audio data that operates directly at the waveform level [141]. We suggest that building a speech quality assessment that utilises WaveNet instead of GAN is a natural future path for this work.

A WaveNet that is trained on the waveforms recorded from human speakers, generates new speech utterances $x = \{x_1, ..., x_T\}$ by generating one sample at a time. At each step, WaveNet computes the probability distribution of output sample given the previous observed samples, $p(x_t|x_1, ..., x_{t-1})$. Then it generates the sample by drawing a value from

the distribution $p(x_t)$ that is computed by the network. In order to control what speech to generate, additional features are generally fed to the network. In this case, the probability distribution of each sample has the form of $p(x_t|x_1, ..., x_{t-1}, c_t)$, where c_t is a set of conditional features.

WaveNet consists of 1) a deterministic mapping from the previous sample and conditioning features to the parameters of the distribution (restricted to be of a particular family) of the next sample, and 2) the drawing of the next sample from that distribution. Naturally, the distance between the predictive distribution given by WaveNet and the distribution of the actual sample observed in the test signal is a candidate metric for rating the quality. Hence, the first step seems to be relevant for developing an unsupervised quality evaluation.

The GAN-based unsupervised quality assessment we implemented in this chapter is a proof of concept specifying how generative models can be the key to the implementation of unsupervised quality assessment. WaveNet-based quality assessment can be a future path of this research for developing an end-to-end unsupervised quality assessment system.

The unsupervised method in this thesis investigated speech that is degraded by background noise in a purely listening-only situation. We practised this setup as the small size spectrograms we used in this research seem not to be sufficient for holding all meaningful information required. However, the end-to-end quality assessment proposed above does not have this restriction and is likely to be useful when the speech quality is affected by other types of distortion due to speech codecs, packet loss, or even those types of degradation in the conversational situation such as talker echo and path delay. Naturally, for investigating these new types of corruptions, new test databases have to be created by adding those types of corruption to the clean files. In the following section, we conclude this thesis by presenting the contributions of this research.

6.3 Contributions

In this thesis, we analysed the performance gain from enlarging the dimensionality of the features in the linear machine learning models. We presented its mathematical model and verified this performance gain for speech quality assessment. The main part of this research is published in [1].

Furthermore, we defined an innovative measure for rating the quality of speech. We designed and implemented the first unsupervised quality assessment. The outcome from this research opens new doors for applying unsupervised learning into speech quality assessment.

To conclude, in this thesis, we investigated ideas to apply machine learning to quality assessment, with the focus of relieving the limitation created from the data specifications. The two quality assessment systems we proposed in this thesis establishes that machine learning has the potential to be further advantageous in this context by removing the effect that the limitation of training data has on the performance of the system.

Appendices

A

Conditions in ITU-T Supplement23

In chapter 4, we evaluated our method with the ITU-T coded-speech data set, Supplement 23 [34]. As explained in section 4.6, we employed experiments one and three from the Supplement 23 database. The following sections provide the full specifications of the conditions defined in these two experiments.

A.1 Experiment one

Table A.1 provides the full complement of the conditions for experiment one. The conditions include single encodings and combinations of the codecs. The codecs will be G.711, G.726, G.728, G.729, GSM-FR, IS-54, or Japanese Digital Cellular-HR (JDC-HR). Modulated Noise Reference Unit (MNRU) is a standalone unit for introducing controlled degradations to speech signals, where Q is the ratio of speech power to modulated noise power.

Table A.1: Allocation of conditions for experiment one (adopted from Table 5.2 in [211]).

Condition	1st codec	2nd codec	3rd codec
C1	G.729		
C2	G.729	G.729	
C3	G.729	G.729	G.729
C4	G.726		
C5	G.726	x4	
C6	G.728		
C7	G.711		
C8	GSM-FR		
C9	IS-54		
C10	JDC-HR		
C11	G.729	G.726	
C12	G.729	G.728	
C13	G.729	GSM-FR	
C14	G.729	IS-54	
C15	G.729	JDC-HR	
C16	G.726	G.729	
C17	G.728	G.729	
C18	GSM-FR	G.729	
C19	IS-54	G.729	
C20	JDC-HR	G.729	
C21	G.729	G.729	GSM-FR
C22	G.729	G.729	IS-54
C23	G.729	G.729	JDC-HR
C24	G.729	G.726	GSM-FR
C25	G.729	G.728	GSM-FR
C26	GSM-FR	G.729	G.729
C27	IS-54	G.729	G.729
C28	JDC-HR	G.729	G.729
C29	GSM-FR	G.726	G.729
C30	GSM-FR	G.728	G.729
C31	GSM-FR	IS-54	
C32	IS-54	JDC-HR	
C33	JDC-HR	GSM-FR	
C34	GSM-FR	G.729	IS-54
C35	IS-54	G.729	JDC-HR
C36	JDC-HR	G.729 GSM-FR	
C37	MNRU (Q=5dB)		
C38	MNRU (Q=10dB)		
C39	MNRU (Q=15dB)		
C40	MNRU (Q=20dB)		
C41	MNRU (Q=25dB)		
C42	MNRU (Q=30dB)		
C43	MNRU (Q=35dB)		
C44	MNRU (Q=50dB)		

A.2 Experiment three

Table A.2 presents the allocation of the conditions in experiment three, which is designed to characterize the performance of codec G.726 under detected frame erasure, and random bit error channel degradation conditions.

Table A.2: Allocation of conditions for experiment three (adopted from Table 7.2 in [211]).

Condition	Codec	Transcodings	Noise Type	Error Type	Error Rate (%)
C1	G.729	1	clean	-	-
C2	G.729	1	clean	Random Frame	3
C3	G.729	1	clean	Rando Frame	5
C4	G.729	1	clean	Bursty Frame	3
C5	G.729	1	clean	Bursty Frame	5
C6	G.729	1	Vehicle	-	-
C7	G.729	1	Vehicle	Random Frame	3
C8	G.729	1	Vehicle	Rando Frame	5
C9	G.729	1	Vehicle	Bursty Frame	3
C10	G.729	1	Vehicle	Bursty Frame	5
C11	G.729	1	Street	-	-
C12	G.729	1	Street	Random Frame	3
C13	G.729	1	Street	Rando Frame	5
C14	G.729	1	Street	Bursty Frame	3
C15	G.729	1	Street	Bursty Frame	5
C16	G.729	1	Hoth	-	-
C17	G.729	1	Hoth	Random Frame	3
C18	G.729	1	Hoth	Rando Frame	5
C19	G.729	1	Hoth	Bursty Frame	3
C20	G.729	1	Hoth	Bursty Frame	5
C21	G.729	2	clean	-	-
C22	G.729	3	clean	-	-
C23	G.729	2	clean	Random Frame	3,3
C24	G.729	3	clean	Random Frame	3, 0, 3
C25	G.729	2	clean	Bursty Frame	3,3
C26	G.729	3	clean	Bursty Frame	3, 0, 3
C27	G.729	2	Vehicle	Random Frame	3,3
C28	G.729	2	Vehicle	Bursty Frame	3,3
C29	G.729	1	clean	Random Bit	1
C30	G.729	1	clean	Random Bit	3
C31	G.729	1	clean	Random Bit	5
C32	G.729	1	clean	Random Bit	10
C33	G.729	1	clean	Burst Frame/Random Bit	3,1
C34	G.729	1	clean	Burst Frame/Random Bit	3,3
3 C35	G.729	1	clean	Burst Frame/Random Bit	3,5
C36	G.729	1	clean	Burst Frame/Random Bit	3,10
C37	G.726	1	Clean	-	-
C38	G.726	1	Vehicle	-	-
C39	G.726	1	Street	-	-
C40	G.726	1	Hoth	-	-
C41	MNRU (Q=10dB)	1	Clean	-	-
C42	MNRU (Q=15dB)	1	Clean	-	-
C43	MNRU (Q=20dB)	1	Clean	-	-
C44	MNRU (Q=25dB)	1	Clean	-	-
C45	MNRU (Q=30dB)	1	Clean	-	-
C46	MNRU (Q=50dB)	1	Clean	-	-
C47	Direct	1	Clean	-	-
C48	Direct	1	Vehicle	-	-
C49	Direct	1	Street	-	-
C50	Direct	1	Hoth	-	-

References

[1] M. Hakami and W. B. Kleijn, "Machine learning based non-intrusive quality estimation with an augmented feature set," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5105–5109, IEEE, 2017. Winner best student paper award (second prize).

