PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

RENATO QUIRINO DE ALBUQUERQUE

A CONVOLUTIONAL NEURAL NETWORK APPROACH FOR SPEECH QUALITY ASSESSMENT

Universidade Federal de Pernambuco
posgraduacao@cin.ufpe.br
www.cin.ufpe.br/~posgraduacao

Recife

2020

RENATO QUIRINO DE ALBUQUERQUE

A CONVOLUTIONAL NEURAL NETWORK APPROACH FOR SPEECH QUALITY ASSESSMENT

Este trabalho foi apresentado à Pós-Graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco como requisito parcial para obtenção do grau de Mestre em Ciência da Computação.

Área de Concentração: Ciência da Computação

Orientador(a): Profº. Dr. Carlos Alexandre Barros de Mello

Recife

2020

Catalogação na fonte
Bibliotecária Monick Raquel Silvestre da S. Portes, CRB4-1217

A345c Albuquerque, Renato Quirino de
A convolutional neural network approach for speech quality assesment / Renato Quirino de Albuquerque. – 2020.
106 f.: il., fig., tab.

Orientador: Carlos Alexandre Barros de Mello.
Dissertação (Mestrado) – Universidade Federal de Pernambuco. CIn, Ciência da Computação, Recife, 2020.
Inclui referências.

1. Ciência da computação. 2. Redes neurais convolucionais. I. Mello, Carlos Alexandre Barros de (orientador). II. Título.

004 CDD (23. ed.) UFPE - CCEN 2020 - 132

Renato Quirino de Albuquerque

“A Convolutional Neural Network Approach for Speech Quality Assessment”

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Pernambuco, como requisito parcial para a obtenção do título de Mestre em Ciência da Computação

Aprovado em: 20/02/2020.

BANCA EXAMINADORA

____
Prof. Dr. Adriano Lorena Inácio de Oliveira
Centro de Informática / UFPE

____
Prof. Dr. Francisco Madeiro Bernardino Junior
Centro de Ciências e Tecnologia/UNICAP

____
Prof. Dr. Carlos Alexandre Barros de Mello
Centro de Informática / UFPE
(**Orientador**)

ACKNOWLEDGEMENTS

I would like to thank Carlos Alexandre Barros de Mello, my advisor at the Centro de Informática (CIn), for open-minded supervision, for his advices and for helping me to write this document. I am also grateful to Everton Barbosa Lacerda for supplying me with meaningful ideias and interesting links related to my research area. Furthermore, I would like to thank the examination commitee composed by Adriano Lorena Inacio de Oliveira and Francisco Madeiro Bernardino Junior for their suggestions to improve this document. Lastly, I am grateful to my family, friends and my girlfriend for supporting me during the development of this work.

ABSTRACT

An important aspect of speech understanding is quality, which can be defined as the fidelity of the signal in relation to its original (or idealized) version when a comparison is allowed. Despite being a subjective issue, there are approaches to measuring speech quality. The most effective approach consists of applying subjective tests, in which individuals evaluate the quality of the speech samples, associating them with quality indexes. However, there are automatic measurement applications that operate at lower costs and generate faster responses. Such solutions can be divided into methodologies that use only the sample to be evaluated (*non-reference*) and those that use the degraded and reference versions of the speech sample (*full-reference*). Unfortunately, for many current applications, it is impossible to obtain the original speech sample, requiring the development and application of *non-reference* techniques. Thus, this dissertation presents a model of convolutional neural network for speech quality assessment (CNN-SQA). This is a *non-reference* methodology that applies fully convolutional layers as extractors of characteristics for speech representation. In addition, fully-connected layers are used to perform the quality assessment step. For the entry of the model, some visual characteristics were evaluated, despite the use of MFCC coefficients having presented the best results. Other parameters of the new model were obtained through an iterative and incremental parameter selection process. The performance of the model was evaluated by comparing it with the PESQ, ViSQOL and P.563 methodologies. Other experiments present analyzes of the model's behavior in isolated situations of speech and noise. The experiments were carried out on publicly available databases, as well as on a new database built to evaluate the new methodology in the context of background noise. Finally, the results were analyzed using correlation measures and statistical descriptions.

Keywords: Speech. Quality. Automatic assessment. Speech quality assessment. Convolutional Neural Networks.

RESUMO

Um aspecto importante do entendimento da fala é a qualidade, esta pode ser entendida como a fidelidade do sinal em relação à sua versão original (ou idealizada) quando uma comparação é permitida. Apesar de ser uma questão subjetiva, existem abordagens para medir a qualidade de fala. A abordagem mais eficaz consiste na aplicação de testes subjetivos, nos quais os indivíduos avaliam a qualidade de amostras de fala, associando-as a índicies de qualidade. No entanto, existem aplicações de medição automática que operam a custos mais baixos e geram respostas mais rápidas. Tais soluções podem ser divididas em metodologias que usam apenas a amostra a ser avaliada (*non-reference*) e aquelas que usam as versões degradada e de referência da amostra de fala (*full-reference*). Infelizmente, para muitas aplicações atuais, é impossível obter a amostra de fala original, contribuindo para o desenvolvimento e a aplicação de técnicas (*non-reference*). Assim, esta dissertação apresenta um modelo de rede neural convolucional para avaliação da qualidade de fala (CNN-SQA). Essa é uma metodologia (*non-reference*) que aplica camadas completamente convolucionais como extratores de características para representação da fala. Além disso, camadas completamente conectadas são utilizadas para executar a etapa de avaliação de qualidade. Para a entrada do modelo algumas características visuais foram avaliadas, apesar do uso de coeficientes MFCC ter apresentado os melhores resultados. Outros parâmetros do novo modelo foram obtidos através de um processo iterativo e incremental de seleção de parâmetros. O desempenho do modelo foi avaliado comparando-o com as metodologias PESQ, ViSQOL e P.563. Outros experimentos apresentam análises do comportamento do modelo em situações isoladas de fala e ruído. Os experimentos foram realizados em bancos de dados publicamente disponíveis, bem como em um novo banco de dados construído para avaliar a nova metodologia no contexto de ruído de fundo. Por fim, os resultados foram analisados usando medidas de correlação e descrições estatísticas.

Palavras-chave: Fala. Qualidade. Avaliação automática. Avaliação da qualidade de fala. Redes Neurais Convolucionais.

LIST OF FIGURES

Figure 1 – Representation schematic for the speech quality judgment by a subject. . . . 23
Figure 2 – Input scheme of a neural network architecture for digit image classification. 31
Figure 3 – Convolutional receptive fields connections. 33
Figure 4 – Feature maps of a convolutional layer. 34
Figure 5 – The result volume of applying the pooling layer. 36
Figure 6 – Pooling layer application using a max pooling function. 36
Figure 7 – A complete CNN architecture with the input, convolutional layers, pooling layers, fully-connected layers and the output. 37
Figure 8 – Error rate and name of the winning architecture from ImageNet Challenge from 2010 to 2015. 38
Figure 9 – Traditional Pattern Recognition architecture diagram. 39
Figure 10 – Lenet-5 architecture. 40
Figure 11 – AlexNet architecture. 42
Figure 12 – NIN network architecture. 45
Figure 13 – Inception module. 47
Figure 14 – Extended Inception module. 48
Figure 15 – Google net parameters. 49
Figure 16 – Residual learning block. 50
Figure 17 – Flat architecture (left) and ResNet architecture (right). 52
Figure 18 – Convolutional blocks with shortcut connections of similar complexity due to the use of the right bottleneck model. 53
Figure 19 – CNN-SQA model steps. 57
Figure 20 – Block diagram of the proposed model. 60
Figure 21 – Diagram of the mel spectrogram generation. 62
Figure 22 – Diagram of the MFCC generation. 63
Figure 23 – Diagram of the PNCC generation. The blue boxes are the ones which does not exist in the MFCC generation diagram. The Green boxes represents the equivalent operations in the MFCC generation diagram. 64
Figure 24 – Database generation steps. 73
Figure 25 – Tensorflow graph of CNN-SQA model. 80

Figure 26 – The training and validation curves for ITU-T Supplement 23 and TCD-VoIP mixed database. On top the training curve, in the middle the validation curve and in the bottom is shown the overlapping curves. 81
Figure 27 – The training and validation curves with smoothing of 0.75 for ITU-T Supplement 23 and TCD-VoIP mixed database. On top the training curve, in the middle the validation curve and in the bottom is shown the overlapping curves. 82
Figure 28 – The training and validation curves for SNDSuppl23 database. On top the training curve, in the middle the validation curve and in the bottom is shown the overlapping curves. 84
Figure 29 – The training and validation curves with smoothing of 0.75 for SNDSuppl23 database. On top the training curve, in the middle the validation curve and in the bottom is shown the overlapping curves. 85
Figure 30 – Scatter results (ITU-T Suppl. 23 and TCD-VoIP). Objective measures against subjective MOS scores using ITU-T Suppl. 23 (Experiments 1 and 3, in blue color) and TCD-VOIP (background noise, clip and competing speakers, in orange color). 87
Figure 31 – Scatter results (ITU-T Suppl. 23 and TCD-VoIP). Objective measures against subjective MOS scores using ITU-T Suppl. 23 (Experiments 1 and 3) and TCD-VOIP (background noise, clip and competing speakers). The red lines are the regression lines generated and the black lines are the ideal reference lines. 88
Figure 32 – Scatter plot of the scores for the CNN-SQA model when training with SNDSuppl23. Objective MOS (vertical) against subjective MOS (horizontal). . . 90
Figure 33 – Scatter plot of the scores for the CNN-SQA model when training with SNDSuppl23. Objective MOS (vertical) against subjective MOS (horizontal). The red line is a linear regression generated for shows the scores set, and the black line a reference line of a ideal linear regression result. 90
Figure 34 – The scatter plot with the samples colored by SNR. 91
Figure 35 – The scatter plot of predictions by Ground Truth (GT) grouped by noise type and colored conform SNR increase. 92
Figure 36 – The KDE plot of predictions by ground truth (GT) grouped by noise type. . 92

LIST OF TABLES

Table 1 – The attributes of the convolutional layers of the AlexNet model. 44
Table 2 – The attributes of the LRN layers of the AlexNet model. 44
Table 3 – The attributes of the pooling layers of the AlexNet model. 44
Table 4 – The attributes of the fully-connected layers of the AlexNet model. 44
Table 5 – Initial feature parameters of the proposed model used for parameter selection step. 59
Table 6 – Initial convolutional layer parameters of the proposed model used for parameter selection step. 59
Table 7 – Initial fully-connected layer parameters of the proposed model used for parameter selection step. 60
Table 8 – Resume of ITU-T Rec. Suppl. 23 database. 69
Table 9 – TCD-VoIP database conditions. 70
Table 10 – Information of noisy samples of ITU-T Rec. Suppl. 23. 72
Table 11 – Parameters related to the input signal. 75
Table 12 – Parameters related to spectrogram generation. 75
Table 13 – Parameters related to feature generation. 75
Table 14 – Convolutional layer parameters. 76
Table 15 – Additional convolutional layers parameters. 77
Table 16 – Fully connected layers parameters. 77
Table 17 – Database partitions sets. 77
Table 18 – Training parameters. 78
Table 19 – Training steps and learning rates. 78
Table 20 – Evaluation properties. 79
Table 21 – Summary of training and validation steps for ITU-T Supplement 23 and TCD-VoIP mixed database. 83
Table 22 – Summary of training and validation steps for SNDSuppl23 database. 83
Table 23 – Pearson correlation for all models on validation set using the database composed by samples from the ITU-T Suppl. 23 and TCD-VOIP datasets. 88
Table 24 – Standard deviation for all models on validation set using the database composed by samples from the ITU-T Suppl. 23 and TCD-VOIP datasets. 89

Table 25 – Pearson correlation for all models on validation set using the SNDSuppl23. . 91

LIST OF ACRONYMS

ACR	Absolute Categorical Rate
ANIQUE	Auditory Non-Intrusive Quality Estimation
ASR	Automatic Speech Recognition
BLSTM	Bidirectional Long Short-Term memory
CCR	Comparison Category Rating
CNN	Convolutional Neural Network
CNN-SQA	Convolutional Neural Network approach for Speech Quality Assessment
CVPR	Conference on Computational Vision and Pattern Recognition
DBN	Deep Belief Networks
DCT	Discrete Cosine Transform
DL	Deep Learning
DNN	Deep Neural Networks
ELU	Exponential linear unit
FaNT	Filtering and Noise Adding Tool
FEC	Forward Error Correction
FNN	Fully-connected Neural Network
GMM	Gaussian Mixture Models
GT	Ground Truth
HMM	Hidden Markov Models
ILSVRC	ImageNet Large-Scale Visual Recognition Challenge
LRN	Local Response Normalization

MFCC	Mel Frequency Cepstral Coefficients
MIRS	Modified Intermediate Reference System
MLP	Multilayer Perceptron
MOS	Mean Opinion Score
NIN	Network In Network
NN	Neural Networks
PCA	Principal Components Analysis
PCM	Pulse Code Modulation
PESQ	Perceptual Evaluation of Speech Quality
PLC	Packet Loss Concealment
PNCC	Power Normalized Cepstral Coefficients
POLQA	Perceptual Objective Listening Quality Assessment
PSQM	Perceptual Speech Quality Measure
RBF	Euclidean Radial Basis Function units
RBM	Restricted Boltzmann Machine
RCS	Random Circular Shifting
ReLU	Rectified Linear Units
RMSE	Root Mean Squared Error
SIP	Speech Intelligibility Prediction
SNDSuppl23	Speech Noise Database with Rec. Suppl23
SNR	Signal-to-Noise Ratio
SoX	Sound eXchange
SRMR	Speech to Reverberation Modulation energy Ratio
STFT	Short Time Fourier Transform
STOI	Short-Time Objective Intelligibility
ViSQOL	Virtual Speech Quality Objective Listener
VoIP	Voice over Internet Protocol

CONTENTS

1 INTRODUCTION 17

1.1 MOTIVATION 19

1.2 OBJECTIVES 20

1.3 DISSERTATION STRUCTURE 21

2 SPEECH QUALITY 22

2.1 DEFINITION 22

2.2 QUALITY FACTORS 23

2.2.1 Degradation 23

2.2.1.1 Background Noise 24

2.2.1.2 Codecs 24

2.2.1.3 Delay 24

2.2.1.4 Jitter 25

2.2.1.5 Packet Loss 25

2.2.1.6 Echo 26

2.2.2 Subjective Effects 26

2.2.3 Mix of effects 27

2.3 MEASUREMENT 27

2.3.1 Full-reference methodologies 28

2.3.1.1 PESQ 28

2.3.1.2 POLQA 29

2.3.1.3 VISQOL 29

2.3.2 Non-reference methodologies 30

2.3.2.1 ANIQUE 30

2.3.2.2 ITU-T Recommendation P.563 30

3 CONVOLUTIONAL NEURAL NETWORKS 31

3.1 GENERAL ARCHITECTURE OF A CNN 32

3.1.1 Input 32

3.1.2 Convolutional layer 33

3.1.3 Pooling layer . . . 35

3.1.4 Fully-connected layer . . . 37

3.2 IMAGENET . . . 37

3.3 CLASSICAL ARCHITECTURES . . . 38

3.3.1 Lenet-5 . . . 39

3.3.2 AlexNet . . . 41

3.3.2.1 Activation Function . . . 41

3.3.2.2 Training . . . 41

3.3.2.3 Local Response Normalization . . . 42

3.3.2.4 Overlapping Pooling . . . 43

3.3.2.5 Architecture . . . 43

3.3.2.6 Decreasing Overfitting . . . 44

3.3.3 GoogleNet . . . 45

3.3.3.1 Inception Architecture . . . 46

3.3.4 ResNet . . . 48

3.4 SPEECH RELATED MODELS . . . 52

3.4.1 Speech Recognition . . . 53

3.4.2 Speech Enhancement . . . 54

3.4.3 Speech Intelligibility . . . 55

3.4.4 Speech Assessment . . . 56

4 PROPOSED MODEL FOR SPEECH QUALITY ASSESSMENT BASED ON CNN . . . 57

4.1 OVERVIEW . . . 57

4.2 CNN-SQA . . . 59

4.2.0.1 Parameter selection . . . 59

4.2.1 Input Transformations . . . 60

4.2.1.1 Spectrogram . . . 61

4.2.1.2 MFCC . . . 61

4.2.2 Feature Extraction . . . 64

4.2.2.1 Convolutional Layers . . . 64

4.2.2.2 ReLU . . . 65

4.2.2.3 Batch Normalization . . . 66

4.2.2.4 Dropout 66
4.2.3 Quality Estimation 66

5 EXPERIMENTS AND ANALYSIS 68
5.1 DATABASES 68
5.1.1 ITU-T Supplement 23 and TCD-VoIP 68
5.1.1.1 ITU-T Supplement 23 68
5.1.1.2 TCD-VoIP 69
5.1.1.3 Data Augmentation 70
5.1.1.4 Consideration 71
5.1.2 SNDSuppl23 71
5.1.2.1 Requirements 72
5.1.2.2 Speech source 72
5.1.2.3 Noise source 72
5.1.2.4 Conditions 73
5.1.2.5 Database generation 73
5.2 TRAINING METHODOLOGY 74
5.2.1 Settings 74
5.2.1.1 Input 74
5.2.1.2 Spectrogram 75
5.2.1.3 Feature Extraction 75
5.2.1.4 Convolutional Layers 76
5.2.1.5 Fully connected Layers 77
5.2.1.6 Database Partition 77
5.2.1.7 Training 77
5.2.1.8 Evaluation 78
5.2.2 Implementation 79
5.2.3 Training 79
5.2.4 ITU-T Supplement 23 and TCD-VoIP 79
5.2.5 SNDSuppl23 83
5.3 RESULTS 83
5.3.1 Performance metrics 86
5.3.2 ITU-T Supplement 23 and TCD-VoIP 86

5.3.3 SNDSuppl23 89

5.3.3.1 General Description 89

5.3.3.2 Categorized Description 90

6 CONCLUSIONS 94

6.1 CONTRIBUTIONS 95

6.2 FUTURE WORKS 96

REFERENCES 98

1 INTRODUCTION

As one of the main human characteristics, the speech is the result of a process that starts in the human vocal apparatus. There, it is produced a structured set of continuous sounds which, when following language rules, becomes understandable by other humans (MCLOUGHLIN, 2009). The speech can be considered a subsystem of a language, used to conduct communication throughout the acoustic channel (RAAKE, 2006). Three main components can be extracted from the human communication process: the produced speech sample, the underlying language and lastly the listener. The object of study of this work is focused on the last component, that involves a human being able to listen and to understand the speech sample. As a matter of human perception, speech understanding is a complex topic that involves the human auditory system and brain processes needed to decode the speech message.

The two main aspects of speech understanding can be defined as intelligibility and quality (MCLOUGHLIN, 2009) (both strongly connected to each other). The first one is concerned on the original message information contained in a speech signal; in this view, even a highly noise speech sample can be considered as with highly intelligibility if its content can be understandable. The quality aspect describes the fidelity of the speech signal in relation of its original (or idealized) version. This does not mean that the speech signal can not be changed, but that the processed version of it must have high similarity with the reference version to achieve high fidelity. Quality is a subjective matter and, due to that, is quite difficult to measure and to predict.

In this work, the focus is on the quality aspect of the speech; the reader can learn more about speech intelligibility consulting (PAVLOVIC, 2018). Although the quality aspect of the speech can be considered a highly subjective metric, there are ways to measure it from a speech sample by means of auditory tests, or even it can be estimated by instrumental methods.

The more confident way to measure quality in speech processing systems is the use of subjective listening tests. On such tests, subjects evaluate the quality of speech samples associating it to a score. One often choice to determine the overall sample score is the use of scales, that insert some semantic to the evaluation result; the most commonly used is the Mean Opinion Score (MOS) (ITU-T, 2016). In its most used version, it is used a scale of five possible rates (with scales graded as integers from 1, meaning the worst quality, to 5, meaning the best quality), the average score is used as the final quality measure. Although the use of subjective listening tests is the most reliable method to evaluate speech quality, there are relevant details concerned to apply them (ITU–T, 1996a), some of them can be listed as:

- It is expensive since depend on high quality sound equipments and specific test conditions.
- It is time consuming since they depend of laborious actions such as to recruit people, to organize experimentation, to acquire database samples, etc.
- It is not practical for systems that needs real-time analysis.
- It is very difficult to be reproduced.

Besides the problems when employing subjective listening tests, the use of digital technologies in telecommunications networks has promoted the emergence of various types of services that increasingly require diagnostic tools. As an example, the development of the fourth generation (4G) for mobile networks technology, making possible the internet access, expanding the use of Voice over Internet Protocol (VoIP) technologies by applications as Hangouts, WhatsApp and Skype. These events contributed to increase the need for development of automatic quality assessment approaches. Thus, there have been an increase number of methodologies for instrumental quality estimation (MÖLLER et al., 2011). These are tools devoted to approximate the subjective quality for an average user automatically.

Since the birth of automatic quality assessment, methodologies dedicated to addressing an increasing number of degradations have raised. Some models have proven effective, becoming international recommendations (ITU-T, 1998), (ITU–T, 2005), (ITU–T, 2011a).

Since the emergence of the area, the vast majority of methodologies that have emerged perform quality assessment by using hand-crafted algorithms, highly based on speech and hearing characteristics. However, because the task is based on modeling human perception, it is not always possible to perfectly compute all sensory effects of a human being. Besides that, each person has a different perception of the outside world based on their own experiences, making the modeling task more complex. With these problems in mind, new methodologies have arisen employing techniques more similar to the human learning process, based on machine learning to model the quality perception stages (ROZHON et al., 2016), (XIE et al., 2016), (WANG et al., 2018).