[2] International Telecommunications Union (ITU-T), "Methods for subjective determination of transmission quality, Recommendation P.800," Online. http://www.itu.int/rec/T-REC-P.800-199608-I/en.

[3] International Telecommunications Union. ITU-R BS.1534, "Method for the subjective assessment of intermediate quality level of coding systems," *International Telecommunications Union, Geneva, Switzerland*, vol. 14, 2003.

[4] M. Karjalainen, "A new auditory model for the evaluation of sound quality of audio system," *IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, pp. 608–611, Mar. 1985.

[5] D. Klatt, "Prediction of perceived phonetic distance from critical-band spectra: A first step," *Proc. Int. Conf. Acoust., Speech, Signal Process.*, vol. 7, pp. 1278–1281, May 1982.

[6] J. G. Beerends and J. A. Stemerdink, "A perceptual speech-quality measure based on psychoacoustic sound representation," *J. Audio Eng. Soc.*, vol. 42, pp. 115–123, Mar. 1994.

[7] International Telecommunications Union (ITU-T), "P.862 : Perceptual evaluation of speech quality (PESQ): An objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs,Recommendation P.862," Online. http://www.itu.int/rec/T-REC-P.862-200102-I/en.

[8] L. Malfait, J. Berger, and M. Kastner, "P.563 - The ITU-T Standard for Single-Ended Speech Quality Assessment," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, pp. 1924–1934, Nov. 2006.

[9] J. Liang and R. Kubichek, "Output-based objective speech quality," *IEEE Vehicular Technology Conf.*, vol. 3, pp. 1719–1723, June 1994.

[10] D. Kim, "ANIQUE: An auditory model for single-ended speech quality estimation," *IEEE Transactions on Speech and Audio Processing*, vol. 13, pp. 821–831, Sept. 2005.

[11] P. Gray, M. P. Hollier, and R. E. Massara, "Non-intrusive speech quality assessment using vocal-tract models," *IEE Proc. Vis. Image Signal Processing*, vol. 147, pp. 493–501, 2000.

[12] International Telecommunications Union (ITU-T), "Single-ended method for objective speech quality assessment in narrow-band telephony applications,Recommendation P.563," Online. https://www.itu.int/rec/T-REC-P.Imp563/en.

[13] G. Mittag and S. Möller, "Non-intrusive speech quality assessment for super-wideband speech communication networks," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7125–7129, IEEE, 2019.

[14] V. Grancharov, D. Y. Zhao, J. Lindblom, and W. B. Kleijn, "Low Complexity, Non-Intrusive Speech Quality Assessment," *IEEE Transactions on Speech and Audio Processing*, pp. 1948–1956, Feb. 2006.

[15] B. Cauchi, K. Siedenburg, J. F. Santos, T. H. Falk, S. Doclo, and S. Goetze, "Non-intrusive speech quality prediction using modulation energies and LSTM-network," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 7, pp. 1151–1163, 2019.

[16] B. Cauchi, J. F. Santos, K. Siedenburg, T. H. Falk, P. A. Naylor, S. Doclo, and S. Goetze, "Predicting the quality of processed speech by combining modulation-based features and model trees," in *Speech Communication; 12. ITG Symposium*, pp. 1–5, VDE, 2016.

[17] J. Ooster, R. Huber, and B. T. Meyer, "Prediction of perceived speech quality using deep machine listening.," in *Interspeech*, pp. 976–980, 2018.

[18] S.-W. Fu, Y. Tsao, H.-T. Hwang, and H.-m. Wang, "Quality-net: An end-to-end non-intrusive speech quality assessment model based on BLSTM," in *INTERSPEECH*, pp. 1873–1877, 2018.

[19] K. Kavukcuoglu, P. Sermanet, Y.-l. Boureau, K. Gregor, M. Mathieu, and Y. Cun, "Learning convolutional feature hierarchies for visual recognition," in *Advances in Neural Information Processing Systems* (J. Lafferty, C. Williams, J. Shawe-Taylor, R. Zemel, and A. Culotta, eds.), vol. 23, pp. 1090–1098, Curran Associates, Inc., 2010.

[20] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, pp. 2672–2680, 2014.

[21] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[22] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *CoRR*, vol. abs/1511.06434, 2015.

[23] S. I. Mossavat, O. Amft, B. de Vries, P. Petkov, and W. B. Kleijn, "A Bayesian hierarchical mixture of experts approach to estimate speech quality," in *QoMEX 2010: Second International Workshop on Quality of Multimedia Experience*, pp. 200–205, IEEE Signal Processing Society, IEEE Signal Processing Society, 2010.

[24] M. Narwaria, W. Lin, I. V. McLoughlin, S. Emmanuel, and C. L. Tien, "Non-intrusive speech quality assessment with support vector regression," in *Advances in Multimedia Modeling* (S. Boll, Q. Tian, L. Zhang, Z. Zhang, and Y.-P. P. Chen, eds.), pp. 325–335, Berlin, Heidelberg: Springer Berlin Heidelberg, 2010.

[25] P. Petkov, H. Helgason, and W. B. Kleijn, "Feature set augmentation for enhancing the performance of a non-intrusive quality predictor," in *Quality of Multimedia Experience (QoMEX), 2012 Fourth International Workshop on*, pp. 121–126, July 2012.

[26] I. Mossavat, P. N. Petkov, W. B. Kleijn, and O. Amft, "A hierarchical Bayesian approach to modeling heterogeneity in speech quality assessment," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 1, pp. 136–146, 2012.

[27] T. Falk and W. Chan, "Nonintrusive speech quality estimation using Gaussian mixture models," *IEEE Signal Process*, vol. 13, pp. 108–111, Feb. 2006.

[28] R. K. Dubey and A. Kumar, "Non-intrusive speech quality assessment using several combinations of auditory features," *International Journal of Speech Technology*, vol. 16, no. 1, pp. 89–101, 2013.

[29] Q. Li, Y. Fang, W. Lin, and D. Thalmann, "Non-intrusive quality assessment for enhanced speech signals based on spectro-temporal features," in *Multimedia and Expo Workshops (ICMEW), 2014 IEEE International Conference on*, pp. 1–6, July 2014.

[30] D. Sharma, Y. Wang, P. A. Naylor, and M. Brookes, "A data-driven non-intrusive measure of speech quality and intelligibility," *Speech Communication*, vol. 80, pp. 84–94, 2016.

[31] M. H. Soni and H. A. Patil, "Effectiveness of ideal ratio mask for non-intrusive quality assessment of noise suppressed speech," in *Signal Processing Conference (EUSIPCO), 2017 25th European*, pp. 573–577, IEEE, 2017.

[32] G. Mittag and S. Möller, "Quality degradation diagnosis for voice networks-estimating the perceived noisiness, coloration, and discontinuity of transmitted speech.," in *INTERSPEECH*, pp. 3426–3430, 2019.

[33] M. Wältermann, K. Scholz, S. Möller, L. Huo, A. Raake, and U. Heute, "An instrumental measure for end-to-end speech transmission quality based on perceptual dimensions: Framework and realization," in *Ninth Annual Conference of the International Speech Communication Association*, 2008.

[34] International Telecommunications Union (ITU-T), "ITU-T coded-speech database." ITU-T Rec. P.Suppl. 23.

[35] N. Harte, E. Gillen, and A. Hines, "TCD-VoIP, a research database of degraded speech for assessing quality in VoIP applications," *2015 7th International Workshop on Quality of Multimedia Experience, QoMEX 2015*, 07 2015.

[36] X. Dong and D. S. Williamson, "A pyramid recurrent network for predicting crowdsourced speech-quality ratings of real-world signals," *arXiv preprint arXiv:2007.15797*, 2020.

[37] P. Ye, J. Kumar, L. Kang, and D. Doermann, "Unsupervised feature learning framework for no-reference image quality assessment," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1098–1105, June 2012.

[38] M. H. Soni and H. A. Patil, "Novel deep autoencoder features for non-intrusive speech quality assessment," in *2016 24th European Signal Processing Conference (EUSIPCO)*, pp. 2315–2319, 2016.

[39] J. Serrà, J. Pons, and S. Pascual, "SESQA: semi-supervised learning for speech quality assessment," 2020.

[40] S. Möller, W.-Y. Chan, N. Cote, T. H. Falk, A. Raake, and M. Wältermann, "Speech quality estimation: Models and trends," *IEEE Signal Processing Magazine*, vol. 28, no. 6, pp. 18–28, 2011.

[41] P. C. Loizou, "Speech quality assessment," in *Multimedia analysis, processing and communications*, pp. 623–654, Springer, 2011.

[42] V. Grancharov and W. B. Kleijn, "Speech quality assessment," in *Springer Handbook of Speech Processing*, pp. 83–100, Springer, 2008.

[43] A. W. Rix, "Perceptual speech quality assessment-a review," in *Acoustics, Speech, and Signal Processing, 2004. Proceedings.(ICASSP'04). IEEE International Conference on*, vol. 3, pp. iii–1056, IEEE, 2004.

[44] D. Kim and A. Tarraf, "ANIQUE+: A new American national standard for non-intrusive estimation of narrowband speech quality," *Bell Labs Technical Journal*, vol. 12, pp. 221–236, May 2007.

[45] T. H. Falk, C. Zheng, and W.-Y. Chan, "A non-intrusive quality and intelligibility measure of reverberant and dereverberated speech," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 7, pp. 1766–1774, 2010.

[46] T. Falk and W. Chan, "Single-ended speech quality measurement using machine learning methods," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 14, pp. 1935–1947, Nov. 2006.

[47] International Telecommunications Union (ITU-T), "Mean opinion score (MOS) terminology, Recommendation P.800.1," Online. http://www.itu.int/rec/T-REC-P.800.1-200303-S/en.

[48] International Telecommunications Union (ITU-T), "P. 564: Conformance testing for voice over IP transmission quality assessment models," *Int. Telecomm. Union, Geneva*, 2007.

[49] A. Raake, S. Möller, M. Wältermann, N. Cote, and J.-P. Ramirez, "Parameter-based prediction of speech quality in listening context—towards a WB E-model," in *2010 Second International Workshop on Quality of Multimedia Experience (QoMEX)*, pp. 182–187, IEEE, 2010.

[50] T. H. Falk and W.-Y. Chan, "Hybrid signal-and-link-parametric speech quality measurement for VoIP communications," *IEEE transactions on audio, speech, and language processing*, vol. 16, no. 8, pp. 1579–1589, 2008.

[51] S. Wang, A. Sekey, and A. Gersho, "An objective measure for predicting subjective quality of speech coders," *IEEE J. Sel. Areas Commun.*, vol. 10, pp. 819–829, June 1992.

[52] M. Hollier, M. Hawksford, and D. Guard, "Error activity and error entropy as a measure of psychoacoustic significance in the percep-

tual domain," *IEE Proc.-Vision Image Signal Process.*, vol. 141, pp. 203–208, June 1994.

[53] S. Voran, "Objective estimation of perceived speech quality Part I: Development of the measuring normalizing block technique," *IEEE Trans. Speech Audio Process.*, vol. 7, pp. 371–382, July 1999.

[54] S. Voran, "Objective estimation of perceived speech quality Part II: Evaluation of the measuring normalizing block technique," *IEEE Trans. Speech Audio Process.*, vol. 7, pp. 383–390, July 1999.

[55] J. G. Beerends, C. Schmidmer, J. Berger, M. Obermann, R. Ullmann, J. Pomy, and M. Keyhl, "Perceptual objective listening quality assessment (POLQA), the third generation ITU-T standard for end-to-end speech quality measurement part I—Temporal alignment," *Journal of the Audio Engineering Society*, vol. 61, no. 6, pp. 366–384, 2013.

[56] D. Campbell, E. Jones, and M. Glavin, "Audio quality assessment techniques—A review, and recent developments," *Signal Processing*, vol. 89, pp. 1489–1500, Aug. 2009.

[57] C. Jin and R. Kubichek, "Vector quantization techniques for output-based objective speech quality," *Proc. Int. Conf. on Acoust., Speech, Signal Processing*, vol. 1, pp. 491–494, May 1996.

[58] J. F. Santos, M. Senoussaoui, and T. H. Falk, "An improved non-intrusive intelligibility metric for noisy and reverberant speech," in *2014 14th International Workshop on Acoustic Signal Enhancement (IWAENC)*, pp. 55–59, IEEE, 2014.

[59] G. Chen and V. Parsa, "Nonintrusive speech quality evaluation using an adaptive neurofuzzy inference system," *IEEE Signal Processing Letters*, vol. 12, pp. 403–406, May 2005.

[60] P. N. Petkov, W. B. Kleijn, and B. d. Vries, "Discrete choice models for non-intrusive quality assessment," in *Twelfth Annual Conference of the International Speech Communication Association*, 2011.

[61] P. N. Petkov, I. S. Mossavat, and W. B. Kleijn, "A bayesian approach to non-intrusive quality assessment of speech," in *Tenth Annual Conference of the International Speech Communication Association*, pp. 2875–2878, 2009.

[62] P. N. Petkov and W. B. Kleijn, "Probabilistic non-intrusive quality assessment of speech for bounded-scale preference scores," in *Quality of Multimedia Experience (QoMEX), 2010 Second International Workshop on*, pp. 188–193, 2010.

[63] D. Sharma, A. O. Hogg, Y. Wang, A. Nour-Eldin, and P. A. Naylor, "Non-intrusive POLQA estimation of speech quality using recurrent neural networks," in *2019 27th European Signal Processing Conference (EUSIPCO)*, pp. 1–5, IEEE, 2019.

[64] A. A. Catellier and S. D. Voran, "Wenets: A convolutional framework for evaluating audio waveforms," *arXiv preprint arXiv:1909.09024*, 2019.