In the last few years, the use of Deep Learning (DL) to solve classical speech processing problems turned the attention to the high accuracy capacity of such models in subjective tasks. More specifically, such advances conducted the development of approaches applied to evaluate the speech quality assessment using Deep Neural Networks (DNN) models. As example, in (JUNIOR; ROSA; RODRIGUEZ, 2018) is presented a non-reference model based on a hybrid Restricted Boltzmann Machine (RBM) to evaluate speech quality that overcame the ITU-T Recommendation P.563 in a built database. In (SONI; PATIL, 2016) is proposed a speech quality assessment model based on deep autoencoder to extract features from spectrum of speech signals, it is compared with the ITU-T Recommendation P.563 and presented more accurate results. A new non-reference model, based on Deep Belief Networks (DBN), is presented in (AFFONSO; ROSA; RODRÍGUEZ, 2018), its performance was evaluated in comparison with the ITU-T Recommendation P.563 in a built database, presenting high accuracy on speech quality classification. In (FU; TSAO; HWANG, et al., 2018) is presented a non-intrusive speech quality assessment model based on Bidirectional Long Short-Term memory (BLSTM) at frame-level, in that work was observed scores with high correlation to PESQ.

1.1 MOTIVATION

As already exposed, the problems involving the applicability of subjective listening tests naturally demand efforts in the development of automatic speech quality assessment methodologies. This is the main motivation to the creation of new approaches that must be capable of addressing the increasing demand of new technologies.

Besides the problematic of subjective listening tests, there exists a remarkable necessity of non-reference methodologies. Actually, state of the art approaches such as PESQ and POLQA – and many others – need a reference signal to evaluate the degradations present in the processed signal. However, a reference signal is not always available, making difficult the applicability of full-reference approaches.

Even for well standardized automatic speech quality assessment techniques, the coverage of all subjective characteristics of quality, experimented by a human being, is not complete. Taking as an example PESQ: this is one of the most popular methodologies in the area, being cited in many works (ROZHON et al., 2016), (KIM, 2004), (CÔTÉ et al., 2010), (LAPIDUS; SHALLOM, 2010). Even so, PESQ is considered as having bad accuracy at specific test conditions (QIAO; SUN; IFEACHOR, 2008). This depicts the complexity involved on modeling the speech quality assessment experimented by average subjects, being necessary the development of models with higher generalization performance attending the whole complexity of such subjective problems.

The need of models with good generalization were partially supplied by the development of Neural Networks (NN) techniques. These types of methodologies are well standardized models, inherently created to have good generalization performance, being able to model various types of problems (ABIODUN et al., 2018). In the speech quality assessment area, the use of NN methodologies brought relevant improvements to solve some barriers not solved by hand-crafted solutions. However, as stated before, NN are models highly based on the brain operation; they mimic the minimal brain units (neurons) and communication paths (synapses) to get a highly generalization model (ABIODUN et al., 2018). Since NN models has been used to solve a multitude of problems, many times presenting low generalization error, it is not proved that they are the best choice when modeling sensory perception, as will be shown next.

Currently, with the advances in the DL area, new models have been created, presenting lower error rates in the modeling of subjective tasks. More specifically, the application of DL approaches to solve classic problems in the area of signal processing (HE; ZHANG, et al., 2016), (CHIU et al., 2018). Many of the improvements achieved are due to the use of new types of architectures. As example, the use of Convolutional Neural Network (CNN) layers as a feature extractors contributed to many improvements on signal processing tasks (ABDEL-HAMID et al., 2014), (ANDERSEN et al., 2018), (PARK; LEE, 2017). In speech related tasks, the use of CNNs has proved to be very important to get highly accuracy, in some cases outperforming human results (ABDEL-HAMID et al., 2014).

With that was exposed in this section, it becomes clear the necessity of study and development of automatic speech quality assessment methodologies, that are non-reference based, with highly capacity for modeling the underlying speech features and to be more faithful to the quality score of an averaged subject.

1.2 OBJECTIVES

This dissertation has as primary objective to analyze the use of CNN as a framework of feature extraction for automatic speech quality assessment. It is expected to be possible to prove the efficiency of convolutional layers for modeling the underlying speech quality aspects. In this scenario, it is crucial the development and evaluation of architectures using convolutional layers for feature extraction. With the use of such architectures, it is expected to reach similar, or even better, accuracy to the current state of the art in speech quality assessment.

1.3 DISSERTATION STRUCTURE

In the second chapter, the main aspects of speech quality are discussed, as well as the most important methodologies that make up the state of the art. In the third chapter, the main components of a CNN model are explained, besides the classical CNN architectures are commented with focus on their design. In the next chapter, the proposed model structure is presented. In Chapter 5, the experiments performed using a classical database, as well as a database built in this work, are presented. Still in the fifth chapter, it is presented the result analysis with the focus on comparing the proposed model with the state of the art methodologies. The last chapter brings the conclusions and future works.

2 SPEECH QUALITY

Differently of intelligibility, the speech quality is concerned on extra linguistic concepts. Its assessment is an important role to the planning of networks and speech processing systems. Because of this, subjective listening tests were primarily developed, using subjects to evaluate corrupted samples. To overcome the problems involving subjective tests, automatic methodologies were developed.

In this chapter, a plan definition about the speech quality term is presented. The main factors that affect it are presented and discussed. The subjective tests are briefly commented. The final part of this chapter present some of the best known instrumental speech quality assessment methodologies.

2.1 DEFINITION

Speech quality is a complex topic in speech processing area, more because of the human subjectivity involved. In this sense, not only typical degradations are analyzed, but many aspects of the emotional subject state need to be considered when addressing quality. A resumed definition can be stated as: speech quality is the result of a perception process in which a speech signal, perceived by a human being, is judged (rated) with respect to an expected internal reference. The representation schematic for the speech quality judgment process is depicted in Fig. 1.

Firstly the perception process involves the sound wave that reaches a human ear (sound event); in the ear, the wave is perceived by the hearing system and then decoded in quality features (perceptual event). The expected internal references are the desired features, generated from an idealized version of the speech signal, created at the communication situation by the listener. In situations where the listener has experience in the communication situation, it is possible that an internal reference schema already exists; besides that, it is even possible that the listener anticipates perception in a limited way. The construction process of the expected reference can be altered by factors as context, situation, motivation, mood, etc. The quality event, in a resumed view, is generated from the comparison between the reference features and the features obtained from the speech sample. The description step is a final process of encoding the quality judgment event into a scale known by other humans, serving as criteria of measure.

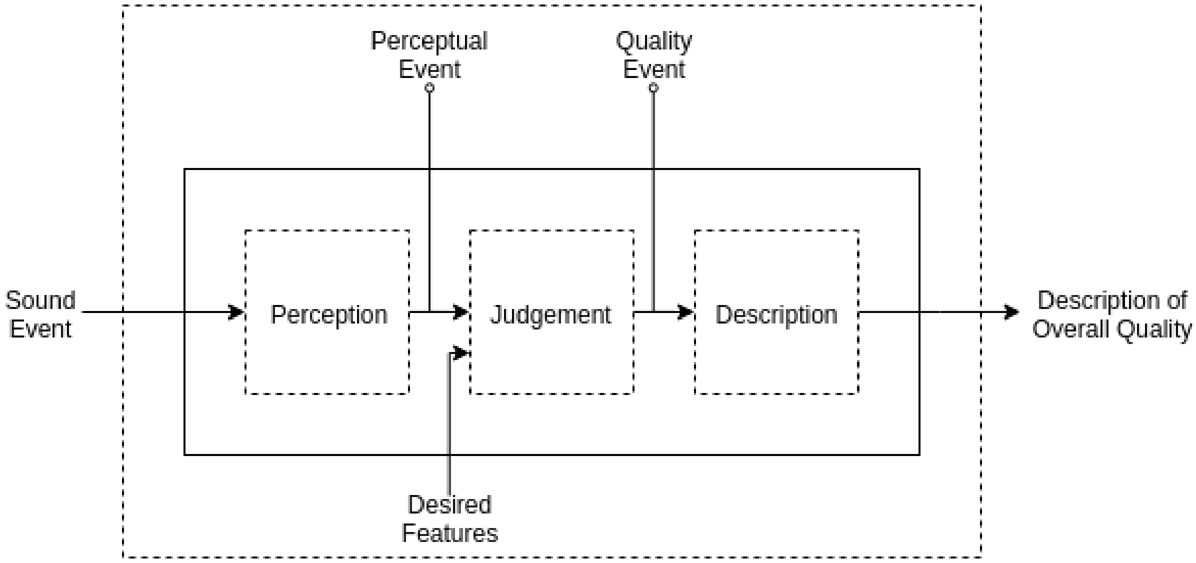


Figure 1 – Representation schematic for the speech quality judgment by a subject.

Source: Generated by the author.

More details about speech quality perception can be found in (RAAKE, 2006), (BENESTY; SONDHI; HUANG, 2008) and (JEKOSCH, 2005).

2.2 QUALITY FACTORS

The most part of the complexity involved to assess automatic speech quality is associated to the countless quantity of factors that can influence perception, whose are ample and highly depends on the application. Due to the degradation that affects speech samples, low quality features are generated (result of perceptual event in Fig. 1) which, when compared with the desired features, in the judgment process, result in low score ratings. Besides, some impairments can affect the generation of the desired reference signal (desired features in Fig. 1); in this case, more subjective effects are present. Actually, a big challenge in the area is the reliable speech quality assessment when all these factors are added together.

2.2.1 Degradation

In this section, a general view of the most important factors that affect speech quality, at the telephone and VoIP scenarios, are discussed.

2.2.1.1 Background Noise

Noise had a great highlight in old analog circuit-switched telephony and regained importance with the arise of mobile communication. The most common type of noise, encountered in telecommunications, is the background noise. This kind of degradation occurs as a background element in a conversation scenario, being added to speech signal at the sender side, during a call for example. At the receiver side, the resultant sound is a composition of speech and noise, and possibly other impairments, that together form the input for a perceptive instance. The background noise presents two factors necessary for the perception scenario: degradation and context. In the first one, the noise is characterized as an impairment element; in this view, it impacts the speech signal with masking, possibly removing important speech components and compromising the intelligibility. In scenarios with low background noise, the impact in quality perception is typically low (RAAKE, 2006); even in scenarios with a louder noise, adaptive users attitudes can compensate the impairments by changing speaking behavior (MÖLLER, 2005). The second factor of perception, played by the background noise impairment, is the context information provided for the listener. The contextual information is crucial to the perception analysis employed by the listener, it is used to form the reference internal model.

2.2.1.2 Codecs

The use of codecs to reduce costs with bandwidth in transmission are typically addressed by means of perceivable degradation of speech quality. When transmitted across different types of networks, the speech signal can be coded multiple times (tandeming), and it can be submitted to different types of coding. In this way, different types of codec degradations can be applied. More details about it can be found in (RAAKE, 2006).

2.2.1.3 Delay

Delay can be originated from diverse sources, as encoders and error correction schemes; it can reduce the communicability, degrading the overall quality. Sometimes, the effects of delay can be masked: users can associate the effects of delays to the communication partner in the call. The effects of delay are prominent to the users with a more experience on delay-type degradation.

2.2.1.4 Jitter

Jitter can be defined as the variation in packet delay time, affecting the order of the packet sequence in transmission events. It is caused by the packet routing over different network paths and the asynchronous characteristics of the networks. To compensate the effects of jitter, buffers are used, at the receiver side, which store packets during a time before playback. The time used to maintain the packets in the jitter buffer can address an important role on the packet loss since the latest packets are discarded. This way similar rules applied to delay and packet loss can be addressed to the jitter control apparatus (BENESTY; SONDHI; HUANG, 2008).

2.2.1.5 Packet Loss

The most characteristic degradation affecting VoIP systems is the packet loss. This type of data loss typically occurs at moments of network congestion, when the packets are delayed to playback, being discarded by jitter buffers, or in case of bit errors. The affected speech quality is highly dependent of some aspects such as the loss distribution, packet size and packet loss recovery methods. The use of large packet sizes can overhead the network congestion contributing to the delay. The packet loss recovery methods can, in some instance, recover, or estimate, the lost packets.

Techniques as Packet Loss Concealment (PLC) compensates lost packets at the receiver side; they can use approaches as insertion of silence, interpolation or frame repetition, sometimes warping effects can affect the result signal. Besides, it can occurs the phenomena called Clock Drift, in which there is loss of synchronization between transmitter and receiver, resulting in packet loss and signal deformation (MELVIN; MURPHY, 2002).

Techniques as PLC are not always a good choice since the estimated data inserted is not the real version, causing degradation on the perceived quality. Other techniques such as Forward Error Correction (FEC) uses duplicated versions of the coded speech data to correct the loss events. The use of an original duplicate version of the data can restore the entire loss section with the cost of more network congestion, and so the overall network delay can be add to the network. More detailed descriptions of packet loss distortions can be found in (BENESTY; SONDHI; HUANG, 2008).

2.2.1.6 Echo

Echo effects can be perceived by a speaker because of reflections at some point in the speech path (from mouth of the talker to the ear of the listener). Nowadays, echo effects are more prominent when using microphones; the talk interface picking up the speech sounds from the hearing part, in a acoustic coupling between loudspeaker and microphone. Due to echo, difficulties in talking may arise; the users can be confused with the real speech being talked. However, masking effects can conduct to less annoying in periods of double talk. The use of Echo Cancelers to reduce effects of talker echo can generate new distortions in the speech signal, such as residual echo, nonlinear distortions, clipping and delay.

Other distortions such as the residual use of noise suppression techniques, bit errors, clipping, loudness loss and more details about quality elements can found in (RAAKE, 2006) and (BENESTY; SONDHI; HUANG, 2008).

2.2.2 Subjective Effects

As a perception matter, the speech quality assessment is affected by many subjective effects, such as mood and context. These effects vary from person to person and so far have been poorly explored in the literature. However, they play important rules in the speech quality assessment process and are discussed next.

Context is one of the most important elements in a conversation task. It acts as a primordial reference to the message being transmitted. So, in a conversation scenario, it is possible to capture clues to identify the context. The main contextual clues are presented in the message content. However, sometimes, just that does not complete the contextual information, so other sources are claimed. As stated before, the presence of background noise provides contextual clues for the listener. These clues serve for adaptation to the perceptual scenario by the listener. In this way, the listener can relate contextual information with the speech message to get better speech understanding.

Other factors have few mentions in literature but are relevant in more specific contexts. As example, the financial cost can alter the expected conversation quality when a user is paying for the call, or service. Another effect can be defined as the experience of the user with the service in use: some users can get used to the quality of a service and form a expected reference quality, while new users can, based on other service experiences, to analyze differently a new service quality. At last, the effects as humor and motivation can play an important relationship with the quality assessment of a service but are further subjective and depend on more specific details. More subjective effects affecting the speech quality perception can be found in (RAAKE, 2006), (MÖLLER, 2000) and (JAUK et al., 2018).

2.2.3 Mix of effects

In the current telecommunication scenario, a composition of networks allows that a signal can travel across multiple types of channels. Thus, the path a speech signal can travel from a talker source until it reaches the listener can be composed by a combination of telephonic network and a packet based network, for example. This architecture of communication networks applies multiple types of impairments (as those listed before) in the speech signal, being perceived in the listener endpoint.

The great challenge when addressing speech quality assessment is the reliable evaluation of the mentioned degradations when they appear together, mixed in a speech signal. The problem becomes even more complex when the impairments interact on a perceptual level. For example, when packet loss and echo are mixed, the subject perceived quality is affected. In the literature, few studies on combined degradations are reported until the production of this Dissertation. The main studies with such analysis is reported in an additivity assumption of the E-model (MÖLLER, 2000).

2.3 MEASUREMENT

As mentioned before, the quality is a perceptual aspect, being internal to each subject. Because of this, the more reliable type of speech quality measure is conducted by the use of auditory tests. However, as appointed out in the first chapter, this type of measurement is time-consuming and costly, restricting their use in real-time applications. The use of instrumental measurements has been proved to provide valid quality predictions and so are the principal way to assess speech quality. This section mentions the main auditory methods recommended in (ITU-T, 2016). Besides, it presents an overview of the main instrumental methodologies.

Auditory methods are typically carried out by the use of listening only tests. In summary, in these methods, subjects are oriented to listen corrupted speech samples and to associate to each one an integral quality rate. A typical test used in telecommunications is the Absolute Categorical Rate (ACR), in this test methodology, the user associates the speech signal to a category rate (ITU–T, 1996a). Different scales are available, but the typical scale used to rate a speech signal is the five point quality scale, in which: 1 (bad), 2 (poor), 3 (fair), 4 (good) and 5 (excellent). The mean over the ratings, for all listeners, is used as the overall quality score being defined as Mean Opinion Score (MOS).

Speech quality assessment methodologies can be divided into two sets: parameter based and signal based. Those in the first set perform a communication channel parameter modeling (or system under test) to assess speech quality. The second set deals with quality assessment using the processed signal.

Another type of classification concerns the use of the original signal as a reference. In case both of original and processed signals are used, the quality assessment methodology is classified as intrusive (or full-reference); if only the processed signal is used the methodology is classified as non-intrusive (or non-reference).

In this section, some of the most renowned methodologies of the area are presented; some of them has become reference in the standardization bodies.

2.3.1 Full-reference methodologies

Full-reference measures form a subset of the quality assessment methodologies. They are characterized by using the original signal and the signal processed by the system under test to perform the assessment. The great advantage of using a reference signal is the possibility of comparing the two versions of the signal, since when processing a signal with a variety of distortions, the original signal serves as a pure version.

2.3.1.1 PESQ

The Perceptual Evaluation of Speech Quality (PESQ) methodology is a tool developed by ITU-T, being the Recommendation P.862 for voice quality assessment (ITU–T, 2005). It was created to complement the Perceptual Speech Quality Measure (PSQM), ITU-T recommendation P.861 (ITU-T, 1998), unable to make predictions under conditions of packet loss, background noise and coding distortions.

The first version of the PESQ model focused on narrowband quality assessments, covering ranges from 100 to 3500 Hz. However, a new standardization was carried out by designing an updated version of PESQ capable of covering distortion in bandwidths up to 7 kHz (ITU–T, 2017).

Although the PESQ model can operate in the aforementioned modes, its use is generally recommended in short-band ranges, as the tests performed have a higher correlation with the test samples (ITU–T, 2005). The PESQ methodology covers a wide scope of distortions and it is one of the reference algorithms for intrusive quality assessment actually.

2.3.1.2 POLQA

With the arise of voice services, introducing new types of broadband frequency band distortion, the PESQ model has begun to fail to deliver good results. Thus, a new standardization was initiated by the ITU-T Study Group 12 to replace the PESQ model. Three candidate algorithms were selected and integrated into a single model, resulting in the Perceptual Objective Listening Quality Assessment (POLQA), a quality assessment standard that became the P.863 recommendation, specified in (ITU–T, 2011a) and subsequently updated in (ITU–T, 2011b). POLQA is a methodology designed for objective quality prediction capable of processing in short-band (up to 3.5 kHz), broadband (up to 7 kHz), super-broadband (up to 14 kHz) and super-wideband (48 kHz).

2.3.1.3 VISQOL

With the adoption of the VOIP standard, a range of problems related to speech quality factors were introduced. This way, it was necessary to adapt the quality assessment models to the new scenario, taking into account some types of distortions commonly found in VoIP: Delay, Jitter, Warping and Clock drift.

Typically, more than one type of distortion is encountered when using VOIP architecture. Thus, the models should be able to detect not only one, but most of the degradations that exist when using the VOIP standard, more details on quality factors can be found in (TORAL-CRUZ et al., 2011).

Primarily designed to solve quality problems associated with the VOIP standard, the Virtual Speech Quality Objective Listener (ViSQOL) (HINES et al., 2015) has emerged as a methodology that assesses speech quality using a spectrum-time measurement of similarity. Its performance competes with POLQA, being an effective alternative in predicting voice quality in VOIP scenarios.

2.3.2 Non-reference methodologies

In several situations, it is impossible to obtain a reference signal for quality assessment. For the most real-time applications is not possible to obtain the original signal from the source, being necessary to perform assessment on just one version of the signal, or on parameters of the communication channel. Thus, quality assessment methodologies have emerged that do not require a reference signal, called non-reference methodologies.

2.3.2.1 ANIQUE

The Auditory Non-Intrusive Quality Estimation (ANIQUE) model employs a non-intrusive approach to speech quality assessment by exploring the sensitivity of the human ear to temporal envelope variation, defined in (KIM, 2005). In this approach, some steps are common to other quality assessment methodologies, such as filtering, psychoacoustic modeling and degradation estimation.

2.3.2.2 ITU-T Recommendation P.563

The ITU-T Recommendation P.563 was the first non-intrusive recommendation for measuring applications, taking the full spectrum of distortions found in public switched telephone networks, capable of assessing quality on the MOS-LQO scale (ITU–T, 2004). This methodology addresses speech quality assessment by employing a linear combination of speech parameters. Various speech parameters are used, some of them being speech descriptors, noise description and voice characterization.

3 CONVOLUTIONAL NEURAL NETWORKS

Artificial neural networks have emerged as models that attempts to mimic the processing in human brain and have become popular over the years, being applied to a host of practical problems. However, the use of such models still encounter some problems that limit their applicability, especially in the context of image processing. As example, the use of Fully-connected Neural Network (FNN) models in tasks as character recognition is accomplished with some limitations. A familiar problem is caused because the dimensions of the input images can be large, increasing the amount of the network training parameters. As example, using a FNN model with input samples of 28x28 pixels, each neuron of the input layer would have a total of 784 connections, one for each pixel of the image (Fig. 2). In this case, in a network of 100 neurons it would be necessary to maintain 78,400 trainable weights for the first layer only. Therefore, a considerable amount of memory capable of maintaining network weights should be available.

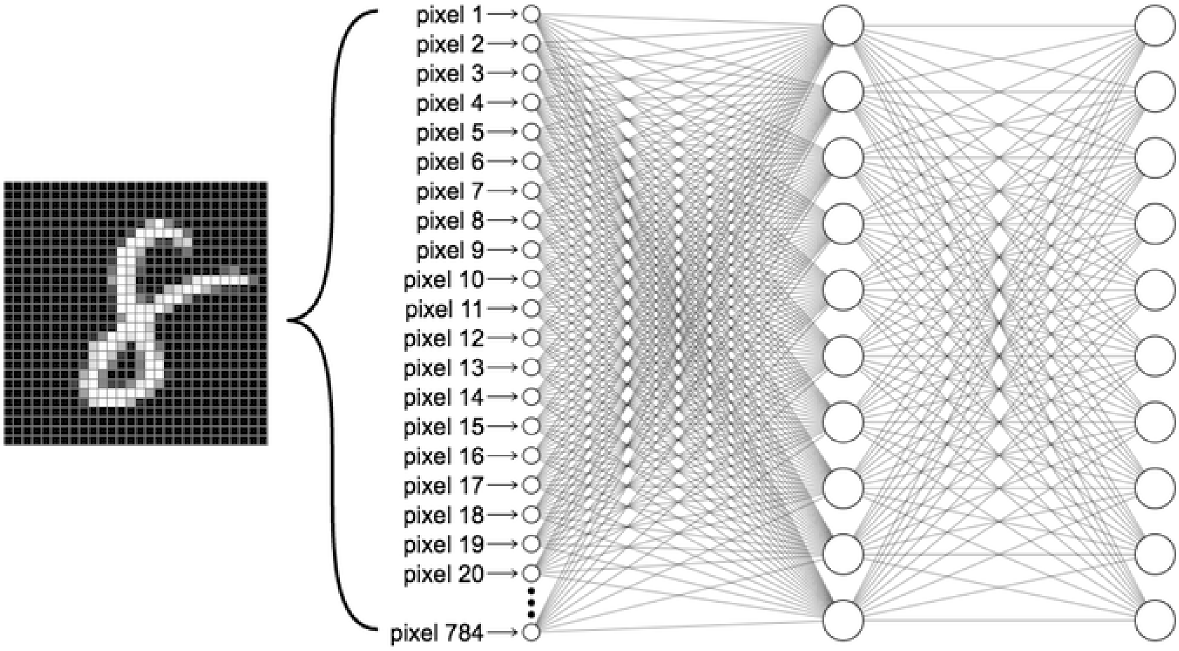


Figure 2 – Input scheme of a neural network architecture for digit image classification.

Source: ml4a.github.io/ml4a/looking_inside_neural_nets/
0

Other factors as the sensitivity to translation, deformation and partial clipping in images with objects on the scene limits their use as input for FNN models. To overpass these problems, pre-processing steps are performed over the input images such as: normalization and centering. Another approach to mitigate the mentioned problems is the insertion of distorted samples to the training set, generating new training images (in the ML area this task is known as data augmentation). Although this approach be useful, by increasing the training set variability generates other problems such as the need for more complex networks, and so more neurons with demands for more connections, the huge number of parameters would quickly lead the model overfitting. In addition, due to the translation of input images, the FNN weights would learn similar patterns, causing redundancy at the learned features. Lastly, FNN models have indifference to the topology of the inputs because the neurons of the input layer connect to all pixels in the image. Therefore, local characteristics such as edges, lines, and corners, and their relative positions in the image, are ignored by the model.

The main studies on the cortex date back to the 1960s (HUBEL; WIESEL, 1959), (HUBEL; WEISEL, 1968). The researchers discovered the presence of local receptive fields in which portions of neurons react only to visual stimulus, located in a limited region of the visual field. Besides that, it was noted that other type of neurons have larger receptive fields reacting to more complex patterns that are combinations of the lower-level patterns. These studies have inspired some models that evolved into what it is called today as Convolutional Neural Network (CNN).

With the ability to overcome many of the problems presented for the use of FNNs arise the CNN architectures. These are specialized neural networks that use convolution operations for processing data with a grid-like topology (GOODFELLOW; BENGIO; COURVILLE, 2016). The rise of these networks represented a milestone in the history of some research fields such as: object recognition, image segmentation and speech recognition.

3.1 GENERAL ARCHITECTURE OF A CNN

In the last few years, motivated by the intensive arise of studies in the DL area, the CNN evolved to multiple types and forms. Although the types we have today, the common architecture of a CNN model remained limited to the use of convolutional Layers, pooling Layers, and fully-Connected Layers.

3.1.1 Input

The CNN architecture make the explicit assumption that the inputs are volumes and so its units are treated as arranged in 3 dimensions: width, height, depth. Besides the input, each convolutional layer accepts as input a 3D volume and transforms it to an output 3D volume through a differentiable function.

3.1.2 Convolutional layer

The main block of an CNN architecture is the convolutional layer. This layer is used to extract features by using learnable filters (kernels). Each filter is a small 3D volume, they typically cover a small region over the height and width dimensions (referred here as planar dimensions), while covering full depth dimension of the input volume. The use of small region sizes over the planar dimensions is called as the sparse connectivity property (GOODFELLOW; BENGIO; COURVILLE, 2016). In the FNN architectures each neuron (unit) in a layer is connected with all other units of a preceding layer, this approach makes the units extract more global features, based on the units of a preceding layer. The CNN filters instead cover a small region aiming to extract localized features as edges, corners, etc. The sparse connectivity property leads the CNN models to use fewer parameters and consequently less resources, such as memory.

To the extension of the local connectivity for a filter is referred as the receptive field, it is defined as the planar region in which the filter acts. In similarity of the receptive fields of the cats and monkeys in the early studies of the visual cortex, this type of visual structure acts by distinguishing local patterns in a image. The connections between receptive fields of different layers can visualized in the Fig.3.

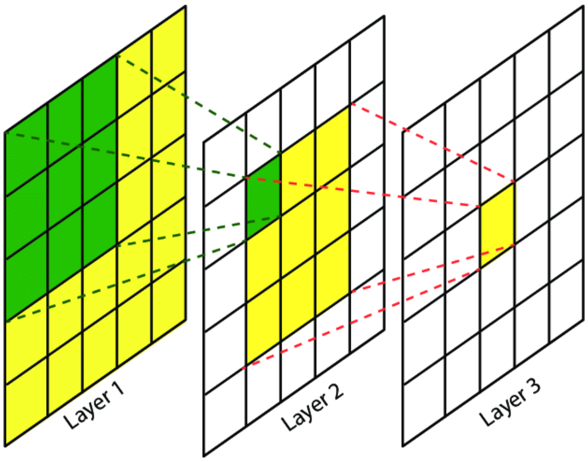


Figure 3 – Convolutional receptive fields connections.

Source: Image taken from (LIN; SHI; ZOU, 2017) Receptive Fields of a convolutional layer.

In the figure, it can be seen that each receptive field of an unit covers an area representing a region of the input volume. Since the depth dimension is entirely covered by a CNN filter, in the Fig.3 the depth dimension is omitted.

Another important property is defined as the parameter sharing; in the case of CNN layers this property defines that the units in a given layer share the same filter weights. So, each unit of one layer in the Fig.3 shares the filter weights related to that layer. This ensure that, in comparison with FNN layers, less weights are used to train a convolutional layer.

For each unit in the CNN layer, it is performed a dot product between the filter weights and that others of the intersection region within the input volume. Because the CNN units share the filter weights, the convolutional layer operates similar to a slide function, applying filters functions (weights) over the input volume. The mentioned slide is similar to a convolution operation because the same function (weights) is convolved over the entire input volume.

The result of the sliding operation performed by one filter is a 2-dimensional activation map, also called of feature map. The generated activation map will have high values for the locations where the filter weights are similar to the values of the region in the input volume because the operation of dot product. Therefore, the operation of convolution acts as a pattern search, in which the filter is the pattern to be found in the input volume. The activation maps, produced by the filters in the same convolutional layer, are stacked together forming the output volume. In the Fig.4 is shown the input and the output feature maps resulting from a convolutional operation.

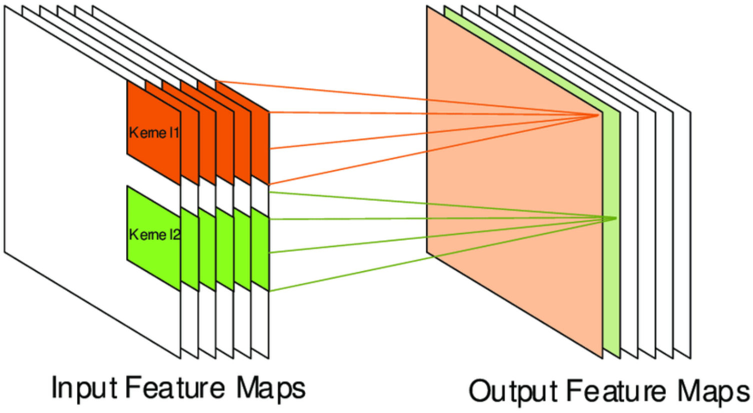


Figure 4 – Feature maps of a convolutional layer.

Source: Image taken from (VÉSTIAS, 2019) Feature maps of a convolutional layer.

In the image can be seen the result of the convolutional operation by using two different filters (kernels). On the left, the input feature maps are shown colored by the instantaneous of two filters locations. On the right, the output feature maps generated by the convolutional operation using the two filters are shown in highlight.

The output volume dimensions of a convolutional layer are defined by the use of three hyperparameters. The first parameter is the number of filters: this parameter defines the number of feature maps in the output volume, and consequently set its depth dimension. Another parameter is related to the slide size used to convolve the filter with the input volume through the planar dimensions: this size is called stride, and in some part it defines the planar dimensions of the output volume. A typical convolution operation is accomplished by centering the filter receptive field to regions of the input volume. If this centering is accomplished at the edge pixels, some units would be behind the extensions of the input volume. To solve this problem, it can be used the padding operation in which the input volume is padded with some values (zero in many cases) around the border. Considering the use of only square filters in a convolutional layer, the mentioned parameters can be used in a equation to calculating the planar dimensions of the output volume (O), as it can be seen in Eq.3.1.

$$O = \frac{(W - F + 2P)}{S + 1} \tag{3.1}$$

where W is the size of the planar dimensions of a input volume, F is the size of the receptive fields, P is the size of the padding and S the stride.

A convolutional layer can have countless filters; the greater the number of filters, the greater the diversity of characteristics extracted from the input volume. Because the coupling between convolutional layers, the higher layers focus on the resulting activation maps from the lower layers. This process generates a hierarchical topology in which complex patterns are build from the low level patterns. This is accomplished because the CNN filters learns the best filters and how to combine them. This is an important property that assign to the CNN architectures the power to extract a structured set of features.

3.1.3 Pooling layer

To reduce the planar dimensions, size pooling layers can be inserted in a typical CNN architecture. This type of layer applies an operation similar to a downsample by applying a transform function over the input volume. Similar to what was presented for filters of convolutional layers, the pooling filters operates by slicing over the input volume, applying the downsample function, but the application of the downsample function is limited for each feature map instead. As it can be seen in Fig.5, the pooling layer applies the downsampling function over the input volume; the result volume has planar dimensions divided by a factor of two while the depth dimension is keep the same.

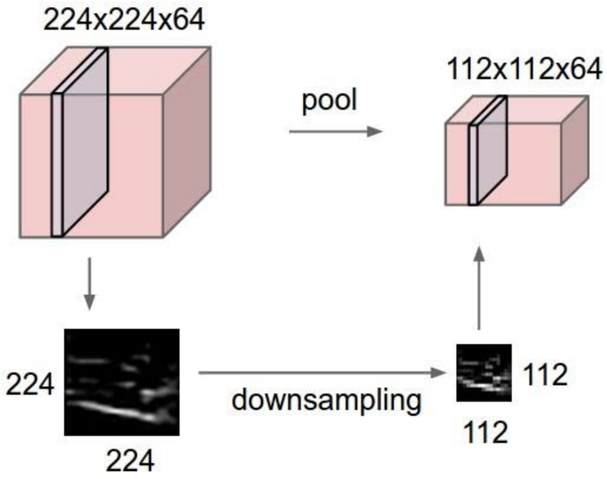


Figure 5 – The result volume of applying the pooling layer.

Source: Image taken from CS231n Convolutional Neural Networks for Visual Recognition Course

The downsample functions used in pooling layers are used to aggregate the values of a planar region in a comprehensive way; in Fig.6, it can be seen the application of a max pooling function (that take only the maximum value of input) over an input feature map.

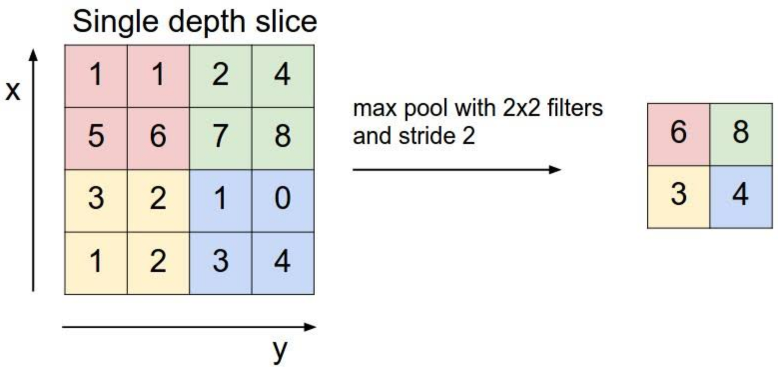


Figure 6 – Pooling layer application using a max pooling function.

Source: Image taken from CS231n Convolutional Neural Networks for Visual Recognition Course

The use of pooling layers are beneficial to avoid the overfitting since they reduce the number of parameters of the network. Besides, small variations in the features of the input volume can be addressed properly by the pooling layers. They reduce the total computation used by the training step of an convolutional model. The application of a pooling layer over one feature map can be seen in the Fig6.

3.1.4 Fully-connected layer

In typical CNN architectures, the final layers are constituted of fully-connected layers. These layers are used to aggregate the feature maps generated by the preceding convolutional layers. In problems as classification, the output of a fully-connected layer is the most probable classes for the data training. In problems of regression, in which a value must be generated as response, the output fully-connected layers has only one output at the final layer.

A complete CNN architecture can be seen in Fig.7. This is a typically CNN architecture, involving the input image, the convolutional layers and the final fully-connected layers.

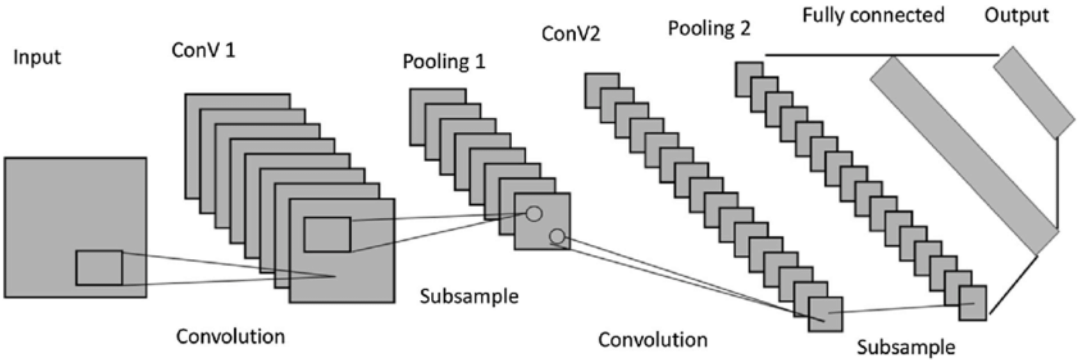


Figure 7 – A complete CNN architecture with the input, convolutional layers, pooling layers, fully-connected layers and the output.

Source: Image taken from (WEI et al., 2019)CS231n Convolutional Neural Networks for Visual Recognition Course

3.2 IMAGENET

The ImageNet project was created to bring together a massive collection of labeled images available to assist research and development in the area of object detection and classification. Since its presentation in 2009 in the Conference on Computational Vision and Pattern Recognition (CVPR) at Princeton University, people around the world have collaborated to label this mass of data.

Since 2010, there has been an annual challenge, the ImageNet Large-Scale Visual Recognition Challenge (ILSVRC), in which teams run their models on the data mass and compete for the highest accuracy in image recognition tasks. In this challenge 1000 categories are used with approximately 1000 images per category. Since the competition began, there has been a rapid decline in error rate due to the use of deep convolutional network architectures. Several convolutional network architectures have emerged and gained prominence and are used in other projects around the world. Some of these architectures and their respective error rates in the ImageNet challenge are presented in Fig.8.

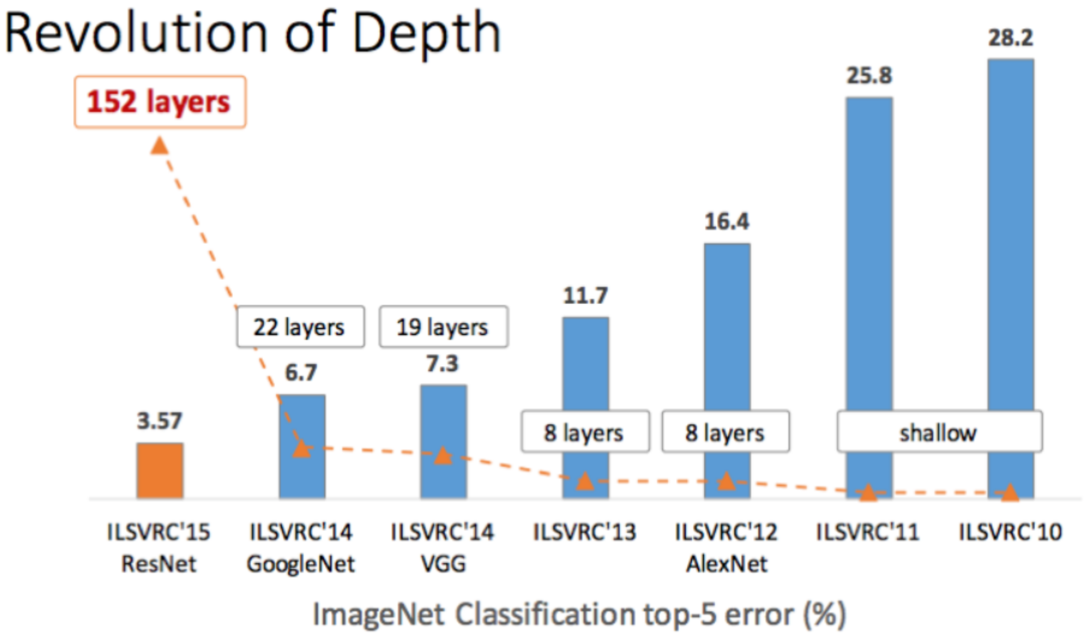


Figure 8 – Error rate and name of the winning architecture from ImageNet Challenge from 2010 to 2015.

Source: Image taken from the Kaiming He presentation, Deep Residual Learning for Image Recognition.

In 2015, a milestone occurred in the history of deep architectures: an architecture achieved an error rate lower than the average human object classification capability with the ResNet architecture. The average human error was approximately 5%, while the new architecture achieved an error rate of 3.57%.

3.3 CLASSICAL ARCHITECTURES

Since the advent of the ImageNet challenge, several network architectures have been created, which has provided a lot of innovation in machine learning. Another contribution was the creation of new learning models, bringing new architectures and models that could be used in various problems.

In this section, some of the most important architectures for advancing the area are presented. The focus will be in describe what innovations in the learning models have been introduced with each new architecture.

3.3.1 Lenet-5

The work presented in (LECUN et al., 1998) was an important breakthrough to the pattern recognition area since it initiate a paradigm breaking of the traditional design. Before the popularization of the convolutional neural networks the pattern recognition area was dominated by approaches with the architecture design following the schema of Fig.9.

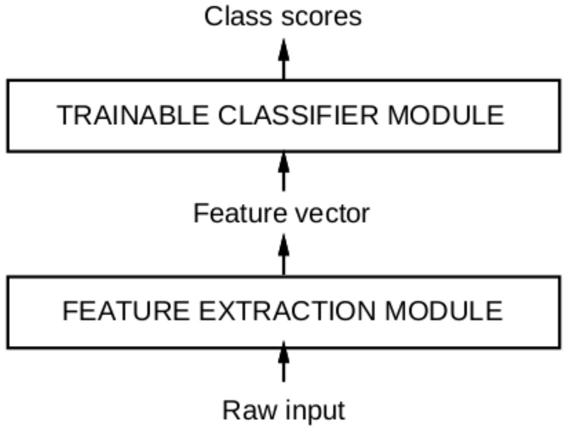


Figure 9 – Traditional Pattern Recognition architecture diagram.

Source: Image taken from (LECUN et al., 1998)

The Feature Extractor Module was typically a heuristic algorithm using approaches specific to the task. In the Trainable Classifier Module the neural network architecture was used. One of the problems with this architecture is in its recognition accuracy, it was limited by the ability of the feature extractor to come up with good features. Therefore, there was a necessity of development of trainable approaches to be used as feature extractor modules.

An alternative at the epoch was the use of fully connected feed-forward networks been fed with raw inputs directly. In (LECUN et al., 1998) is presented as example the use of such networks for tasks of character recognition. Although the arise of such architecture its use come up with some problems such as:

- The size of images is typically large increasing the total number of the network trainable weights.
- The complete no built-in invariance to input translations, or local distortions.