[65] J. Ooster and B. T. Meyer, "Improving deep models of speech quality prediction through voice activity detection and entropy-based measures," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 636–640, IEEE, 2019.

[66] L. Ding, Z. Lin, A. Radwan, M. S. El-Hennawey, and R. A. Goubran, "Non-intrusive single-ended speech quality assessment in VoIP," *Speech Communication*, vol. 49, no. 6, pp. 477–489, 2007.

[67] A. Hines, E. Gillen, and N. Harte, "Measuring and monitoring speech quality for voice over IP with POLQA, ViSQOL and P. 563,"

in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[68] H. Yang, K. Byun, H.-G. Kang, and Y. Kwak, "Parametric-based non-intrusive speech quality assessment by deep neural network," in *2016 IEEE International Conference on Digital Signal Processing (DSP)*, pp. 99–103, IEEE, 2016.

[69] G. Paolacci, J. Chandler, and P. G. Ipeirotis, "Running experiments on Amazon Mechanical Turk," *Judgment and Decision making*, vol. 5, no. 5, pp. 411–419, 2010.

[70] M. H. Soni and H. A. Patil, "Novel subband autoencoder features for non-intrusive quality assessment of noise suppressed speech.," in *INTERSPEECH*, pp. 3708–3712, 2016.

[71] R. Schatz, T. Hoßfeld, L. Janowski, and S. Egger, "From packets to people: quality of experience as a new measurement challenge," in *Data traffic monitoring and analysis*, pp. 219–263, Springer, 2013.

[72] A. A. de Lima, F. P. Freeland, P. A. Esquef, L. W. Biscainho, B. C. Bispo, R. A. de Jesus, S. L. Netto, R. W. Schafer, A. Said, B. Lee, *et al.*, "Reverberation assessment in audioband speech signals for telepresence systems.," in *SIGMAP*, pp. 257–262, 2008.

[73] International Telecommunications Union (ITU-T), "P. 830: Subjective performance assessment of digital telephone-band and wideband digital codecs," *International Telecommunication Union, Geneva*, 1996.

[74] International Telecommunications Union (ITU-T), "P. 48: "specification for an intermediate reference system," *International Telecommunication Union*, 1988.

[75] International Telecommunications Union (ITU-T), "P. 56: Objective measurement of active speech level," *International Telecommunication Union*, 1993.

[76] L. R. Rabiner, "Digital processing of speech signal," *Digital Processing of Speech Signal*, 1978.

[77] J. Le Roux and C. Gueguen, "A fixed point computation of partial correlation coefficients," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 25, no. 3, pp. 257–259, 1977.

[78] J. Schur, "Über potenzreihen, die im innern des einheitskreises beschränkt sind.," *Journal für die reine und angewandte Mathematik*, vol. 147, pp. 205–232, 1917.

[79] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 1*, vol. 1, 1986.

[80] M. Hansen and B. Kollmeier, "Continuous assessment of time-varying speech quality," *The Journal of the Acoustical Society of America*, vol. 106, no. 5, pp. 2888–2899, 1999.

[81] France Telecom, RD, "Study of the relationship between instantaneous and overall subjective speech quality for time-varying quality speech sequences: influence of the recency effect," *ITU Study Group 12, contribution D*, vol. 139, 2000.

[82] A. M. Turing, "Computing Machinery and Intelligence," *Mind*, vol. LIX, pp. 433–460, 1950.

[83] A. L. Samuel, "Some studies in machine learning using the game of checkers," *IBM Journal of Research and Development*, vol. 3, no. 3, pp. 210–229, 1959.

[84] E. Alpaydin, *Introduction to Machine Learning*, ch. 1. MIT press, 2010.

[85] M. Mohri, A. Rostamizadeh, and A. Talwalkar, *Foundations of machine learning*, pp. 1–10. MIT press, 2012.

[86] R. K. Ando, T. Zhang, and P. Bartlett, "A framework for learning predictive structures from multiple tasks and unlabeled data," *Journal of Machine Learning Research*, vol. 6, pp. 1817–1853, 2005.

[87] J. Wu, C. Zhang, T. Xue, B. Freeman, and J. Tenenbaum, "Learning a probabilistic latent space of object shapes via 3D generative-adversarial modeling," in *Advances in Neural Information Processing Systems*, pp. 82–90, 2016.

[88] Y. Bengio, F. Bastien, A. Bergeron, N. Boulanger-Lewandowski, T. Breuel, Y. Chherawala, M. Cisse, M. Côté, D. Erhan, J. Eustache, *et al.*, "Deep learners benefit more from out-of-distribution examples," in *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pp. 164–172, 2011.

[89] N. J. Nilsson, "Introduction to Machine Learning. An early draft of a proposed textbook," 1996.

[90] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[91] S. Ruder, "An overview of gradient descent optimization algorithms," *arXiv preprint arXiv:1609.04747*, 2016.

[92] M. Aizerman, E. Braverman, and L. Rozonoer, "Theoretical foundations of the potential function method in pattern recognition learning," *Automation and Remote Control*, no. 25, pp. 821–837, 1964.

[93] G. Zhong, X. Ling, and L.-N. Wang, "From shallow feature learning to deep learning: Benefits from the width and depth of deep architectures," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 9, no. 1, p. e1255, 2019.

[94] N. Cohen, O. Sharir, and A. Shashua, "On the expressive power of deep learning: A tensor analysis," in *Conference on learning theory*, pp. 698–728, 2016.

[95] E. Alpaydin, "Machine learning," *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 3, no. 3, pp. 195–203, 2011.

[96] C. E. Rasmussen and C. K. I. Williams, *Gaussian processes for machine learning*. Cambridge, MA: MIT Press, 2006.

[97] M. E. Tipping, "Sparse Bayesian learning and the relevance vector machine," *J. Mach. Learn. Res.*, vol. 1, pp. 211–244, 2001.

[98] M. E. Tipping and A. C. Faul, "Fast marginal likelihood maximisation for sparse Bayesian models," in *Proc. 9th Int. Workshop Artif. Intelligence and Stat.*, (Key West), 2003.

[99] I. Psorakis, T. Damoulas, and M. A. Girolami, "Multiclass Relevance Vector Machines: Sparsity and Accuracy," *IEEE Trans. on Neural Networks*, vol. 21, no. 10, 2010.

[100] F. R. Burden and D. A. Winkler, "Relevance vector machines: sparse classification methods for QSAR," *Journal of chemical information and modeling*, vol. 55, no. 8, pp. 1529–1534, 2015.

[101] C. M. Bishop and M. E. Tipping, "Variational relevance vector machines," in *Proc. 16th Conf. Uncer. in Artif. Intell.*, (San Francisco), pp. 46–53, Morgan Kaufmann Publishers Inc., 2000.

[102] D. Shutin, T. Buchgraber, S. R. Kulkarni, and H. V. Poor, "Fast adaptive variational sparse Bayesian learning with automatic relevance determination," in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, pp. 2180–2183, May 2011.

[103] B. McWilliams, D. Balduzzi, and J. M. Buhmann, "Correlated random features for fast semi-supervised learning," *Advances in Neural Information Processing Systems*, vol. 26, pp. 440–448, 2013.

[104] C. Williams and M. Seeger, "Using the Nyström method to speed up kernel machines," *In NIPS*, 2001.

[105] S. Kakade and D. Foster, "Multi-view regression via canonical correlation analysis," *In Computational Learning Theory (COLT)*, 2007.

[106] I. Arel, D. C. Rose, and T. P. Karnowski, "Research frontier: deep machine learning a new frontier in artificial intelligence research," *Comp. Intell. Mag.*, vol. 5, pp. 13–18, Nov. 2010.

[107] L. Arnold, S. Rebecchi, S. Chevallier, and H. Paugam-Moisy, "An introduction to deep-learning," in *Advances in Computational Intelligence and Machine Learning, ESANN'2011*, pp. 477–488, Apr. 2011.

[108] J. Xie, H. Lu, D. Nan, and C. Nengbin, "Sparse deep belief net for handwritten digits classification," in *Proceedings of the 2010 international conference on Artificial intelligence and computational intelligence: Part I*, AICI'10, (Berlin, Heidelberg), pp. 71–78, Springer-Verlag, 2010.

[109] V. Nair and G. E. Hinton, "3-D object recognition with deep belief nets," in *Advances in Neural Information Processing Systems 22*.

[110] G. B. Huang, "Learning hierarchical representations for face verification with convolutional deep belief networks," in *Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, CVPR '12, (Washington, DC, USA), pp. 2518–2525, IEEE Computer Society, 2012.

[111] B. Kwolek, "Face detection using convolutional neural networks and Gabor filters," in *Artificial Neural Networks: Biological Inspirations*

– *ICANN 2005* (W. Duch, J. Kacprzyk, E. Oja, and S. Zadrożny, eds.), (Berlin, Heidelberg), pp. 551–556, Springer Berlin Heidelberg, 2005.

[112] A.-R. Mohamed, T. N. Sainath, G. E. Dahl, B. Ramabhadran, G. E. Hinton, and M. A. Picheny, "Deep belief networks using discriminative features for phone recognition," in *ICASSP*, pp. 5060–5063, 2011.

[113] A. Mohamed, G. E. Dahl, and G. E. Hinton, "Acoustic modeling using deep belief networks," *IEEE Transactions on Audio, Speech & Language Processing*, vol. 20, no. 1, pp. 14–22, 2012.

[114] H. Lee, P. Pham, Y. Largman, and A. Ng, "Unsupervised feature learning for audio classification using convolutional deep belief networks," *Advances in neural information processing systems*, vol. 22, pp. 1096–1104, 2009.

[115] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, pp. 1527–1554, July 2006.

[116] Y. Bengio, P. Lamblin, D. Popovici, H. Larochelle, U. D. Montréal, and M. Québec, "Greedy layer-wise training of deep networks," in *In NIPS*, MIT Press, 2007.

[117] M. Ranzato, Y.-L. Boureau, and Y. LeCun, "Sparse Feature Learning for Deep Belief Networks," in *Advances in Neural Information Processing Systems 20* (J. Platt, D. Koller, Y. Singer, and S. Roweis, eds.), pp. 1185–1192, Cambridge, MA: MIT Press, 2008.

[118] R. Salakhutdinov and G. Hinton, "Deep Boltzmann Machines," in *Proceedings of the International Conference on Artificial Intelligence and Statistics*, vol. 5, pp. 448–455, 2009.

[119] H. Larochelle, Y. Bengio, J. Louradour, and P. Lamblin, "Exploring Strategies for Training Deep Neural Networks," *J. Mach. Learn. Res.*, vol. 10, pp. 1–40, June 2009.

[120] A. Grubb and J. A. D. Bagnell, "Boosted Backpropagation Learning for Training Deep Modular Networks," in *Proceedings of the 27th International Conference on Machine Learning*, May 2010.

[121] H. Lee, R. Grosse, R. Ranganath, and A. Y. Ng, "Unsupervised learning of hierarchical representations with convolutional deep belief networks," *Commun. ACM*, vol. 54, pp. 95–103, Oct. 2011.

[122] G. E. Hinton and T. J. Sejnowski, "Optimal perceptual inference," in *In CVPR, Washington DC*, 1983.

[123] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proceedings of the 25th international conference on Machine learning*, ICML '08, (New York, NY, USA), pp. 1096–1103, ACM, 2008.

[124] A. Makhzani, J. Shlens, N. Jaitly, and I. Goodfellow, "Adversarial autoencoders," in *International Conference on Learning Representations*, 2016.

[125] Y. Bengio, *Learning Deep Architectures for AI*, ch. 1. Hanover, MA, USA: Now Publishers Inc., 2009.

[126] J. Weston, F. Ratle, and R. Collobert, "Deep learning via semi-supervised embedding," in *Proceedings of the 25th international conference on Machine learning*, ICML '08, (New York, NY, USA), pp. 1168–1175, ACM, 2008.

[127] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," *CoRR*, vol. abs/1207.0580, 2012.

[128] N. Srivastava, "Improving Neural Networks with Dropout," Master's thesis, University of Toronto, Toronto, Canada, January 2013.

[129] Y. LeCun, Y. Bengio, *et al.*, "Convolutional networks for images, speech, and time series," *The handbook of brain theory and neural networks*, vol. 3361, no. 10, p. 1995, 1995.

[130] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[131] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, *et al.*, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[132] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao, "3D shapenets: A deep representation for volumetric shapes," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[133] J. Salamon and J. P. Bello, "Deep convolutional neural networks and data augmentation for environmental sound classification," *IEEE Signal Processing Letters*, vol. 24, pp. 279–283, March 2017.

[134] R. Schirrmeister, L. Gemein, K. Eggensperger, F. Hutter, and T. Ball, "Deep learning with convolutional neural networks for decoding and visualization of EEG pathology," in *2017 IEEE Signal Processing in Medicine and Biology Symposium (SPMB)*, pp. 1–7, Dec 2017.

[135] T. Mikolov, M. Karafiát, L. Burget, J. Cernocký, and S. Khudanpur, "Recurrent neural network based language model," in *INTERSPEECH*, 2010.

[136] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel, "InfoGan: Interpretable representation learning by information maximizing generative adversarial nets," in *Advances in neural information processing systems*, pp. 2172–2180, 2016.

[137] O. Fabius and J. R. van Amersfoort, "Variational recurrent auto-encoders," *arXiv preprint arXiv:1412.6581*, 2014.

[138] Y. Pu, W. Wang, R. Henao, L. Chen, Z. Gan, C. Li, and L. Carin, "Adversarial symmetric variational autoencoder," in *Advances in Neural Information Processing Systems*, pp. 4330–4339, 2017.

[139] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[140] A. Alemi, B. Poole, I. Fischer, J. Dillon, R. A. Saurous, and K. Murphy, "Fixing a broken ELBO," in *International Conference on Machine Learning*, pp. 159–168, 2018.

[141] A. Van Den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. W. Senior, and K. Kavukcuoglu, "WaveNet: A generative model for raw audio.," in *SSW*, p. 125, 2016.

[142] E. Denton, S. Chintala, A. Szlam, and R. Fergus, "Deep generative image models using a Laplacian pyramid of adversarial networks," 2015.

[143] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training GANs," in *Advances in Neural Information Processing Systems*, pp. 2234–2242, 2016.

[144] X. Huang, Y. Li, O. Poursaeed, J. E. Hopcroft, and S. J. Belongie, "Stacked generative adversarial networks.," in *CVPR*, vol. 2, p. 3, 2017.

[145] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. P. Smolley, "Least squares generative adversarial networks," in *Computer Vision (ICCV), 2017 IEEE International Conference on*, pp. 2813–2821, IEEE, 2017.

[146] D. Michelsanti and Z.-H. Tan, "Conditional generative adversarial networks for speech enhancement and noise-robust speaker verification," *arXiv preprint arXiv:1709.01703*, 2017.