- The topology of the input is ignored.

The first convolutional network which has presented good performance in character image classification was known as Lenet-5 (LECUN et al., 1998). Its architecture is shown in Fig.10. It was one of the first networks to use convolutional architecture, being able to obtain highly accurate classification results.

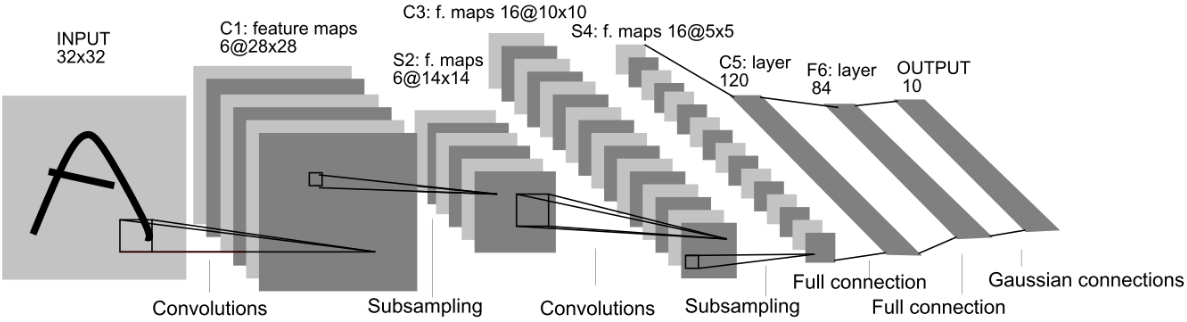


Figure 10 – Lenet-5 architecture.

Source: Image taken from (LECUN et al., 1998)

Fig.10 shows a block diagram of a typical CNN architecture. The inputs are images of characters zero-padded and normalized in size. The transformations applied to the original images was useful to force the centralization of the receptive fields of the highest-level feature detector.

Convolutional layers are presented with the *CN* label, where the *N* is the layer index. In the *CN* layers, typical convolutional layers are presented, used as the feature extractors of the character image. In the $C1$ and $C5$, classical connection between feature maps and receptive fields of the next layer are used. However, for the $C3$ layer, it is used a different approach. In that layer, different sub-sets of the feature maps from $S2$ are connected to the $C3$ receptive fields. This approach was mainly applied to force a break of the symmetry in the network, forcing the different kernels to learn different sets of features. The last convolutional layer applies receptive fields of the same spatial size of the feature maps from the $S4$ layer, generating feature maps of 1x1 dimensions. In the Lenet-5 diagram, the feature maps and the receptive fields are shown in each layer labelled as $F@KxK$ where the F label represents the number of feature maps and the K is used to represent the dimensions of receptive fields.

The pooling layers are presented with the *SN* label, such layers were called Subsampling layers at epoch; because of this, there is the use of labels with *S* instead of *P*. In such layers, it is used a specific type of averaged pooling, the sequence of operations are: each neuron computes the mean of its inputs, then multiplies the result by a learnable coefficient, adds a learnable bias and, finally, it applies an activation function. The pooling layers are mainly used to reduce the precision in which the position of different features are encoded in a feature map. It was noted that this approach could be beneficial to recognition of different images of characters with variable deformation.

The use of successive convolutional layers and pooling layers, each one applying a processing stage in which occurs the increasing of the feature maps and decreasing of the spatial dimension, was adopted with the expectation of achieve a large degree of invariance to geometric transformations.

The $F6$ layer is a typical fully-connected layer with the activation hyperbolic tangent. The output layer is composed of Euclidean Radial Basis Function units (RBF), one for each class. Instead of applying the typically operation of dot product between input and weights, the output layer computes an Euclidean distance between its input and its weights. This can be interpreted as a penalty measuring the fit between input and the weights representing a model of the desired class.

3.3.2 AlexNet

One of the first convolutional networks to perform well in the ILSVRC challenge is popularly known as AlexNet in honor of one of the creators (KRIZHEVSKY; SUTSKEVER; HINTON, 2012). The network architecture is presented in Fig.11.

3.3.2.1 Activation Function

For the training of the model AlexNet, it was used the activation function known as Rectified Linear Units (ReLU). The choice was made because, as shown in (NAIR; HINTON, 2010), some activation functions, such as logistic sigmoid and hyperbolic tangent, used for convolutional network training, take longer time, on average, to converge.

3.3.2.2 Training

Given the computational capabilities of that time, the AlexNet network was trained using a GTX 580 card with 3GB of memory. With low memory available and due to the size of the network it was necessary to use two cards working together to perform the network training.

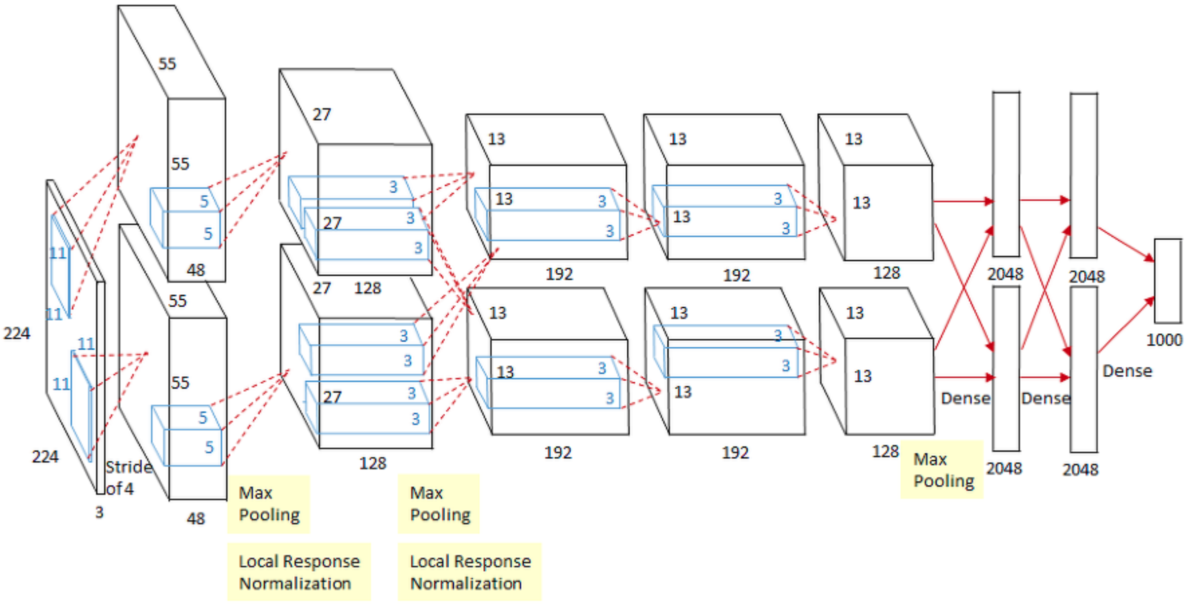


Figure 11 – AlexNet architecture.

Source: Image taken from bskyvision.com/421

The approach used was to couple the two networks only a few layers, making the convolutional input layer, the third layer and the fully connected layers share all the weights of the previous layers. The network sharing and weight sharing scheme is shown in Fig.11. This selected layer sharing and weight sharing scheme provided a 1.7% reduction in top-1 error rate (the best result at that time) and 1.2% on top-5 error rate (top five results at that time).

3.3.2.3 Local Response Normalization

Even using an activation function that does not present saturation problems, the authors proposed a scheme that makes the resulting model generalize more. The developed normalization scheme is defined in the Eq3.2.

$$b^i_{x,y} = a^i_{x,y} \Big/ \left(\kappa + \sum_{j=\max(0,i-n/2)}^{\min(N-1,i+n/2)} \left(a^i_{x,y}\right)^2 \right)^{\beta} \tag{3.2}$$

where:

- $b^i_{x,y}$ is the normalized activation response
- $a^i_{x,y}$ is the activation of a neuron by applying kernel i to position (x, y) and then applying the ReLU function
- N is the quantity of kernels

- n is the quantity of adjacent kernels
- κ, α and β are constants

In AlexNet, it was used the values of $\kappa = 2$, $n = 5$, $\alpha = 10^{-4}$ and $\beta = 0.75$. These values were the best parameters found using the validation set, when training the model. Analysis of Eq.3.2 shows that, given an activation, its neighbors (activations that are in the vicinity) will be inhibited, what means that their values will be decreased by normalization. The authors call this method "brightness normalization" since normalization is applied directly, without using the mean value, as another classical normalization techniques.

3.3.2.4 Overlapping Pooling

Newer convolutional network pooling layers are used such that there is no overlap between pooling kernels. Typically, max-pooling layers are used, which get the highest activation value in a kernel-size fetch region. However, in the AlexNet network, an overlapping pooling scheme with a kernel size of 3 and a stride of 2 is used; the overlapping size is set to the unit value.

3.3.2.5 Architecture

Fig.11 shows the 8-layer AlexNet network; 5 convolutional and 3 completely connected. This network was created to solve a classification problem with 1,000 classes. Thus, the network output is formed by applying the Softmax function to 1,000 different classes.

From Fig.11 it is possible to verify the division applied to the network due to limitations in the hardware, as already mentioned before. In this split scheme the second, fourth and fifth convolutional layer kernels are connected only to the feature maps of the previous layers that are in the same GPU. This can be seen by checking the mentioned layers and the dashed lines crossing the parallel networks in the tables 1, 2, 3 and 4. The kernels of the third convolutional layer are connected to all feature maps of the second layer, just as neurons of the fully connected layers are attached to all neurons of the previous layers.

The Local Response Normalization (LRN) scheme is applied only to the first and second convolutional layer outputs. Pooling is applied overlapping and max-pooling (kernel-pooling) is used on the standard response outputs of the first and second tiers, and a fifth con-

Convolutional layers	conv1	conv2	conv3	conv4	conv5
kernel	11x11	5x5	3x3 3x3	3x3	3x3
padding	0	2	1 1	1	1
stride	4	1	1 1	1	1
channels	3	48	128 128	192	192
filters	48	128	192 192	192	128
neuron	-	ReLU	ReLU	ReLU	-

Table 1 – The attributes of the convolutional layers of the AlexNet model.

LNR	conv1	conv2	conv3	conv4	conv5
channels	48	128	-	-	-
size	5	5	-	-	-

Table 2 – The attributes of the LRN layers of the AlexNet model.

Max Overlap Pooling	conv1	conv2	conv3	conv4	conv5
kernel	3x3	3x3	-	-	3x3
stride	2	2	-	-	2
channels	48	128	-	-	128
neuron	ReLU	-	-	-	ReLU

Table 3 – The attributes of the pooling layers of the AlexNet model.

Fully-Connected layers	fc1	fc2	fc3
outputs	2048 2048	2048 2048	1000
neuron	ReLU	ReLU	Softmax
dropout	0.50	0.50	-

Table 4 – The attributes of the fully-connected layers of the AlexNet model.

volutional tier is also applied. The ReLU activation function is applied to the output of each convolutional layer and completely connected.

3.3.2.6 Decreasing Overfitting

In order to reduce the possibility of overfitting some measures were taken, the first one was the creation of data from the training images (Data Augmentation). Two types of transformations were made in the input images; translation and reflection. Since the original images are 256x256 pixels in size, 224x224 pixels random cropping has been extracted.

In the test stage, the input image is translated by shifting the crop to the four corners of the image plus the central region; Then the image is reflected horizontally and the same translation and crop extraction operation is performed. This operation generates ten patches, for each one the prediction is generated and, finally, the predictions for the ten patches of the original image are averaged.

The second form of data generation was the use of Principal Components Analysis (PCA) to change the intensity values of the RGB image. This simulates lighting changes in a scene with a given object.

Another way to combat overfitting is to apply dropout at the training stage. The AlexNet network applies dropout into the first and second convolutional layers with a 50% probability. Without the use of this method there is the incidence of overfitting reported by the authors in (KRIZHEVSKY; SUTSKEVER; HINTON, 2012).

3.3.3 GoogleNet

The work developed in (SERRE et al., 2007) used Gabor filters of different sizes to manipulate multiple scales. In the development of the GoogleNet network a similar approach was used: filters of different sizes were used. Besides, the use of internal networks, as presented in (LIN; CHEN; YAN, 2013), was taken as inspiration for the creation of GoogleNet. In that work a network named Network In Network (NIN) is presented in which subnets - a convolutional layer connected to a Multilayer Perceptron (MLP) - are used to compose the network layers, the architecture used in (LIN; CHEN; YAN, 2013) is presented. in Fig.12.

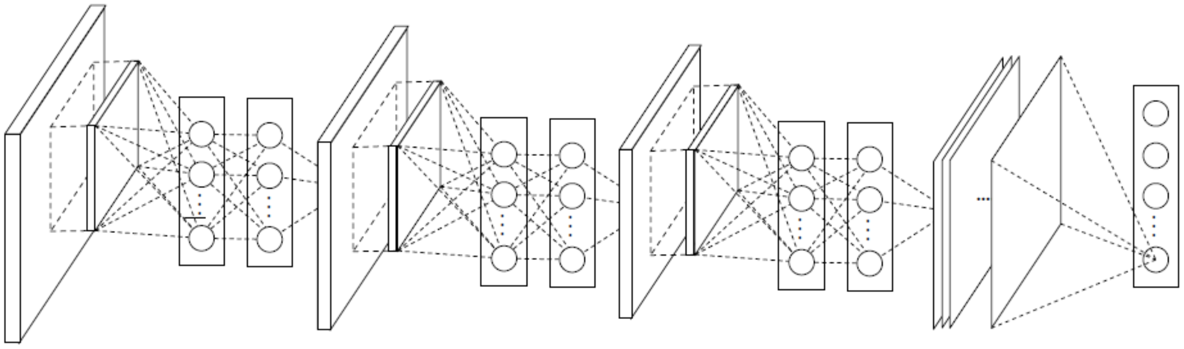


Figure 12 – NIN network architecture.

Source: Image taken from (LIN; CHEN; YAN, 2013)

Another inspiration from the NIN network was the use of convolutional layers with additional 1x1 filter dimensions. These additional layers serve two purposes in the GoogleNet network architecture: firstly to perform the dimensionality reduction, explained below, and second to increase the depth and width of the network, factors that tend to contribute to the generalizability of the network, as discussed in (ZEILER; FERGUS, 2014) and (SZEGEDY et al., 2015).

As discussed in (ZEILER; FERGUS, 2014) and (SZEGEDY et al., 2015), increasing the depth of the net typically results in increase of accuracy. However, the increase in the dimensions of a deep net brings consequences such as the tendency to overfitting, more apparent in cases where the complexity of the net is too much in relation to the size and variability of the training database. In such cases, it is possible to increase the database, a very expensive task that repeatedly requires expert analysis. Another problem with increasing the size of a network is the increased computational load: a deeper network with more training parameters usually requires more memory and processing resources for training. One solution addressed to solve such problems is the insertion of sparsity into the network. However, computation of sparse models is currently not efficient, the problem gets even bigger as current hardware and libraries are designed to perform well on dense components such as dense matrix multiplication.

One way to use sparsity was shown in (LECUN et al., 1998) with presentation of the Lenet-5 network. In this network, it was used a sparse connection table grouping neurons in order to break the symmetry of the network and improve learning. However, newer architectures have begun to utilize dense, uniformly structured connections, made possible by the use of parallel computing and increased training parameters set. The big challenge facing GoogleNet architecture was the creation of a sparse architecture model that makes use of the existing enhanced computing infrastructure for dense computing.

3.3.3.1 Inception Architecture

The fundamental idea behind the creation of the Inception architecture was achieved by considering how an optimal local sparse structure of a typical convolutional network can be approximated and covered by dense components (SZEGEDY et al., 2015).

To achieve translation invariance, in the construction of GoogleNet, convolutional filters were used. The construction addressed by Arora et al. (ARORA et al., 2014) suggests an optimal network construction performed layer by layer, using clustering of the most correlated neural units. These groupings form the units of the next layer and are connected to the previous layer. This process repeats until the end of the network. In the Inception model, it was assumed that each unit, generated as described, corresponds to a region of the input image and these regions are grouped into filter banks. In layers closer to the image (first layers), coverage would be concentrated in more local regions, finer details of the image. Thus, several clusters would be located in a single region, or in a few regions, and as suggested in (LIN; CHEN; YAN, 2013), these thinner regions could be covered by convolutions with 1x1 dimensions. However, other larger and more spread out regions could be covered by wider convolutions, so larger convolutional filters are used for such regions. For the Inception network 1x1, 3x3 and 5x5 filter dimensions were used, the convolutional filter outputs are concatenated, generating a single convolutional filter output. In addition to convolutional filters, a pooling block is used parallel to convolutional filters, due to high acceptance and success when using this type of block. The result can be seen in Fig.13.

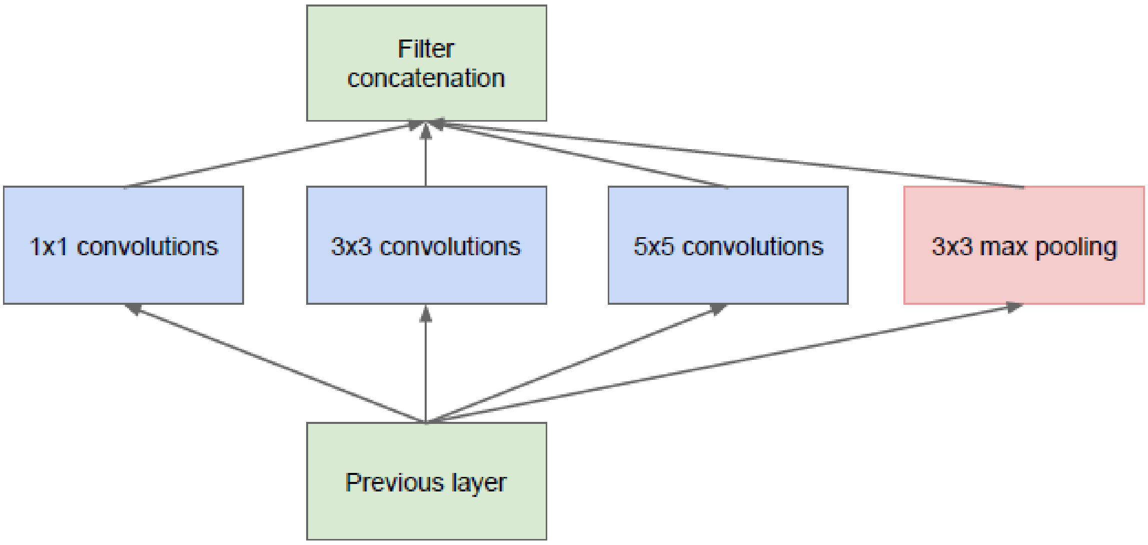


Figure 13 – Inception module.

Source: Image taken from (SZEGEDY et al., 2015)

From the figure, it can be seen the input from a previous layer; in this architecture it is expected that the previous layer follows the same construction model presented in (SZEGEDY et al., 2015) and modeled with convolutional networks. In the middle, there is a layer with the convolutional blocks of variable sizes and a pooling block; this layer represents the grouping of correlated regions, to capture groupings in more or less scattered regions of the image, convolutional filters of different sizes are used. At the top, there is a layer that concatenates the result from the middle blocks; this layer addresses the generation of the next layer elements in the Arora model.

The presented Inception module is said to be naive as it performs direct convolutions on the input layer. Thus, many parameters are calculated and a massive number of operations are performed, causing bottlenecks in the network. To mitigate this problem without decreasing network performance, convolutional blocks with 1x1 filters were inserted into the inception block as shown in Fig.14.

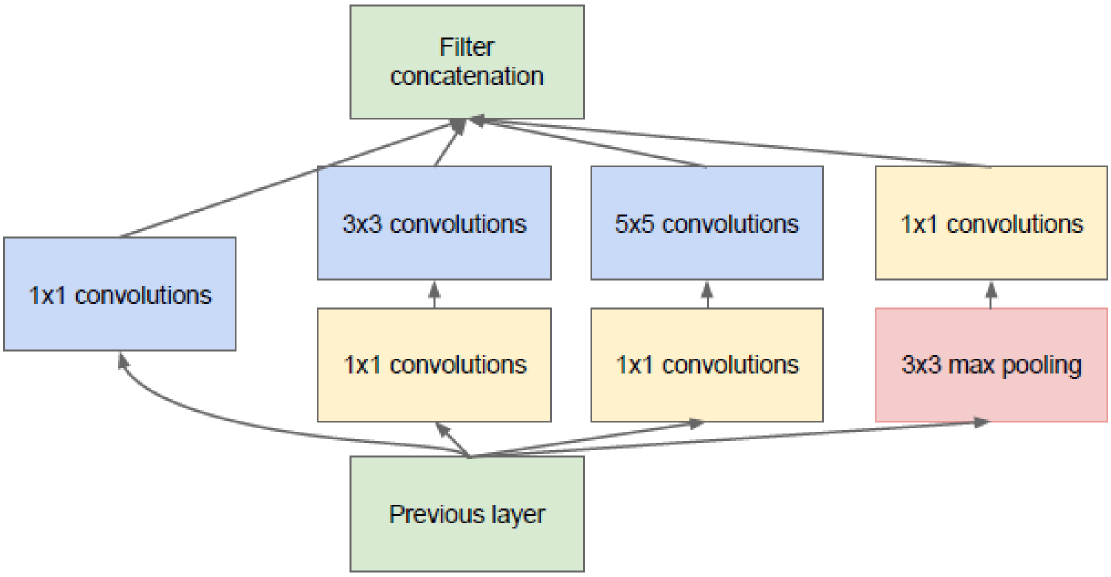


Figure 14 – Extended Inception module.