[147] A. Creswell, T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta, and A. A. Bharath, "Generative adversarial networks: An overview," *IEEE Signal Processing Magazine*, vol. 35, no. 1, pp. 53–65, 2018.

[148] M. Mathieu, C. Couprie, and Y. LeCun, "Deep multi-scale video prediction beyond mean square error," *arXiv preprint arXiv:1511.05440*, 2015.

[149] C. Vondrick, H. Pirsiavash, and A. Torralba, "Generating videos with scene dynamics," in *Advances In Neural Information Processing Systems*, pp. 613–621, 2016.

[150] C.-C. Hsu, H.-T. Hwang, Y.-C. Wu, Y. Tsao, and H.-M. Wang, "Voice conversion from unaligned corpora using variational autoencoding Wasserstein generative adversarial networks," *arXiv preprint arXiv:1704.00849*, 2017.

[151] S. Pascual, A. Bonafonte, and J. Serra, "SEGAN: Speech enhancement generative adversarial network," *arXiv preprint arXiv:1703.09452*, 2017.

[152] J. Chang and S. Scherer, "Learning representations of emotional speech with deep convolutional generative adversarial networks," in *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*, pp. 2746–2750, IEEE, 2017.

[153] X. Huang, Y. Li, O. Poursaeed, J. Hopcroft, and S. Belongie, "Stacked generative adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5077–5086, 2017.

[154] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *International Conference on Machine Learning*, pp. 214–223, 2017.

[155] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, "Improved training of Wasserstein GANs," in *Advances in Neural Information Processing Systems*, pp. 5767–5777, 2017.

[156] J. Zhao, M. Mathieu, and Y. LeCun, "Energy-based generative adversarial network," *arXiv preprint arXiv:1609.03126*, 2016.

[157] D. Berthelot, T. Schumm, and L. Metz, "BEGAN: boundary equilibrium generative adversarial networks," *arXiv preprint arXiv:1703.10717*, 2017.

[158] G. Louppe, M. Kagan, and K. Cranmer, "Learning to pivot with adversarial networks," in *Advances in Neural Information Processing Systems*, pp. 981–990, 2017.

[159] J. Gui, Z. Sun, Y. Wen, D. Tao, and J. Ye, "A review on generative adversarial networks: Algorithms, theory, and applications," *arXiv preprint arXiv:2001.06937*, 2020.

[160] S. Kullback and R. A. Leibler, "On information and sufficiency," *The annals of mathematical statistics*, vol. 22, no. 1, pp. 79–86, 1951.

[161] F. Huszar, *Scoring rules, divergences and information in Bayesian machine learning*. PhD thesis, University of Cambridge, 2013.

[162] J. Lin, "Divergence measures based on the Shannon entropy," *IEEE Transactions on Information theory*, vol. 37, no. 1, pp. 145–151, 1991.

[163] C. R. Rao, "Differential metrics in probability spaces," *Differential geometry in statistical inference*, vol. 10, pp. 217–240, 1987.

[164] B. K. Sriperumbudur, K. Fukumizu, A. Gretton, B. Schölkopf, and G. R. Lanckriet, "On integral probability metrics, ϕ-divergences and binary classification," *arXiv preprint arXiv:0901.2698*, 2009.

[165] E. del Barrio, E. Giné, and C. Matrán, "Central limit theorems for the wasserstein distance between the empirical and the true distributions," *Annals of Probability*, pp. 1009–1071, 1999.

[166] A. Gretton, K. M. Borgwardt, M. Rasch, B. Schölkopf, and A. J. Smola, "A kernel method for the two-sample-problem," in *Advances in neural information processing systems*, pp. 513–520, 2007.

[167] N. Aronszajn, "Theory of reproducing kernels," *Transactions of the American mathematical society*, vol. 68, no. 3, pp. 337–404, 1950.

[168] S. T. Rachev *et al.*, "Duality theorems for Kantorovich-Rubinstein and Wasserstein functionals," 1990.

[169] L. Wan, M. Zeiler, S. Zhang, Y. Le Cun, and R. Fergus, "Regularization of neural networks using dropconnect," in *International conference on machine learning*, pp. 1058–1066, 2013.

[170] W. Wang, Y. Huang, Y. Wang, and L. Wang, "Generalized autoencoder: A neural network framework for dimensionality reduction," in *2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 496–503, June 2014.

[171] A. L. Blum and P. Langley, "Selection of relevant features and examples in machine learning," *Artificial intelligence*, vol. 97, no. 1-2, pp. 245–271, 1997.

[172] M. A. Hall and L. A. Smith, "Feature selection for machine learning: Comparing a correlation-based filter approach to the wrapper,"

in *Proceedings of the Twelfth International Florida Artificial Intelligence Research Society Conference*, pp. 235–239, AAAI Press, 1999.

[173] Z. Zhao, F. Morstatter, S. Sharma, S. Alelyani, A. Anand, and H. Liu, "Advancing feature selection research," *ASU feature selection repository*, pp. 1–28, 2010.

[174] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *Journal of Machine Learning Research*, vol. 3, pp. 1157–1182, 2003.

[175] M. A. Ranzato, Y. lan Boureau, and Y. L. Cun, "Sparse feature learning for deep belief networks," in *Advances in Neural Information Processing Systems 20* (J. C. Platt, D. Koller, Y. Singer, and S. T. Roweis, eds.), pp. 1185–1192, Curran Associates, Inc., 2008.

[176] R. Bellman, *Dynamic Programming*. Princeton, NJ, USA: Princeton University Press, 1 ed., 1957.

[177] W. I. Grosky and R. Agrawal, "Narrowing the semantic gap in image retrieval: A multimodal approach," in *Multimedia Information Extraction and Digital Heritage Preservation* (U. M. Munshi and B. B. Chaudhuri, eds.), pp. 89–118, World Scientific, 2011.

[178] E. Keogh and A. Mueen, *Curse of Dimensionality*, pp. 257–258. Boston, MA: Springer US, 2010.

[179] H. Liu and H. Motoda, *Computational Methods of Feature Selection*, ch. 8.4. Chapman & Hall/CRC Data Mining and Knowledge Discovery Series, CRC Press, 2007.

[180] D. Chen, X. Cao, F. Wen, and J. Sun, "Blessing of dimensionality: High-dimensional feature and its efficient compression for face verification," in *Proceedings of the 2013 IEEE Conference on Computer*

Vision and Pattern Recognition, CVPR '13, (Washington, DC, USA), pp. 3025–3032, IEEE Computer Society, 2013.

[181] P. Gehler and S. Nowozin, "On feature combination for multiclass object classification," in *2009 IEEE 12th International Conference on Computer Vision*, pp. 221–228, Sept 2009.

[182] X. Liu, L. Zhang, M. Li, H. Zhang, and D. Wang, "Boosting image classification with LDA-based feature combination for digital photograph management," *Pattern Recogn.*, vol. 38, pp. 887–901, June 2005.

[183] M. Gonen and E. Alpaydin, "Multiple kernel learning algorithms," *J. Mach. Learn. Res.*, vol. 12, pp. 2211–2268, July 2011.

[184] H. Sak, A. Senior, and F. Beaufays, "Long short-term memory recurrent neural network architectures for large scale acoustic modeling," in *Fifteenth annual conference of the international speech communication association*, 2014.

[185] C. Donahue, J. McAuley, and M. Puckette, "Synthesizing audio with generative adversarial networks," *CoRR*, vol. abs/1802.04208, 2018.