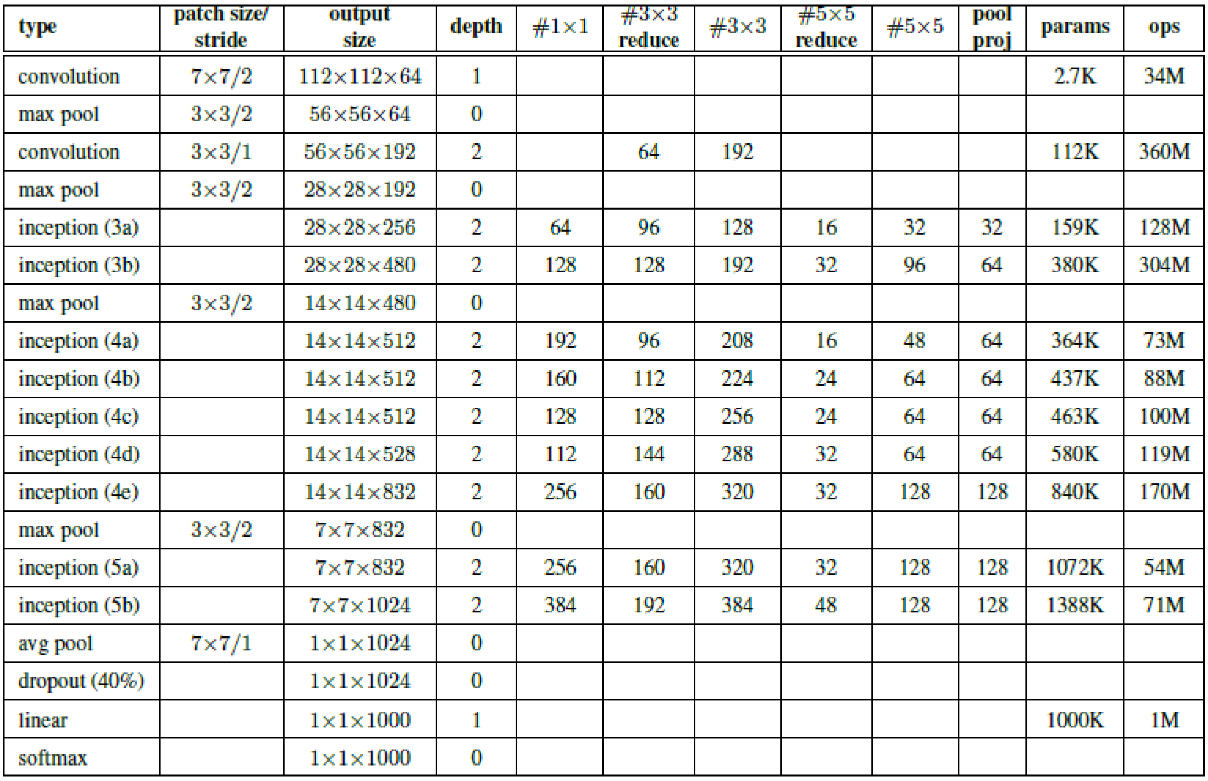
Source: Image taken from (SZEGEDY et al., 2015)

The GoogleNet model can be generically presented as a composition of several connected inception blocks creating a long convolutional network. In the final architecture, traditional convolutional blocks are used in the first layers, this approach yielded the best results. Fig.15 shows the parameters of the ILSVRC 2014 winning GoogleNet network.

3.3.4 ResNet

The use of deep convolutional networks has been recurrent in the ImageNet challenge over the years. Exploitation of this type of architecture and the increase in the number of layers caused the test error to reach almost the average user value (5%). Increasing the number of layers has proved crucial to increase network accuracy.

As the number of layers increased, some problems arose as disappearance or saturation of the gradient value. To avoid such problems, normalization techniques were created: normalized initialization and batch normalization. Even with such solutions, adding too many layers



type	patch size/ stride	output size	depth	#1×1	#3×3 reduce	#3×3	#5×5 reduce	#5×5	pool proj	params	ops
convolution	7×7/2	112×112×64	1							2.7K	34M
max pool	3×3/2	56×56×64	0								
convolution	3×3/1	56×56×192	2		64	192				112K	360M
max pool	3×3/2	28×28×192	0								
inception (3a)		28×28×256	2	64	96	128	16	32	32	159K	128M
inception (3b)		28×28×480	2	128	128	192	32	96	64	380K	304M
max pool	3×3/2	14×14×480	0								
inception (4a)		14×14×512	2	192	96	208	16	48	64	364K	73M
inception (4b)		14×14×512	2	160	112	224	24	64	64	437K	88M
inception (4c)		14×14×512	2	128	128	256	24	64	64	463K	100M
inception (4d)		14×14×528	2	112	144	288	32	64	64	580K	119M
inception (4e)		14×14×832	2	256	160	320	32	128	128	840K	170M
max pool	3×3/2	7×7×832	0								
inception (5a)		7×7×832	2	256	160	320	32	128	128	1072K	54M
inception (5b)		7×7×1024	2	384	192	384	48	128	128	1388K	71M
avg pool	7×7/1	1×1×1024	0								
dropout (40%)		1×1×1024	0								
linear		1×1×1000	1							1000K	1M
softmax		1×1×1000	0								

Figure 15 – Google net parameters.

Source: Image taken from (SZEGEDY et al., 2015)

to a network causes a saturation problem. As discussed in (HE; SUN, 2015), which analyzes the performance impact of a network by varying its parameters, the excessive increase in the number of layers causes the network error to increase, for each new layer added the error value is increased. Although it may seem, this type of behavior is not associated with the phenomenon of overfitting, in which, with the network increase, would be expected the error to stop growing in some moment.

Since there is a problem with the way multi-layered networks learn, a learning method called Residual Learning was created. In this method, the mapping function is replaced by a residual version, in which a term called residue is added to the function (HE; ZHANG, et al., 2016). $F(x)$ is the objective function to be approximated:

$$F(x) = H(x) - x \tag{3.3}$$

where x is the input of the function, and $H(x)$ represents the network mapping (adjustment of the weights in the intermediate layers).

The most straightforward example of the usefulness of this learning method is when one wants to approximate the identity function. This function only maps the input to the output, so the function $F(x)$ would be equal to x. Therefore, it would only be necessary to turn to zero the value of $H(x)$ in equation Eq.3.3. However, in a conventional learning situation, for a network to learn the identity function, it is necessary the weights of their layers to represent the mapping function being adjusted in training. In deep networks, a residual learning block is implemented following the model of Fig.16.

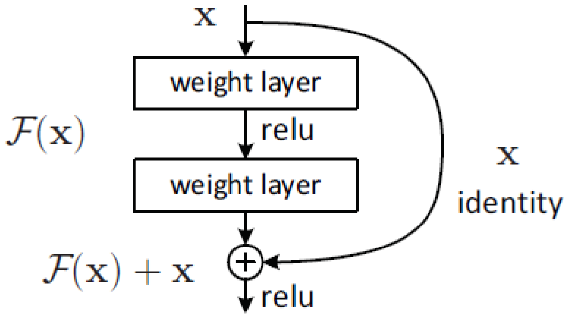


Figure 16 – Residual learning block.

Source: Image taken form (HE; ZHANG, et al., 2016)

In this block, one or more layers are placed between the input and the residual function. The objective function $F(x)$ and the input of the previous block, the x in the 16, are summed before function activation. One limitation is related to input and function sizes: both must be the same size to maintain the consistency of the sum operation. For this, a linear projection can be performed.

In Fig.17, two neural network architectures are presented: a flat architecture and a similar architecture with shortcut connections. The first is used as a comparative model for creating the ResNet architecture. The two architectures have similar features: neighboring convolutional layers are used such that, for feature maps of the same size, the layers have the same number of filters, and, if the size of the feature map is cut in a half, the number of filters is folded. Thus, the height and width of feature maps are decreased and their depth is doubled with each new set of convolutions. This approach is performed in order to maintain the original volume and hence the time complexity in each layer. Downsampling uses a stride of 2 for convolutions, reducing the width and height by half when using 3x3 dimension filters. At the end of the network, an average pooling block and a completely connected layer with 1,000 units and softmax are used. Shortcut connections are inserted in the residual architecture. The solid lines in Fig.17 represent layers with equal feature map sizes and those with dashed lines represent layers with different feature map sizes. The increase in dimensions between feature maps of neighboring layers makes it necessary to perform some adaptation to shortcut connections so that the dimensions at the sum are equal. Thus, as mentioned earlier, one option is to perform a linear projection, an approach that requires additional parameters in the network. Another possibility introduced is to perform padding to increase the dimension. Either approach is still required to downsample using a 1x1 convolution with a stride of 2.

Training parameters and network normalization methods can be found in (SZEGEDY et al., 2015). In the same work, several residual network configurations are evaluated and, for comparison, the flat architecture is used. Among the experiments performed it was noted that:

- The increase in the depth of the flat net caused the error rate to increase in the validation training set, being affected by the degradation problem.
- The increased depth of residual networks caused the error rate to decrease in the validation training set, favored by the greater complexity of the network and overcoming the degradation problem.
- The approaches of using padding, linear projection only in layers with different characteristic maps or linear projection in all layers to solve the problem in the shortcut connections step presented similar results, and for the first solution no parameters are inserted in the network in relation to other alternatives.

To make even deeper networks, while trying to maintain time limits on the number of operations, a methodology of feature map complexity reduction, similar to that used in the GoogleNet network, was tested: 1x1 filters were inserted before and after the 3x3 filters in network layers, Fig.18.

With this model, residual networks with 50, 101 and 152 layers were evaluated. For the summation between neighboring feature maps step, linear projection is used only between layers of different sizes. Each convolutional two-layer block in Fig.17 is replaced by a three-layer block with complexity reduction (Fig.18 on the right). The increase in the number of layers presented results similar to those already observed for residual networks: the reduction in the error rate, reaching the top-5 validation error value of 4.49% with the 152 layer architecture. Finally, 6 models of different depths were combined, culminating in an error rate of 3.57% and winning the 2015 ILSVRC.

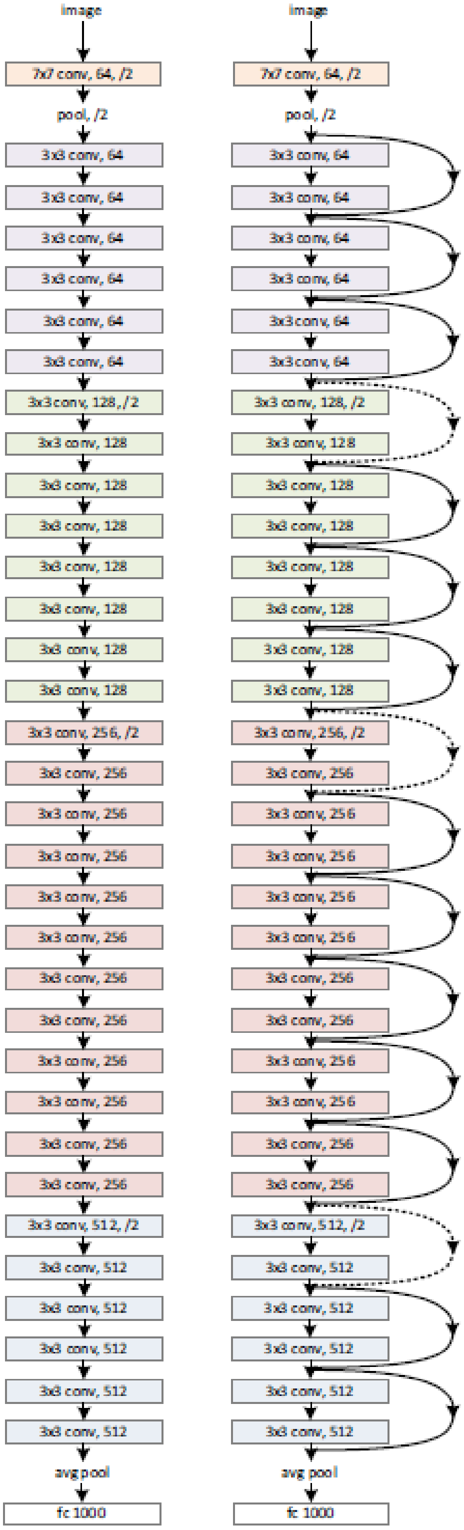


Figure 17 – Flat architecture (left) and ResNet architecture (right).

Source: Image edited form (HE; ZHANG, et al., 2016)

3.4 SPEECH RELATED MODELS

In the last few years, CNN based approaches brought new solutions to problems related to speech. In some cases the appropriate use of the convolutional approaches overpass the

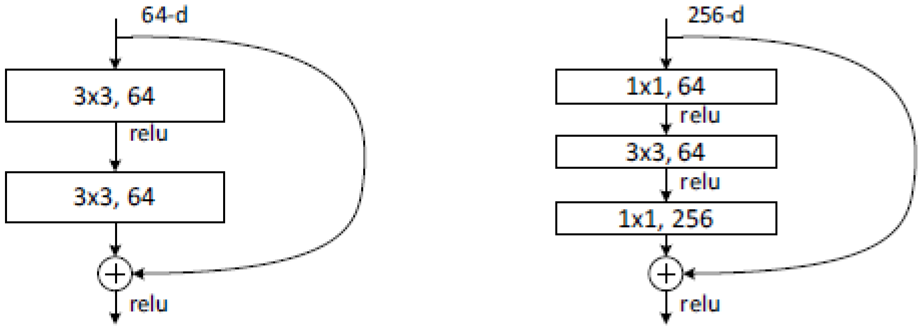


Figure 18 – Convolutional blocks with shortcut connections of similar complexity due to the use of the right bottleneck model.

Source: Image edited form (HE; ZHANG, et al., 2016)

human accuracy in some tasks, in another cases the solutions bring new ways to solve old problems.

In this section are briefly discussed important works whose presented solutions to solve speech related problems using CNN. The discussion is focused on the main aspects which contributed to the performance of the models. Some of the models presented in this section were used as the inspirations to the development of the proposed model presented in the next chapter.

3.4.1 Speech Recognition

The Automatic Speech Recognition (ASR) term refers to a classical recognition area in which the transcription of speech into spoken words is the main object of study. Before the appearing of CNNs, the ASR models was typically using an architecture based on the Hidden Markov Models (HMM). This type of model is still largely used because it is useful to model the changes between phones of a language. Besides the use of HMM, a classical architecture used in the ASR task involves the use of Gaussian Mixture Models (GMM). This algorithm was useful to model the probability of a feature vector, representing a speech frame, to be the equivalent form of a phone. In conjunction, the GMM-HMM model was topically presented as the state of the art when treating ASR tasks.

With the introduction of deep learning models new ASR approaches arise, many of them still employing the HMM model. As presented in (ABDEL-HAMID et al., 2014) it is used an approach involving the use of a CNN instead of the GMM, the so called CNN-HMM model. In that work is highlighted that the use of CNN layers was benefical in a sort of manners.

Firstly, the CNN locality property gives a better treatment for features affected by noise. As the receptive fields have limited dimensions, they cover a limited area of the input feature map, their weights will learning local features that can be combined in higher convolution layers. This represent an improvement in relation the classical DNNs where all cells of a feature map is connected to an unit to the DNN layer. The weight sharing property is too appointed as benefical to the modelling of speech in ASR tasks, that property reduce the overfitting by applying an analysis by the same weights for multiple frequency band, instead of only one predefined region as occurs in the DNNs models. Besides, the use of pooling in CNN models is useful to model small frequency shifts that occurs in speech. In resume, the use of CNNs is a benefical approach because these models are better to model speech variability than the standard DNNs. In the work presented in (ABDEL-HAMID et al., 2014) was showed a performance improvement when using CNN instead of a classical DNN approach, while maintaining a similar number of parameters to train the model.

3.4.2 Speech Enhancement

For speech enhancement, in (PARK; LEE, 2017), a fully convolutional model was used to remove babble noise. In that work, a generative approach with convolutional layers was used to obtain the enhanced speech version. The CNN structure used was virtually separated in two parts: an encoder and a decoder. At the encoder, the feature extraction processing was addressed by the use of layers with a convolutional layer, batch normalization and the ReLU activation. The number of filters at the encoder was gradually increased. At the decoder, the same layers are used, although the number of filters are gradually decreased. The encoder-decoder approach was used as a resource for firstly encoder the features into a higher dimension space and later applying a compression along the decoder. The mentioned model achieved similar or higher accuracy with less parameters than the state of the art recurrent network model at the epoch.

The results achieved by the fully-connected approach were associated, by the authors, to the increasing dimension of the feature space by the encoder and later decreasing by the decoder. Two details are important in the success of (PARK; LEE, 2017): firstly, pooling layers were not used, thus no data was lost and all information was preserved at the encoder. Secondly, the increase of feature space by increasing convolutional layers, at the encoder, conducted the generated features to a higher dimension representation.

3.4.3 Speech Intelligibility

As aforementioned, intelligibility and quality are the two mainly aspects which characterize speech. The intelligibility aspect is more concerned to the quantity of samples correctly understood. It can be defined as the average of words that listeners can understand in a given listening condition. As in the case of speech assessment algorithms, the SIP models can be divided into reference and non-reference models. Similar to the speech quality assessment area, there is the necessity of the development of non-reference models.

The use of algorithms to address Speech Intelligibility Prediction (SIP) is extensive, and in the literature methodologies arise aiming to predict it by using the intrusive and non-intrusive approaches. Some models are widely known, such as the Speech to Reverberation Modulation energy Ratio (SRMR) (FALK; ZHENG; CHAN, 2010) and the Short-Time Objective Intelligibility (STOI) (TAAL et al., 2011).

More recently, with the popularization of deep learning, new SIP methodologies emerged, some of them adopting data-drive approaches. In (ANDERSEN et al., 2018) a non-intrusive SIP methodology is proposed based on the Convolutional Neural Network architecture. The CNN structure was chosen because its handle well inputs of varying size dimensions and because its convolutional kernels can be visually inspected. In the referred study the evaluation is accomplished for full signal samples consisting of multiple sentences. This approach was a experimental design choice since many of the systems processing speech works on the frames of a signal.

Many of the SIP algorithms assume that contributions to intelligibility are supplied from separated channels (different frequency bands), and that these contributions are combined by a linear weighted function. The work presented in (ANDERSEN et al., 2018) mimics this behavior by combining the outputs from the fully-connected layers to form the prediction of intelligibility.

A series of experiments were accomplished by datasets combined from different sources. Another experiments were taken from additional datasets which were not used for training. They compare the CNN model with non-intrusive and intrusive SIP algorithms, some of them from the STOI family. The main results appointed that the accurate predictions were taken for unseen conditions. Besides, the model respond well for clean speech or noise speech only datasets.

3.4.4 Speech Assessment

It must be highlighted that, during the final development phase of this dissertation, it was found that some researches explored the capacities of convolutional layers to evaluate speech quality in an automatic manner (LO et al., 2019), (AVILA et al., 2019). These works could not be tested due to the lack of time.

4 PROPOSED MODEL FOR SPEECH QUALITY ASSESSMENT BASED ON CNN

Based on the CNN models presented in the last chapter, it becomes evident that convolutional layers achieve high performance at the speech and noisy characterization. Then, it becomes natural their use in areas involving speech quality demands. Therefore, the focus of the present work is directed to the evaluation of a new model addressing speech quality assessment by using convolutional layers. The next section shows the aspects of the model developed.

As one of the most important contributions of the present work, the details of the new methodology addressing automatic speech quality assessment is explained in this chapter. The proposed model was developed based on the findings presented in the previous section, and by the incremental improvements when evaluating the model on available speech quality databases.

4.1 OVERVIEW

To analyse the performance of CNNs in the task of speech quality assessment, it was created a new model, referenced as Convolutional Neural Network approach for Speech Quality Assessment (CNN-SQA). The model can be divided into three main steps: input transformations which convert the input speech signal into a visual representation, the feature extraction in which are generated features representing the speech characteristics, and finally a quality estimation step that generates the quality judgment from the extracted features. The model steps are shown in the diagram of Fig.19. Each layer is indicated by a label, the topmost labels are used to group related layers.

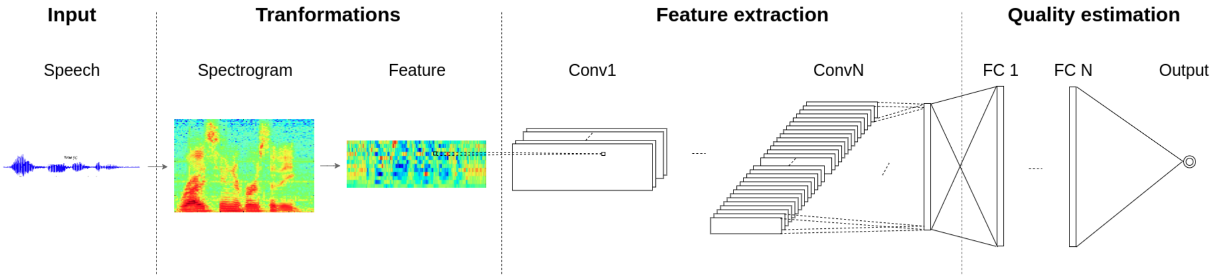


Figure 19 – CNN-SQA model steps.

Source: Generated by the author.

The model receives the input speech samples as illustrated in blue waveform at the left side of Fig.19. The speech samples are uncompressed digital records in raw format, in which it is expected to have parts of speech activity, with continuous utterances in a human language, and silence (typically with some noise). At the picture, it is seen only one speech sample, but in the practical model realization the application receives mini-batches, random groups of samples used to train models.

In the first step of the pipeline, transformations are applied to the input samples, resulting in visual speech features, used to train the convolutional layers. The first transformation is the spectral generation in which speech samples are converted into spectrograms. In the spectral representation, the underlying frequencies are better presented just as the time variation is maintained at the appropriate axis. Other transformations can be applied with the aim to generate the final features; Fig.19 shows only one, for simplicity. A possible second transformation is the generation of new features from spectrogram; this step is typically applied in order to compress the features dimension since, in the spectrogram, the visual representation is sparse and it maintains redundant information.

The second phase of pipeline shown in Fig.19 is the feature extraction step. From a visual representation, generated in the transformation step, are extracted the main features representing the speech and distortion characteristics. To do such job is crucial the use of good feature extractors, which can represent the more important characteristics related to speech quality perception. As stated in literature and presented in the previous chapter, convolutional layers were successfully employed in speech and noisy related tasks, as in (PARK; LEE, 2017), (ABDEL-HAMID et al., 2014), (ANDERSEN et al., 2018). These works show that convolutional layers are able to produce feature extractors with higher generalization performance, often outperforming the state of the art methodologies. The feature extractors are generated by training the convolutional layers; it is expected they are able to learn the best features representing the speech signal.

The final step is the quality estimation; it simulates the human quality assessment processing. As it has been described in literature (ABIODUN et al., 2018), (JELASSI et al., 2012), fully connected layers were applied to speech problems, presenting good results when comparing with handcrafted algorithms. Beyond traditional speech applications, parametric approaches had been employed neural networks architectures to map speech related network parameters (delay, jitter, packet-loss, etc.) for quality estimation (ROZHON et al., 2016), (YANG et al., 2016). So, based on the performance of neural networks on speech perception tasks, the final step of the proposed model is accomplished by the use of fully connected layers (dense layers), which aim to map the high level features to a quality score estimation. The output is generated by the use of a regressor, an output layer that aggregates the trained weights into a value, representing a quality score.

4.2 CNN-SQA

The proposed model described in this section is based on the general steps shown in Fig.19. It is presented in a block diagram format, with architectural information in each block; the layers are depicted in Fig.20.

4.2.0.1 Parameter selection

To define the parameter set for the proposed model it was performed a parameter selection step with a grid search approach. The ITU-T Supplement 23 database was used to perform the parameter selection, 60% of data was used for training and 20% for parameter validation, while the remaining 20% was used in the test phase, explained in the experiments chapter. In the first step was choose an initial configuration (dummy) for the model parameters, this configuration can be seen in Tables 5, 6 and 7. Then successive parameter selection steps were performed, so that each model step would be trained with the selected parameters, varying, within predefined values (grid search), while the remaining parameters were kept the same. At the end of each selection step the best value for the evaluated parameter is maintained. Thus, it becomes possible to verify the improvement of the model according to the parameters change, in addition to making a combination of all possible model parameters would be impractical which requires the adoption of methods such as the one shown.

Feature	Coefficients
MFCC	40

Table 5 – Initial feature parameters of the proposed model used for parameter selection step.

Layers	Filters	Filter dimensions	Activation
1	50	5x5	ReLU

Table 6 – Initial convolutional layer parameters of the proposed model used for parameter selection step.

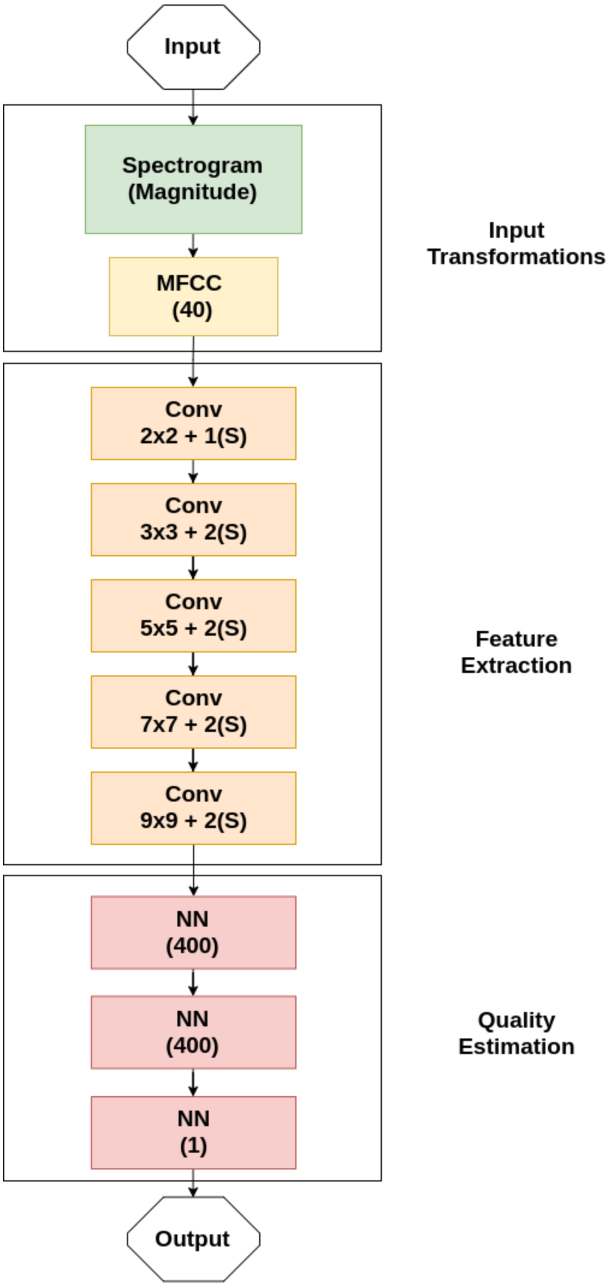


Figure 20 – Block diagram of the proposed model.

Source: Generated by the author.

Hidden layers	Units per layer	Activation
1	400	ReLU

Table 7 – Initial fully-connected layer parameters of the proposed model used for parameter selection step.

4.2.1 Input Transformations

The input transformations, used in the proposed model, are shown in the first stage of Fig.20. In this work, two transformation steps are used: a spectrogram conversion and generation of the corresponding Mel Frequency Cepstral Coefficients (MFCC).

4.2.1.1 Spectrogram

The input speech signal is transformed into a spectral-temporal representation by using the Short Time Fourier Transform (STFT). To do so, it is usual to define a proper analysis window length. As explained in (MCLOUGHLIN, 2009), the speech presents slowly changes in its spectral characteristics, being able to assume it is pseudo-stationary state over the period of about 20-30 ms. In this view, it is used a frequency analysis window with duration of 32.0 ms, the first multiple of 8, greater than pseudo-stationary range (as the typically sampling frequency used on speech is multiple of 8 KHz). The overlap used is 75% over the frame length, and the Hanning window is applied. To avoid manipulation of complex numbers, their absolute values are taken.

4.2.1.2 MFCC

As for the selection of other model parameters, the selection of the model input feature was performed by evaluating the visual features: spectrogram absolute values, mel spectrogram, MFCC and PNCC.

The mel scale is a nonlinear psychoacoustic scale (STEVENS; VOLKMANN; NEWMAN, 1937). The distances between their units (mels) are judged by listeners to be equivalent in the perceived sound distance. A classical equation to convert Hz to mels is presented in Eq.4.1.

$$m = 2595 \log_{10}\left(1 + \frac{f}{700}\right) \tag{4.1}$$

The mel spectrogram is accomplished by the use of half-overlapped triangular windows (band pass filters) equally spaced on the mel scale. Firstly is applied a windowing to get the frames of spectrogram. Each magnitude spectrogram frame is multiplied by band pass filters and summed, resulting one bin per filter. The whole process, applied over one frame, is resumed in the diagram of Fig.21.

As stated in (DAVIS; MERMELSTEIN, 1980) the MFCC generation is accomplished by using a similar pipeline of Fig.21, with additional processing steps. A logarithmic non

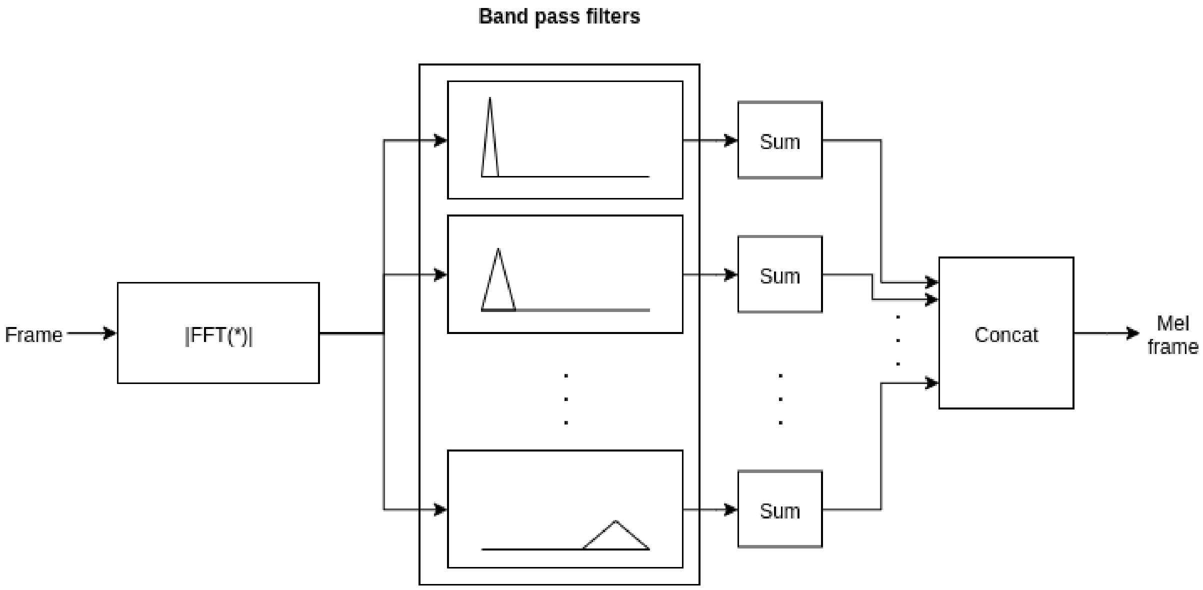


Figure 21 – Diagram of the mel spectrogram generation.

Source: Generated by the author.

linearity is applied on the result of the triangular frequency integration. Lastly, to generate the cepstral coefficients, it is used a Discrete Cosine Transform (DCT), Eq.4.2.

$$X_k = \sum_{n=1}^{N} x_n \cos\left[\frac{\pi}{N}\left(n+\frac{1}{2}\right)k\right] \qquad k = 1,\ldots,N. \tag{4.2}$$

where N is the number of cepstrum coefficients, and x_n is the log-energy output of the k-th filter. The general steps can be seen in the diagram of the Fig.22.

The last feature evaluated is the PNCC, presented in (KIM; STERN, 2016). The PNCC generation has similarities with the generation steps of MFCC; the differences can be seen in the colored blocks of Fig.23. In the figure, it can be seen that the blocks of STFT, magnitude squared and DCT are the same as used in the MFCC generation. The Pre-Emphasis and Mean Normalization blocks are steps that can be applied in MFCC, they act as compensation filters used before and later of coefficients computation. Equivalently to the use of triangular integration in MFCC, in the PNCC generation, it is used a gammatone-shaped filter bank, and it is used a different linear spacing approach. It was verified, by the authors, that the use of gammatone frequency weighting provides a better ASR accuracy in white noise. In the Time Processing block, Fig.23, are performed nonlinear time-varying operations to ease the environmental degradation by use of a noise subtraction approach. The step of Mean Power Normalization is used to minimize the impact of amplitude scaling, that can be affected by the use of the power-

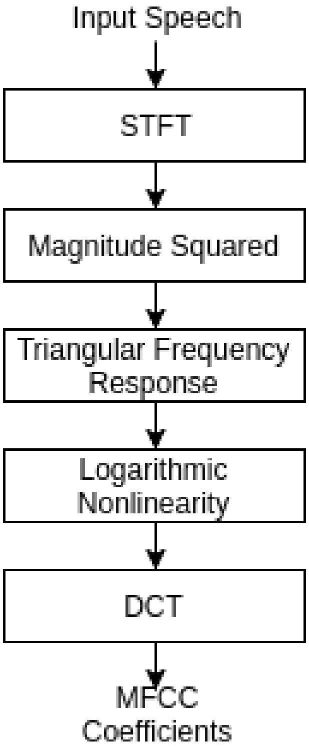


Figure 22 – Diagram of the MFCC generation.

Source: Generated by the author.

law nonlinearity. In the PNCC, after frequency integration, instead of apply a Logarithmic nonlinearity, as in MFCC, it is applied a Power-law nonlinearity; this choice was done based on the believe that this nonlinearity provides a superior robustness when suppressing small signals.

The feature which resulted in the best trade-off between results and space complexity for our model was MFCC; so it is used as the main visual representation of the speech signal. To define the quantity of MFCC coefficients as input for the CNN layers were evaluated the use of 13, 20, 40 and 60 coefficients, in the parameter selection phase. The use of 13 coefficients is well reported in literature, the other values were used to verify how the proposed model changes with the use of more or less coefficients. The result that presented the best trade-off between model accuracy and memory usage was obtained by applying 40 MFCC coefficients. The coefficients are generated using a mel scale spaced from 20Hz to 4kHz. The results found for MFCC can be associated to the fact that this type of visual feature is very sensitive to noise interference, providing a greater correlation with quality perception aspects.

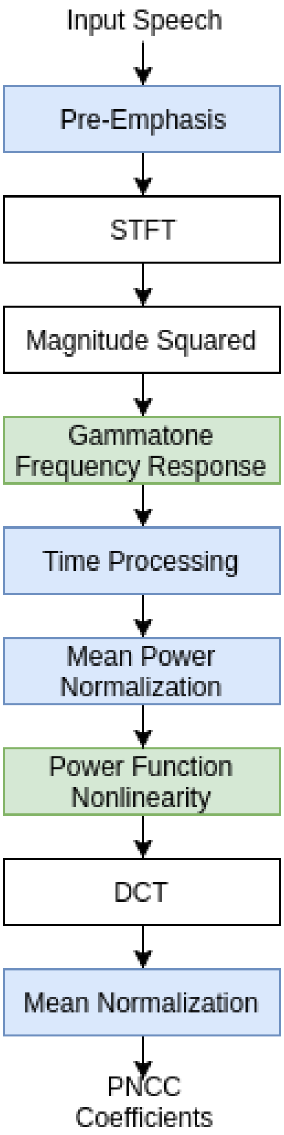


Figure 23 – Diagram of the PNCC generation. The blue boxes are the ones which does not exist in the MFCC generation diagram. The Green boxes represents the equivalent operations in the MFCC generation diagram.

Source: Generated by the author.

4.2.2 Feature Extraction

As presented previously, the use of convolutional layers as feature extractor is inspired on the works that successfully employed CNNs to solve problems on speech enhancement and speech assessment. The use of such layers can be seen in the middle block of Fig.20.

4.2.2.1 Convolutional Layers

To determine the number of convolutional layers networks with 1 to 7 layers were evaluated. In that evaluation was found that after the sixth layer the results did no improve significantly. To define the quantity of convolutional filters the use of 20 to 100 filters was evaluated in increments of 20 per convolutional layer. Regarding the dimensions of the convolutional filters, square filters with dimensions of height x width ranging from 2x2 to 9x9 were evaluated, the evaluation of filters with increasing dimensions per convolutional layer was also carried out ranging from 2x2 to 9x9 in steps of 1 and 2 pixels per dimension. Finally, the use of filter dimensions of 2x2, 3x3, 5x5, 7x7 and 9x9 was selected because presented the best results in validation set. The stride parameter was selected based on the evaluation of the possible values 1, 2 and a combination of that values for the five convolutional layers, finally the use of a stride of 1 for the first convolutional layer and 2 for the remaining layers presented the best results.

Altogether are used five convolutional layers with different kernel sizes. The first layer aims to conduct the cepstral coefficients to the feature map domain, so it is used a square kernel of 2x2 bins, and a stride of one bin. Thus, at the output of the first convolutional layer, feature maps are generated with same dimensions of the MFCC result, but with the extended channels. In the following convolutional layers, the kernels dimensions are increased (3x3, 5x5, 7x7 and 9x9), at the same time as the stride is kept in two bins. The increase in kernels dimensions is necessary to build the hierarchical feature space used by the quality estimation module and it is in conformity with (PARK; LEE, 2017). In all convolutional layers, eighty feature maps are generated, this approach presented the best results on the validation step.

4.2.2.2 ReLU

The use of stable activation functions is crucial to successfully train convolutional layers. The most common problem occasioned by the use of unstable activation functions occurs when algorithms, as back-propagation, progresses down to the lower layers with gradients getting smaller and smaller, culminating on weights unchanged, a problem known as vanishing gradients. In opposite, sometimes the gradients can grow bigger and bigger, culminating in large weights and leading the algorithm to diverge, a problem know as exploding gradients. The aforementioned problems are investigated in more details in (BENGIO; GLOROT, 2010). To avoid the problems with activation functions, in the present work, we evaluated the use of three activation functions well reported in the state of the art. The result was that the Rectified Linear Units (ReLU) outperformed the Softplus and Exponential linear unit (ELU) activations.

The ReLU equation is defined in Eq.4.3.

$$f(x) = max(0, x) \tag{4.3}$$

4.2.2.3 Batch Normalization

Another problem involving convolutional layers and their training is called the Internal Covariate Shift. It is caused by the layer's input/output distribution changes during training. The problem can be compensated by the use of a normalization technique called Batch Normalization (IOFFE; SZEGEDY, 2015). This technique acts before the activation function in each layer, applying a zero-centering and normalizing the inputs, then scaling and shifting the result. The Batch Normalization parameters (scale and mean) are learned during the training, so that the model can learn their best parameters. Besides the advantages of Batch Normalization, its use can be benefit to accelerate the training, getting a faster convergence time. As well as for another model parameters, in the parameter selection phase the use of Batch Normalization was beneficial to reduce the model error rate. So, before the activation function of each layer, it is added a Batch Normalization step in the model of Fig.20 (not shown in this figure).

4.2.2.4 Dropout

As a final step in the training of convolutional layers, the use of regularization techniques is applied to avoid overfitting. The most popular and successful regularization technique is called dropout (SRIVASTAVA et al., 2014). This technique consists on the association of an activation probability to a neuron, meaning that it can be entirely ignored during training. This technique avoids the neurons to co-adapt with their neighbors. In the parameter selection phase was evaluated the model performance with and without Dropout as a final step for the convolutional layers, the results appointed improvements in the error rate when using Dropout. This way it is used Dropout as a final step of each convolutional layer, in which a probability of 50% is associated to each neuron.

4.2.3 Quality Estimation

In the final step typically neural network layers are used to the regression of the speech quality score. The selection of the parameters of the fully-connected layer was carried out by evaluating one to three hidden layers, with the same number of neurons for all hidden layers, from 200 to 800 neurons per layer. But the final configuration, which is used two hidden layers with four hundreds neurons each, shows better results in the validation step. As well as for the convolutional layers, the use of ReLU activations presented better results and are used in the two fully-connected layers. The final layer is a regressor which maps the trained weights to a decimal value, the quality score.

5 EXPERIMENTS AND ANALYSIS

To evaluate the proposed model, an experimental workflow is performed using two databases. One of them being a proposed database developed in this work. To conduct the experiments the new model is trained across the databases; its validation errors are compared with the results of other methodologies available. The results analysis, in resume, suggest a better fit of the proposed model on the databases.

5.1 DATABASES

In this section are presented the databases used in the experiments. The database used in the second experiment is a built dataset, used to evaluate the proposed model in a larger number of samples with background noise.

5.1.1 ITU-T Supplement 23 and TCD-VoIP

The first database is a composition of the classical ITU-T Supplement 23 database, with addition of samples from TCD-VoIP database and the use of a data augmentation technique to expand the number of trainable samples. Next are presented the details about the mentioned database.

5.1.1.1 ITU-T Supplement 23

The largest portion of the first database used in the experiments is composed of samples from ITU-T Recommendation Suppl. 23 (ITU–T, 1998). This source was initially created to evaluate subjective tests on CODECs (coder-decoder) at 8 Kbits/s. In the construction of the source material by ITU-T Group, sentences with 2-3 seconds length were used combined with silence periods; the final speech samples have 8 seconds. All sources have been sampled at 16KHz with 16-bit of precision per sample. Three experiments were created in the process, each addressing a subset of degradation available in the context of speech CODECs evaluation. The configuration of each experiment is presented in Table 8.

Each laboratory used four speakers (two males and two females) with different nationalities to create their sources, totaling seven languages used in the experiments. Adding the samples generated by all laboratories, it was built a dataset with 1,480 speech samples.

Experiment	Description	Labs	# samples	Languages
1	Interworking With Other Wireless and Transmission Standards	3	188	French, Japanese, American English, German, Norwegian
2	Effect of Environmental Noise	3	28	French, Japanese, American English, German
3	Effect of Channel Degradation	4	208	French, Japanese, American English, German, Italian

Table 8 – Resume of ITU-T Rec. Suppl. 23 database.

The use of the ITU-T Recommendation Suppl. 23 database is restricted by the utilization of the experiments 1 and 3. This was motivated by the test method employed in each experiment. In experiments 1 and 3 the overall sample quality is rated using the aforementioned ACR test with the MOS scale. On the another hand in experiment 2 is used the Comparison Category Rating (CCR) procedure, in this test is used a different approach of ACR, to the subjects are presented two samples, a distorted and a reference one, in random order. In CCR tests, to rate samples is used a rate scale, to address the quality of the second compared to the quality of the first in a range from much worse quality to much better quality (ITU–T, 1996a). Besides the different rate scale, the use of reference samples makes it incompatible with the ACR approach.