[186] R. Ruiz, J. C. R. A, and J. S. A. ruiz B, "Incremental wrapper-based gene selection from microarray data for cancer classification," *Pattern Recognition*, pp. 2383–2392, 2006.

[187] V. Balasubramanian, S. S. Ho, and V. Vovk, *Conformal Prediction for Reliable Machine Learning: Theory, Adaptations and Applications*, ch. 6.4. Elsevier Science, 2014.

[188] S. W. Nydick, "The Wishart and inverse Wishart distributions," *Electronic Journal of Statistics*, vol. 6, pp. 1–19, 2012.

[189] S. B. Kotsiantis and et al., "Data preprocessing for supervised learning," 2006.

[190] S. Ioffe and C. Szegedy, "Batch normalization: accelerating deep network training by reducing internal covariate shift," in *Proceedings of the 32nd International Conference on International Conference on Machine Learning-Volume 37*, pp. 448–456, JMLR. org, 2015.

[191] D. Steinley, "Standardizing variables in k-means clustering," in *Classification, clustering, and data mining applications*, pp. 53–60, Springer, 2004.

[192] D. Larose, "k-Nearest Neighbor Algorithm," in *Discovering Knowledge in Data: An Introduction to Data Mining*, pp. 90–106, Wiley, 2005.

[193] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition*, pp. 61–79. Springer Series in Statistics, Springer New York, 2009.

[194] E. Frank, "Fully supervised training of Gaussian radial basis function networks in WEKA," Tech. Rep. 04/14, Department of Computer Science, University of Waikato, 2014.

[195] A. B. Graf and S. Borer, "Normalization in support vector machines," in *Joint pattern recognition symposium*, pp. 277–282, Springer, 2001.

[196] S. Garcia, "Data preprocessing in data mining," 2014.

[197] M. J. L. Orr, "Introduction to radial basis function networks," 1996.

[198] R. Zhao, H. Zhang, J. Lu, C. Li, and H. Zhang, "A new machine learning method based on PCA and SVM," in *2006 International Conference on Computational Intelligence and Security*, vol. 1, pp. 187–190, Nov 2006.

[199] W. S. Sarle, "Faq - part 2: Learning," Oct. 2002.

[200] D. Pyle, *Data Preparation for Data Mining*, ch. 7. No. v. 1 in Data Preparation for Data Mining, Morgan Kaufmann Publishers, 1999.

[201] D. Cousineau and S. Chartier, "Outliers detection and treatment: a review.," *International Journal of Psychological Research*, vol. 3, pp. 58–67, 2010.

[202] K. J. Max Kuhn, *Applied Predictive Modeling*, ch. 3. Data Preparation for Data Mining, Springer-Verlag New York, 2013.

[203] G. E. P. Box and D. R. Cox, "An analysis of transformations," *Journal of the Royal Statistical Society. Series B (Methodological*, pp. 211–252, 1964.

[204] J. L. Devore and K. N. Berk, *Modern Mathematical Statistics with Applications*, ch. 4.7. Springer Texts in Statistics, New York, NY: Springer New York, 2012.

[205] M. S. Waterman and D. E. Whiteman, "Estimation of probability densities by empirical density functions," *Int. J. Math. Educ. Sci. Technol*, vol. 9, pp. 127–137, 1978.

[206] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 39, no. 1, pp. 1–38, 1977.

[207] J. Patel and C. Read, *Handbook of the normal distribution*, pp. 179–240. New York: M. Dekker, 1982.

[208] A. W. Rix, "Comparison between subjective listening quality and P.862 PESQ score," *Proc. Meas. Speech Qual. Net. (MESAQIN)*, pp. 17–25, 2003.

[209] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An introduction to statistical learning: With Applications in R*, vol. 112, pp. 175–187. Springer, 2013.

[210] R. Salami, C. Laftamme, J.-P. Adoul, A. Kataoba, S. Hayashi, C. Lamblin, D. Massaloux, S. Proust, P. Kroon, and Y. Shoham, "Description of the proposed ITU-T 8 kb/s speech coding standard," in *Proceedings. IEEE Workshop on Speech Coding for Telecommunications*, pp. 3–4, IEEE, 1995.

[211] International Telecommunications Union (ITU-T), "SQ-46.95R3: Subjective Test Plan for Characterization of an 8 kbit/s Speech Codec," September 1995.

[212] H. Akaike, "Information theory and an extension of the maximum likelihood principle," in *Second International Symposium on Information Theory* (B. N. Petrov and F. Csaki, eds.), (Budapest), pp. 267–281, Akadémiai Kiado, 1973.

[213] W. A. Jassim and M. S. Zilany, "NSQM: A non-intrusive assessment of speech quality using normalized energies of the neurogram," *Computer Speech & Language*, vol. 58, pp. 260–279, 2019.

[214] G. Chen and V. Parsa, "A Bayesian estimator for non-intrusive speech quality evaluation in psychoacoustic domain," in *2006 IEEE International Symposium on Signal Processing and Information Technology*, pp. 438–441, Aug 2006.

[215] D. Kim, *Personal Communication*.

[216] J. H. Zar, *Biostatistical analysis*, pp. 43–45. Upper Saddle River, N.J. : Prentice-Hall, 1999.

[217] W. C. Navidi, *Statistics for engineers and scientists*, pp. 400–523. McGraw-Hill Higher Education New York, NY, USA, 2014.

[218] R. A. Fisher, "Statistical methods for research workers. oliver and boyd, edinburgh (1925)," *Google Scholar*, 1950.

[219] M. Cowles and C. Davis, "On the origins of the .05 level of statistical significance.," *American Psychologist*, vol. 37, no. 5, p. 553, 1982.

[220] W. B. Kleijn, F. S. Lim, A. Luebs, J. Skoglund, F. Stimberg, Q. Wang, and T. C. Walters, "WaveNet based low rate speech coding," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 676–680, IEEE, 2018.

[221] Y. Bengio, "Learning deep architectures for AI," *Foundations and Trends in Machine Learning*, vol. 2, no. 1, pp. 1–127, 2009.

[222] D. H. Hubel and T. N. Wiesel, "Receptive fields, binocular interaction and functional architecture in the cat's visual cortex," *The Journal of physiology*, vol. 160, no. 1, pp. 106–154, 1962.

[223] D. H. Hubel and T. N. Wiesel, *Brain and visual perception: the story of a 25-year collaboration*. Oxford University Press, 2004.

[224] T. N. Wiesel and D. H. Hubel, "Comparison of the effects of unilateral and bilateral eye closure on cortical unit responses in kittens," *Journal of neurophysiology*, vol. 28, no. 6, pp. 1029–1040, 1965.

[225] D. H. Hubel and T. N. Wiesel, "The period of susceptibility to the physiological effects of unilateral eye closure in kittens," *The Journal of physiology*, vol. 206, no. 2, pp. 419–436, 1970.

[226] D. H. Hubel and T. N. Wiesel, "Ferrier lecture-functional architecture of macaque monkey visual cortex," *Proc. R. Soc. Lond. B*, vol. 198, no. 1130, pp. 1–59, 1977.

[227] E. H. Lenneberg, "The biological foundations of language," *Hospital Practice*, vol. 2, no. 12, pp. 59–67, 1967.

[228] D. Purves, G. Augustine, D. Fitzpatrick, L. Katz, A. LaMantia, J. McNamara, and S. Williams, "The development of language: A criti-

cal period in humans," *Neuroscience. Sunderland: Sinauer Associates*, 2001.

[229] H. J. Neville and D. Bavelier, "Neural organization and plasticity of language," *Current opinion in Neurobiology*, vol. 8, no. 2, pp. 254–258, 1998.

[230] C. E. Snow and M. Hoefnagel-Höhle, "The critical period for language acquisition: Evidence from second language learning," *Child development*, pp. 1114–1128, 1978.

[231] J. F. Werker and R. C. Tees, "Phonemic and phonetic factors in adult cross-language speech perception," *The Journal of the Acoustical Society of America*, vol. 75, no. 6, pp. 1866–1878, 1984.

[232] Ç. Aytekin, X. Ni, F. Cricri, and E. Aksu, "Clustering and unsupervised anomaly detection with L2 normalized deep auto-encoder representations," *CoRR*, vol. abs/1802.00187, 2018.

[233] J. An and S. Cho, "Variational autoencoder based anomaly detection using reconstruction probability," *Special Lecture on IE*, vol. 2, no. 1, pp. 1–18, 2015.

[234] Y. Xia, X. Cao, F. Wen, G. Hua, and J. Sun, "Learning discriminative reconstructions for unsupervised outlier removal," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1511–1519, 2015.

[235] M. Goldstein and S. Uchida, "A comparative evaluation of unsupervised anomaly detection algorithms for multivariate data," *PloS one*, vol. 11, no. 4, p. e0152173, 2016.

[236] H. Zenati, C. S. Foo, B. Lecouat, G. Manek, and V. R. Chandrasekhar, "Efficient GAN-Based anomaly detection," *CoRR*, vol. abs/1802.06222, 2018.

[237] T. Schlegl, P. Seeböck, S. M. Waldstein, U. Schmidt-Erfurth, and G. Langs, "Unsupervised anomaly detection with generative adversarial networks to guide marker discovery," *CoRR*, vol. abs/1703.05921, 2017.

[238] D. Elbaz and M. Zibulevsky, "Perceptual audio loss function for deep learning," *CoRR*, vol. abs/1708.05987, 2017.

[239] K. Lin, D. Li, X. He, Z. Zhang, and M.-T. Sun, "Adversarial ranking for language generation," in *Advances in Neural Information Processing Systems*, pp. 3155–3165, 2017.

[240] M. O. Vertolli and J. Davies, "Image quality assessment techniques show improved training and evaluation of autoencoder generative adversarial networks," *arXiv preprint arXiv:1708.02237*, 2017.

[241] L. Theis, A. Oord, and M. Bethge, "A note on the evaluation of generative models," *CoRR*, vol. abs/1511.01844, 2016.

[242] A. Grover, M. Dhar, and S. Ermon, "Flow-GAN: Combining maximum likelihood and adversarial learning in generative models," in *Thirty-second AAAI conference on artificial intelligence*, 2018.

[243] V. Dumoulin, I. Belghazi, B. Poole, O. Mastropietro, A. Lamb, M. Arjovsky, and A. Courville, "Adversarially learned inference," *arXiv preprint arXiv:1606.00704*, 2016.

[244] J. Donahue, P. Krähenbühl, and T. Darrell, "Adversarial feature learning," *arXiv preprint arXiv:1605.09782*, 2016.

[245] J.-Y. Zhu, P. Krähenbühl, E. Shechtman, and A. A. Efros, "Generative visual manipulation on the natural image manifold," in *European Conference on Computer Vision*, pp. 597–613, Springer, 2016.

[246] A. Creswell and A. A. Bharath, "Inverting the generator of a generative adversarial network (ii)," *arXiv preprint arXiv:1802.05701*, 2018.

[247] Z. C. Lipton and S. Tripathi, "Precise recovery of latent vectors from generative adversarial networks," *arXiv preprint arXiv:1702.04782*, 2017.

[248] Y. LeCun and C. Cortes, "MNIST, the modified national institute of standards and technology database," 2010. http://yann.lecun.com/exdb/mnist/.

[249] E. Kussul and T. Baidyk, "Improved method of handwritten digit recognition tested on mnist database," *Image and Vision Computing*, vol. 22, no. 12, pp. 971–981, 2004.

[250] L. Schott, J. Rauber, M. Bethge, and W. Brendel, "Towards the first adversarially robust neural network model on MNIST," *arXiv preprint arXiv:1805.09190*, 2018.

[251] R. F. Alvear-Sandoval, J. L. Sancho-Gómez, and A. R. Figueiras-Vidal, "On improving CNNs performance: The case of MNIST," *Information Fusion*, vol. 52, pp. 106–109, 2019.

[252] L. Deng, "The MNIST database of handwritten digit images for machine learning research," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 141–142, 2012.

[253] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms," *arXiv preprint arXiv:1708.07747*, 2017.

[254] J. S. Garofolo, "TIMIT acoustic phonetic continuous speech corpus," *Linguistic Data Consortium, 1993*, 1993.

[255] C. Lopes and F. Perdigao, "Phone recognition on the TIMIT database," *Speech Technologies*, vol. 1, pp. 285–302, 2011.

[256] L. Deng, D. Yu, and J. Platt, "Scalable stacking and learning for building deep architectures," in *2012 IEEE International conference on*

Acoustics, speech and signal processing (ICASSP), pp. 2133–2136, IEEE, 2012.

[257] Wikipedia contributors, "Seven-segment display — Wikipedia, the free encyclopedia," 2021. [Online; accessed 13-July-2021].

[258] Yi Hu and P. C. Loizou, "Subjective comparison of speech enhancement algorithms," in *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, vol. 1, pp. I–I, 2006.

[259] V. Zue, S. Seneff, and J. Glass, "Speech database development at MIT: TIMIT and beyond," *Speech Communication*, vol. 9, no. 4, pp. 351 – 356, 1990.

[260] Y. Hu and P. C. Loizou, "Subjective comparison and evaluation of speech enhancement algorithms," *Speech Communication*, vol. 49, no. 7, pp. 588 – 601, 2007. Speech Enhancement.

[261] I. of Electrical and E. Engineers, "IEEE Recommended Practice for Speech Quality Measurements," *IEEE Trans. Audio Electroacoust.*, vol. 17, pp. 225–246, 1969.

[262] H.-G. Hirsch and D. Pearce, "The Aurora experimental framework for the performance evaluation of speech recognition systems under noisy conditions," in *ASR2000-Automatic speech recognition: challenges for the new Millenium ISCA tutorial and research workshop (ITRW)*, 2000.

[263] L. Wyse, "Audio spectrogram representations for processing with convolutional neural networks," *CoRR*, vol. abs/1706.09559, 2017.

[264] M. Dörfler, R. Bammer, and T. Grill, "Inside the spectrogram: Convolutional neural networks in audio processing," in *2017 International Conference on Sampling Theory and Applications (SampTA)*, pp. 152–155, 2017.

[265] S. Becker, M. Ackermann, S. Lapuschkin, K. Müller, and W. Samek, "Interpreting and explaining deep neural networks for classification of audio signals," *CoRR*, vol. abs/1807.03418, 2018.

[266] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," tech. rep., Citeseer, 2009.

[267] A. J. R. Simpson, "Time-frequency trade-offs for audio source separation with binary masks," *CoRR*, vol. abs/1504.07372, 2015.

[268] L. Dinh, J. Sohl-Dickstein, and S. Bengio, "Density estimation using real NVP," in *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*, OpenReview.net, 2017.