5.1.1.2 TCD-VoIP

The another portion of the first database is composed of samples from TCD-VoIP (HARTE; GILLEN; HINES, 2015). This is a database of degraded speech for assessing quality in VoIP applications. Five types of degradation are in the TCD-VoiP database: background noise, intelligible competing speakers, echo effects, amplitude clipping, and choppy speech (it simulates the missing samples degradation). These are considered platform independent and are the most common VoIP degradation. The instructions in (ITU–T, 1996a) were used for the database creation, Therefore the samples are compatible with the ITU-T Recommendation Suppl. 23 dataset. In the TDC-VoIP database each experiment applied to ACR test comprised a number of conditions in which an amount of quality degradation was applied. Differently of Recommendation Suppl. 23 dataset, in the TCD-VoIP database there are reference condition samples for what no degradation has been applied. Including all samples the packet has a total of 400 samples, in the Table 9 is exposed the resume for the TCD-VoIP database.

Degradation	Conditions	# samples
Competing Speakers	10	56
Amplitude Clippings	10	56
Background Noises	20	96
Choppy Speechs	20	96
Echo Effectss	20	96

Table 9 – TCD-VoIP database conditions.

The original samples of TCD-VoIP database were sampled at 48kHz, so it was necessary apply a downsample to 16kHz to fit them to the same sample rate of the ITU-T Recommendation Suppl. 23. Because the proposed model was not developed to address effects as echos and missing samples, the samples with these degradation were removed from TCD-VoIP database before the experiments. A total of 192 samples were removed, remaining 208 samples.

5.1.1.3 Data Augmentation

In scenarios where few samples are available for training, data augmentation techniques are typically used to supply new samples. In audio processing context, a few approaches are commonly used: speed change, pitch change and noise insertion (CHAN et al., 2016), (PARK; CHAN, et al., 2019). However, for quality analysis, most of them can not be applied because they can change the original samples affecting speech quality aspects. Therefore, a less popular data augmentation approach was used, the Random Circular Shifting (RCS). The RCS is a data augmentation approach, that shifts the data in an array, keeping all original samples. Firstly, a quantity (q) is randomly chosen to be used as total shifting; later it is used to shift all samples to a direction (left or right).

When using the database resulting from the combination of ITU-T Suppl. 23 and TCD-VoIP databases, during the training phase, it was used an RCS approach for data augmentation; for each training sample, it was generated a new version with shifting between 10% and 90% of the audio sample length. This approach was used to avoid small shifts in which the signal would remain pretty much its original version. In order to characterize the levels which a speech signal can be distorted, six different Signal-to-Noise Ratio (SNR) were used; they are evaluated by getting the active levels of speech and noise. The submitted SNR levels were 5dB, 10dB, 15dB, 20dB, 25dB and 30dB. These conditions are used to simulate a range of possible speech levels in a telephonic call.

5.1.1.4 Consideration

The composed database presented in this section was used as a first experimental resource to evaluate the proposed model. To evaluate the model in a large dataset, with better background noise levels representation, it was built a database presented next.

5.1.2 SNDSuppl23

In the last few years, there has been an increasing number of deep learning solutions motivated by the evolution of neural network techniques and computers processing capacity. However, to train the new deep models it was necessary to increase the number of samples in datasets to a scale of thousands. Even with the data available on the internet, and other sources, some challenges must be transposed by those who try to train deep models.

Some areas are well served with data, such as speech recognition, sometimes motivated by the global competitions, as ImageNet Challenge (DENG et al., 2009), and other events. On the other hand, for some areas, as quality assessment, the creation of large labelled databases is time consuming and costly. Some problems in the development of new databases can be identified in (ITU-T, 2016). To solve such problems some solutions were developed: in some cases the dataset can be crafted directly; besides that, it is possible to use small datasets that can be expanded using techniques like data augmentation (MIKOŁAJCZYK; GROCHOWSKI, 2018). Some of the latest models using deep learning to solve speech problems have adapted existing databases inserting speech noisy samples (FU; TSAO; HWANG, et al., 2018), (FU; TSAO; LU, 2016). Even so, there are still many other problems making it difficult to create new datasets, such as in quality assessment area, that depends on human subjective evaluation. As can be expected, the new deep learning solutions involving speech quality assessment need large labelled datasets.

Motivated by the necessity of large datasets in the context of speech quality assessment, it was built a database called Speech Noise Database with Rec. Suppl23 (SNDSuppl23), labelled by the Perceptual Evaluation of Speech Quality (PESQ) (ITU–T, 2005), described in this documentation.

Following, it is described all the details for the built of the Speech Noise Database with Rec. Suppl23 (SNDSuppl23) dataset, including clean speech signal information, degradation methods, the utilities used to build speech and noisy samples, all the main characteristics of the database and the software used to generate the quality scores.

5.1.2.1 Requirements

To correctly use PESQ as quality assessment tool, it was followed the guidelines from the Application guide for objective quality measurement based on Recommendations P.862, P.862.1 and P.862.2 (ITU–T, 2007). Besides PESQ requirements, in order to choose the speech source, it was stated as necessary a representative variability of the samples and a minimum sample length.

5.1.2.2 Speech source

The speech source must be a set of human voice samples created by recording spoken phrases of native human subjects. To aggregate phonemes variability in the speech corpus, speakers of different nationalities are necessary; besides nationality, age and genre are also important factors of human voice diversity taken into account. Therefore, to built the SNDSuppl23, it was used the speech material from ITU-T Recommendation Suppl. 23.

5.1.2.3 Noise source

The distortions used in the development of the built database are composed by background noise samples. These samples were taken from the Suppl.23 dataset, they were considered important to evaluate CODECs in some experiments (ITU–T, 1998). Information about noise is summarized in Table 10.

Label	Description	Length (seconds)
Babble	Office bable	8
Car	Inside car	8
Hoth	Simulated room noise	12
Music	Violins playing	8
Street	Street noise	120
White	White noise	12

Table 10 – Information of noisy samples of ITU-T Rec. Suppl. 23.

5.1.2.4 Conditions

In order to characterize the levels which a speech signal can be distorted, six different SNR were used; they are evaluated by getting the active levels of speech and noise. The submitted SNR levels were 5dB, 10dB, 15dB, 20dB, 25dB and 30dB. These conditions are used to simulate a range of possible speech levels in a telephonic call.

5.1.2.5 Database generation

To prepare the input signals for a validation stage, four steps are applied as shown in the diagram of Fig. 24 and detailed next.

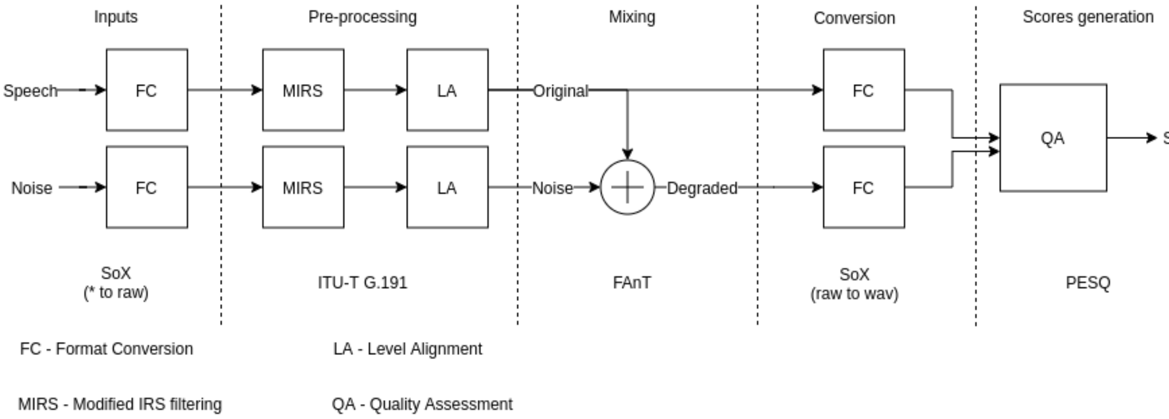


Figure 24 – Database generation steps.

Source: Generated by the author.

Because the pre-processing tool works only on .raw files, it was necessary to convert the .wav input files into a more proper format. To perform the conversion the Sound eXchange (SoX) tool was used; this is a cross-platform command line utility that can convert and apply effects to sound files (NORSKOG, 2014).

The pre-processing steps are used to prepare the input data before applying the mixing operation. Primarily, the signal is filtered by a Modified Intermediate Reference System (MIRS) (ITU–T, 1996b); this is an ITU-T recommendation that simulates the frequencies when transmitting the signals by a network. PESQ assumes that the signals reflects the electro-acoustic characteristics of transmission.

The second pre-processing step is the Level Alignment (LA) used to level the active values in both signal. This step is crucial to conduct the signals to appropriate reference levels before applying mixing. Both signals are level aligned to -30dBov[1].

To pre-processing .raw files, it was used the ITU-T Recommendation G.191 tool to address audio coding standardization (ITU–T, 2019).

The mixing block is responsible to merge the pre-processed signals at a specific SNR level. This is done by applying the Filtering and Noise Adding Tool (FaNT) in the sources signals and the conditions. The FaNT tool (HIRSCH, 2005) is a mixing and filtering signal utility used to generate the Aurora-2 and Aurora-4 databases.

The second conversion step is applied to convert the raw files into an proper wav format used by the quality assessment tool. Here again, SoX tool is used to convert the files.

In the final stage, it is generated the virtual quality scores; it is generated by a full-reference tool simulating the real quality score. The tool used is the PESQ, ITU-T Recommendation to address speech quality assessment in a sort of distortions (ITU–T, 2005). PESQ represents the actual open source state of the art tool in speech quality assessment. As a full-reference tool, PESQ needs the reference signal and its distorted version in order to generate the quality score. Thus, the final step is the PESQ score generation using the reference and distorted speech signals.

5.2 TRAINING METHODOLOGY

In this section, it is presented the details involving model implementation, their parameters, and training approach employed to fit the CNN-SQA model on the databases.

5.2.1 Settings

This section presents the model details and show all parameters used on the model realization. Along of the model validations, each parameter was set to benefit some processing stage, aiming to achieve the best results.

5.2.1.1 Input

The input signal is represented as a waveform, loaded from an audio file. In our case, it is in the PCM 16-bit format. The default sampling frequency used was 16kHz. The parameters are resumed in Table 11.

[1] The dB overload (dBov) is a like decibel measure relative to the overload point of a system.

Extension	Digital format	Quantization	Clip duration	Sampling frequency	Byte order
wav	PCM	16-bit	8000 ms	16kHz	little endian

Table 11 – Parameters related to the input signal.

5.2.1.2 Spectrogram

Generating a spectrogram typically involves to use parameters related to windowing. The window size parameter states the size in samples, being necessary to apply the STFT. Besides, it is necessary to set the overlapping samples which can be equivalently addressed by the window stride parameter; this one represents the samples to jump in each STFT windowing step. To apply windowing, it is also necessary to choose the window format. In this work, it is used the Hanning window as this is a common window for speech applications. The parameters are resumed in Table 12.

Window size	Window stride	Window	Feature
32 ms	8 ms	Hanning	absolute

Table 12 – Parameters related to spectrogram generation.

5.2.1.3 Feature Extraction

The parameters related to the feature extraction process are dependent on the kind of feature used. In this work, some features were evaluated from: spectrogram absolute values, Mel spectrogram, PNCC and MFCC. The feature which resulted in the best results for our model was MFCC; so it was used as the default feature. In a typical MFCC, the main parameter is the number of coefficients generated; in most part of our experiments, 40 coefficients are used. Other parameters are related to the search in the spectral range used to generate coefficients. The parameters are summarized in Table 13. Besides the dimension problem, the use of spectral features to solve problems involving speech is extensive (DUBEY; KUMAR, 2013), (UPADHYAY; KARMAKAR, 2015). Its frequent use is related to its high capacity of speech representation (WU; CAO, 2005).

Type	Coefficients	Lower edge (Hertz)	Upper edge (Hertz)
MFCC	40	20	4000

Table 13 – Parameters related to feature generation.

5.2.1.4 Convolutional Layers

The convolutional layers cover many of the parameters used; their presentation is conducted to the most general to most specific. The first, and one of the most important parameters, is the number of convolutional layers: it is responsible to define how many levels of high level feature extraction are applied on the input visual representation.

The next parameters are related to the layer itself; so each one of them exists for each layer. The first one is the number of convolutional feature maps: each one is generated by applying a convolutional filter fitted in the training model, with more feature maps more characteristics of the preceding layer are captured. Besides the feature maps, the width and the height of each convolutional filter are necessary to set the aspect of visual field in each convolutional layer, this parameter depends on the visual characteristics mapped in the convolutional process. The last parameter involving the convolutional filter is the stride, used in convolution process to define how many feature pixels must be jumped to access the next convolutional mapping.

Other parameters are not involved with each convolutional layer itself but are applied later than each convolutional layer. The activation function is chosen to map the feature map values to activations, typically just one activation function type is shared to all convolutional layers. Other parameters are used to apply regularization steps in order to constraint the model and avoid overfitting. The use of batch normalization steps (IOFFE; SZEGEDY, 2015) is applied later than activation layer and is responsible to normalize the output of the layer, constraining the range values to the next layer. One last parameter can be applied to avoid overfitting too, it does adding an inactivation probability on each layer, this parameter is called dropout (SRIVASTAVA et al., 2014), Table 14 and Table 15 show default values to the presented parameters.

Layer	Feature maps	Filter width	Filter height	Stride
Conv1	80	2	2	1
Conv2	80	3	3	2
Conv3	80	5	5	2
Conv4	80	7	7	2
Conv5	80	9	9	2

Table 14 – Convolutional layer parameters.

Layer	Activation	Apply Batch normalization	Apply dropout
Conv1	ReLU	true	true
Conv2	ReLU	true	true
Conv3	ReLU	true	true
Conv4	ReLU	true	true
Conv5	ReLU	true	true

Table 15 – Additional convolutional layers parameters.

5.2.1.5 Fully connected Layers

As stated before, for convolutional layers, the number of layers is a parameter used to define how many mapping layers must be used. In the context of fully connected networks the more layers added than more complex models are generated.

In a FNN, each layer has the quantity of neurons and the activation function as parameters. The parameters used to set the default fully connected layers are shown in 16.

Layer	Neurons	Activation
FC1	400	ReLU
FC1	400	ReLU

Table 16 – Fully connected layers parameters.

5.2.1.6 Database Partition

To training and validation on the proposed database it was used a stratified partition approach that divides the training and validation sets with similar diversity of samples into noise type and SNR levels categories. The parameters used and the total amount of samples in each set is shown in Table 17.

Set	Parcel	# samples
Training	80%	42624
Validation	20%	10656

Table 17 – Database partitions sets.

5.2.1.7 Training

Other parameters are related to training process itself, they are important to define the time and hardware resources consuming in the training phases; besides, they are crucial to generate good results in generalization of the model. The batch size parameter sets the total of samples applied (in batch) during model training steps; increasing this parameter can accelerate the training but, at the same time, it can be memory consuming and must be handled carefully.

The optimizer sets the algorithm used to train the model, the most used is the Batch Gradient Descent (RUDER, 2016); this algorithm gets the gradient errors to point to the right direction of learning variables changes. The loss is a measure used to define the error during the training process, comparing the predictions with the ground truth values. The default loss used to train the model is the Root Mean Squared Error (RMSE), a typical error measure used when comparing decimal values. In the training process, the batches of samples are used and evaluated by RMSE loss, however during the training process, to evaluate the validation set, it is necessary to apply all validation samples that is much bigger than batch size. Thus to the validation set, it is applied a weighted version of the RMSE measure. The RMSE measure is weighted by batch size on validation step; this approach conducts to a very close RMSE value over validation set.

To avoid overfitting and to add variability on databases, a schema where samples are artificially generated can be used, a process called data augmentation. The process of new samples generation includes the use of the original database samples applying data transformations.

To execute the training, it is necessary to set how many steps must be applied and how the learning rate changes during training step. All the parameters are presented in Table 18 and 19.

Batch size	Optimizer	Loss	Validation loss	Data augmentation algorithm
30	Gradient descent	RMSE	Weighted RMSE	RCS

Table 18 – Training parameters.

Step	0-5000	5001-5500	5501-6000	6001-6500	6501-7000
Learning rate	0.01	0.007	0.005	0.003	0.001

Table 19 – Training steps and learning rates.

5.2.1.8 Evaluation

The last set of parameters covert the evaluation process; the summary step interval is used to summarize how many training steps are applied until the training error is evaluated.

The eval step interval summarizes the interval which the validation error is evaluated. Table 20 shows their default values.

Summary step interval	Eval step interval
5	100

Table 20 – Evaluation properties.

5.2.2 Implementation

The implementation was conducted by using Python language, version 2.7.15, with Tensorflow framework version 1.11.0. The model building using this framework involves the creation of a graph; this is a connectionist representation of the model implementation. The Tensorflow graph of the proposed model, with the default parameters is shown in Fig. 25. The training involves the graph execution using the hardware infrastructure, more details can be found on Tensorflow documentation.

5.2.3 Training

To generate the objective speech quality scores used in the experimental methodology, the model was trained across the two databases presented in this chapter. The parameters used to train the model are the same already presented as default parameters. The training was executed along 7,000 steps with the decreasing of learning rate following the 19. Next are shown the learning curves, the training (in orange) and the validation (in blue) curves are showed for each database. In the vertical axis, it is presented the RMSE of training and the weighted RMSE of the validation set. In the graphics can be seen that the validation curve starts some steps later; more precisely, it starts at the 100th step. This was defined as an initial step, done to avoid printing large outliers at the beginning of the training phase. In the initial steps, it is notable the usual decaying of the curves with a high negative slope. In those steps, the model is starting from an initial random state to a poorly fitted state.

5.2.4 ITU-T Supplement 23 and TCD-VoIP

In the Fig.26 is shown the training and validation curves for database composed of ITU-T Suppl. 23 and TCD-VoIP databases. As can be seen, the two curves follow similar trends across the steps. The similar trend is more clearly shown in the bottom graphic, with the overlapping curves.

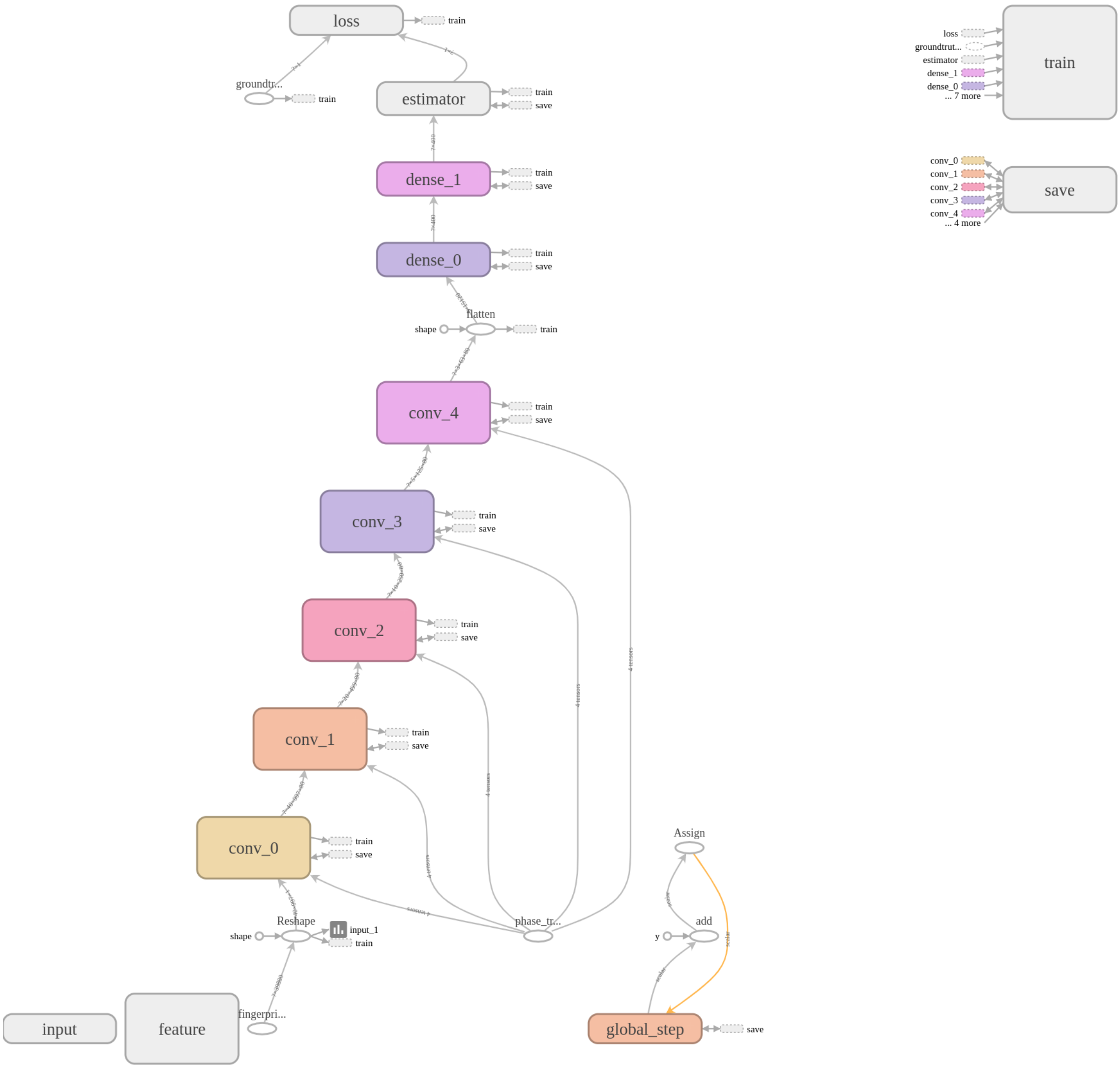


Figure 25 – Tensorflow graph of CNN-SQA model.

Source: Image generated by the author by using the Tensorboard tool (www.tensorflow.org/tensorboard/get_started)

Applying a smoothing in the curves of the Fig.26 are generated the corresponding curves in the Fig.27. The smoothed graphic version is better representation for visualize trends. It can be seen, the two curves began fairly high (both above the RMSE value of 1.5) and, as more training steps are performed, the curves find a plateau, with final RMSE values next to 0.5. This a expected behavior when training the machine learning models, indicating a good generalization fit in the validation set. A summary about RMSE values for the training and validation curves can be seen in Table 21.

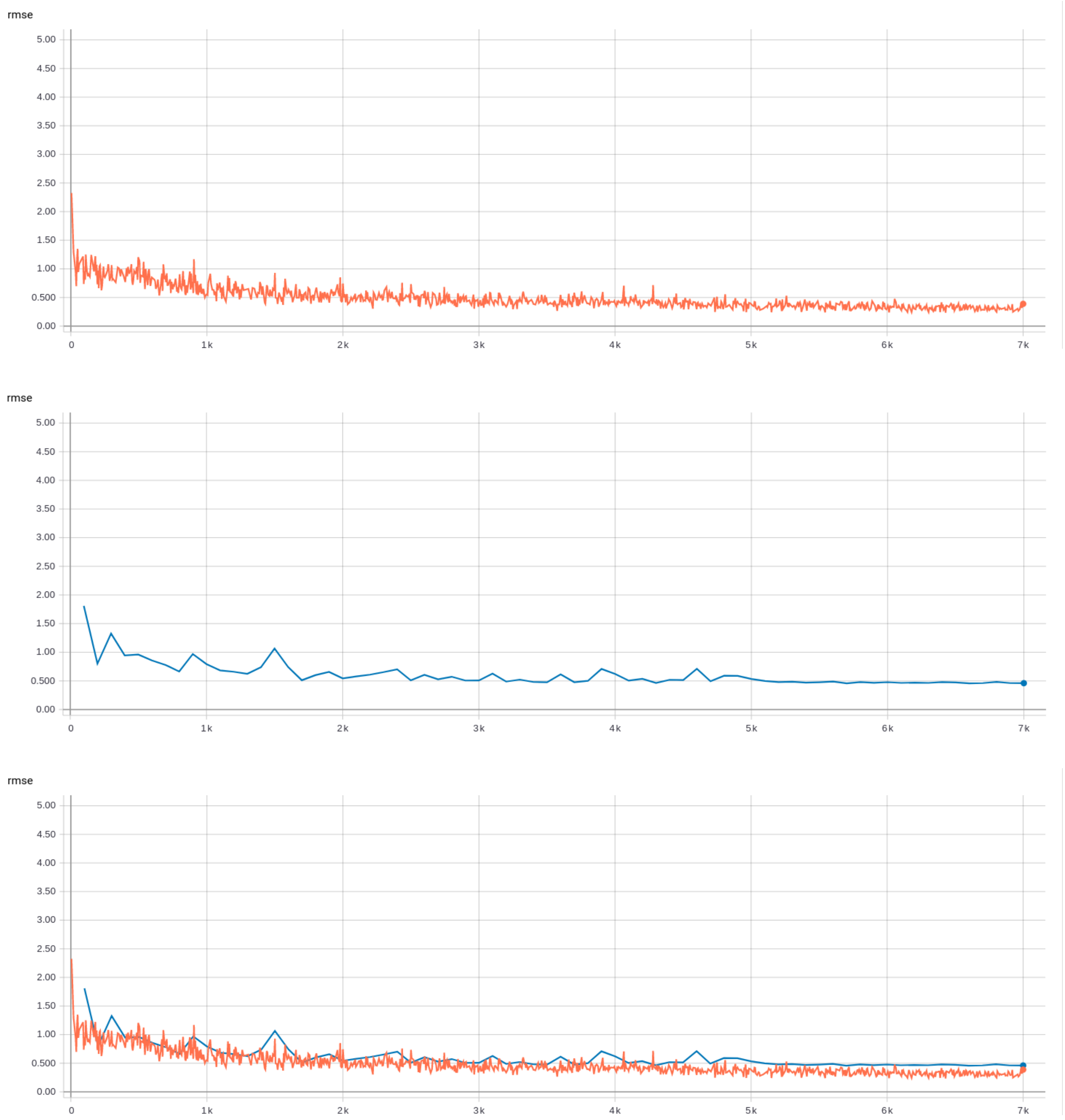


Figure 26 – The training and validation curves for ITU-T Supplement 23 and TCD-VoIP mixed database. On top the training curve, in the middle the validation curve and in the bottom is shown the overlapping curves.

Source: Image generated by the author by using the Tensorboard tool (www.tensorflow.org/tensorboard/get_started)

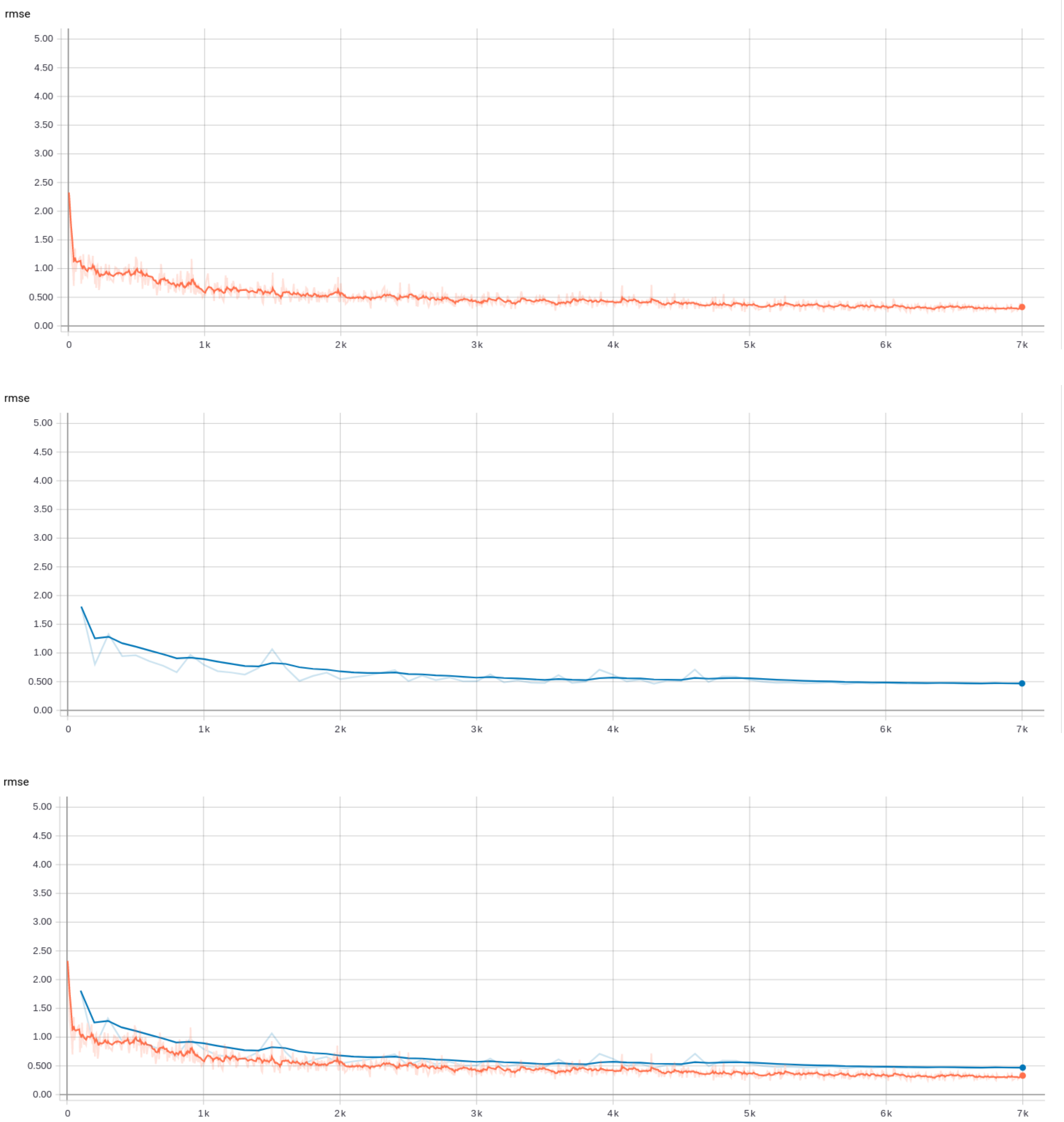


Figure 27 – The training and validation curves with smoothing of 0.75 for ITU-T Supplement 23 and TCD-VoIP mixed database. On top the training curve, in the middle the validation curve and in the bottom is shown the overlapping curves.

Source: Image generated by the author by using the Tensorboard tool (www.tensorflow.org/tensorboard/get_started)

Step	5	100	5,000	5,500	6,000	6,500	7,000
Learning rate	0.01	0.01	0.01	0.007	0.005	0.003	0.001
Training RMSE	2.32	0.96	0.40	0.28	0.35	0.35	0.38
Validation weighted RMSE	-	1.81	0.53	0.47	0.47	0.47	0.46

Table 21 – Summary of training and validation steps for ITU-T Supplement 23 and TCD-VoIP mixed database.

Source: Table generated by the author.

5.2.5 SNDSuppl23

In Fig.28, it is shown the training and validation curves for database SNDSuppl23. The curves begin at a high RMSE value (around 2.0) and goes down to find the plateau. As expected, the two curves follow similar trends across the steps, however the similar trends is more clearly shown when applying a smoothing.

Fig. 29 presents the smoothed version of the curves from Fig. 28. Differently of the learning curves for the fist database, the curves of Fig.29 shown the final RMSE values with a shorter gap between curves, and with lower RMSE values. This can be explained based on the total amount of samples involved in training. The SNDSuppl23 database has significantly more samples, with more variability on SNR conditions, than the dataset composed by the ITU-T Suppl.23 and TCD-VoIP databases. A summary about training steps can be seen in Table 22.

Step	5	100	5,000	5,500	6,000	6,500	7,000
Learning rate	0.01	0.01	0.01	0.007	0.005	0.003	0.001
Training RMSE	2.07	0.25	0.18	0.22	0.21	0.20	0.12
Validation weighted RMSE	-	1.95	0.19	0.34	0.22	0.24	0.19

Table 22 – Summary of training and validation steps for SNDSuppl23 database.

5.3 RESULTS

In this section are presented the experimental results when evaluating the CNN-SQA model on the two databases presented at the beginning of this chapter. The evaluation is addressed by comparing the CNN-SQA results with the publicly available state of the art methodologies: PESQ, ViSQOL and P563. These models were chosen because they represent standards in full-reference (PESQ) and non-reference (P563) methodologies. The use of ViSQOL model, in the experiments, was applied as an additional model to evaluate the results from

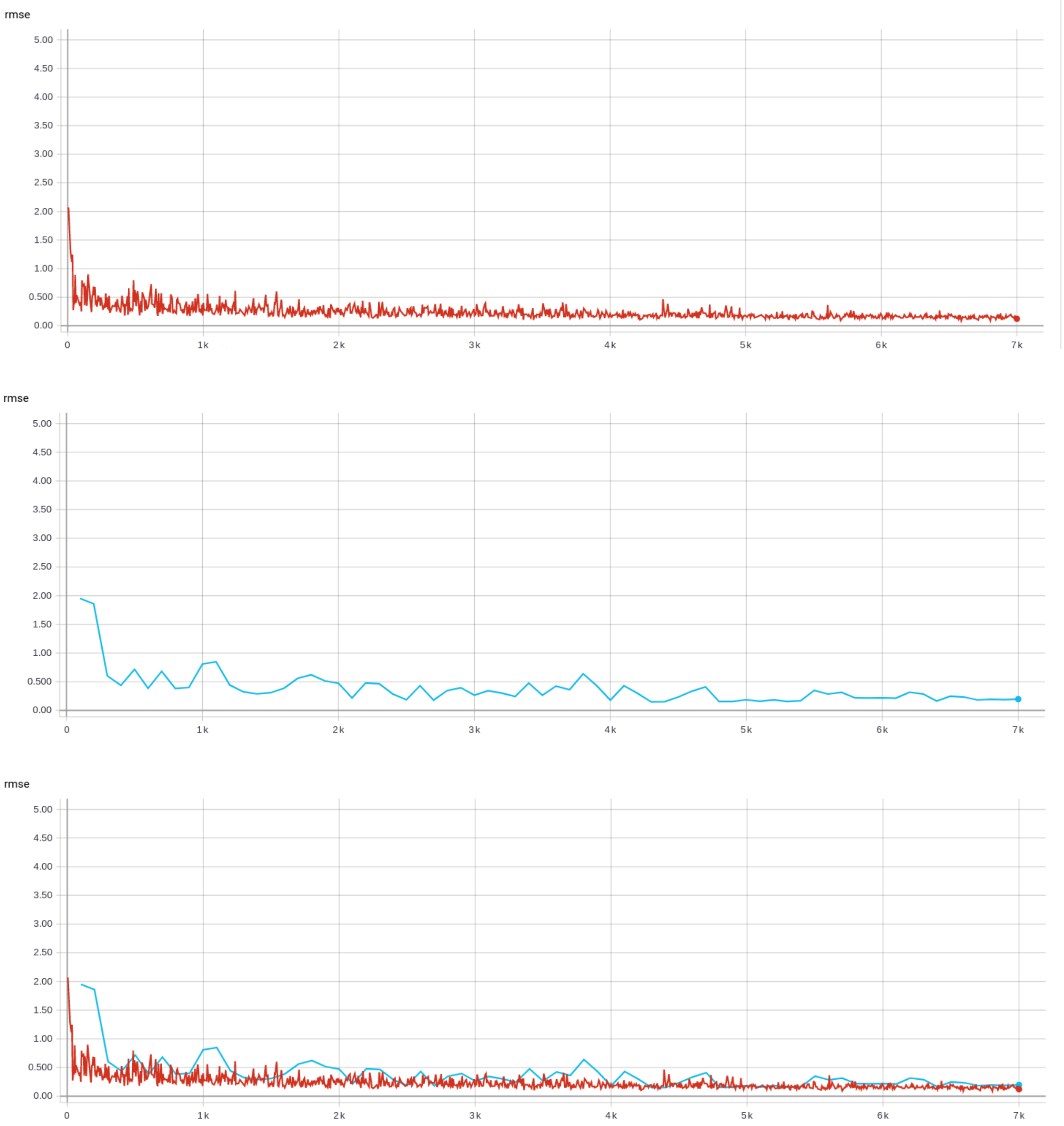


Figure 28 – The training and validation curves for SNDSuppl23 database. On top the training curve, in the middle the validation curve and in the bottom is shown the overlapping curves.

Source: Image generated by the author by using the Tensorboard tool (www.tensorflow.org/tensorboard/get_started)

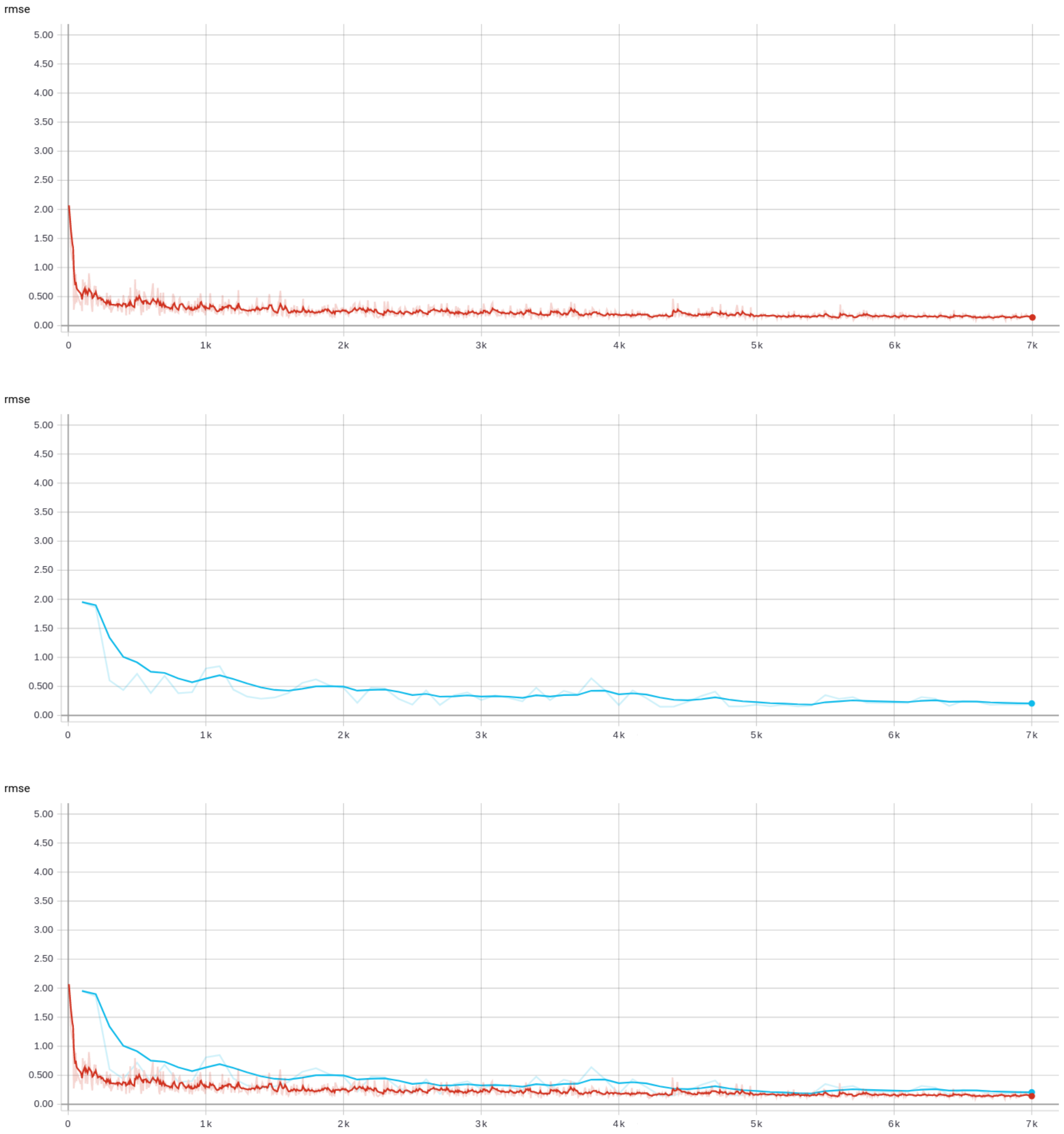


Figure 29 – The training and validation curves with smoothing of 0.75 for SNDSuppl23 database. On top the training curve, in the middle the validation curve and in the bottom is shown the overlapping curves.

Source: Image generated by the author by using the Tensorboard tool (www.tensorflow.org/tensorboard/get_started)

CNN-SQA model. Other models presented in this work, like as POLQA and ANIQUE, were not used in experiments because they are not publicly available.

5.3.1 Performance metrics

In the present work, the performance evaluation is conducted by using two metrics from (HU; LOIZOU, 2008). In that work, the metrics, named Pearson's correlation coefficient and the standard deviation of the error, are used to evaluate objective speech quality measures over noise speech enhanced samples.

The absolute values of Pearson's correlation coefficient ρ can be calculated by:

$$\rho = \frac{\sum_{i=1}^{n}(s_i - \overline{s})(o_i - \overline{o})}{\sqrt{\sum_{i=1}^{n}(s_i - \overline{s})^2}\sqrt{\sum_{i=1}^{n}(o_i - \overline{o})^2}} \tag{5.1}$$

where s_i and o_i are the equivalent values from two samples in index i; in this case, s can be treated as the reference signal (the subjective quality rate) and y the evaluated signal (the objective quality rate), the $\overline{s}$ and $\overline{o}$ are the average values of s and o, respectively. n is the maximum index; it represents the quantity of samples in the evaluated set. Pearson's correlation coefficient measures the strength and direction of a linear relationship between two sets of samples. Its values are limited to the range from -1 to 1; in that range a big positive coefficient represents a positive linear correlation, while a big negative value represents the inverse, a negative linear correlation. The zero value for Pearson's correlation means no linear relationships between the sets of samples.

The standard deviation of the error $\widehat{\sigma_e}$ can be calculated by:

$$\widehat{\sigma_e} = \widehat{\sigma_s}\sqrt{1-\rho^2} \tag{5.2}$$

where $\widehat{\sigma_s}$ is the standard deviation of the subjective quality scores and ρ is the Pearson's correlation coefficient. This metric represents a dispersion measure where big values can be interpreted as a high variability of the error.

5.3.2 ITU-T Supplement 23 and TCD-VoIP

In this subsection, it is presented the statistical analysis of the scores on validation set for the database composed by the ITU-T Suppl.23 and TCD-VoIP; more specifically, the ground truth mean opinion scores (MOS) and the scores predicted by CNN-SQA, and other models, are compared using statistical tools.

In Fig.30, it is presented the scatter plot for each model; it can be seen, by the color, the distribution for the two databases. The ViSQOL scores appear to be more spread in relation to the other results.

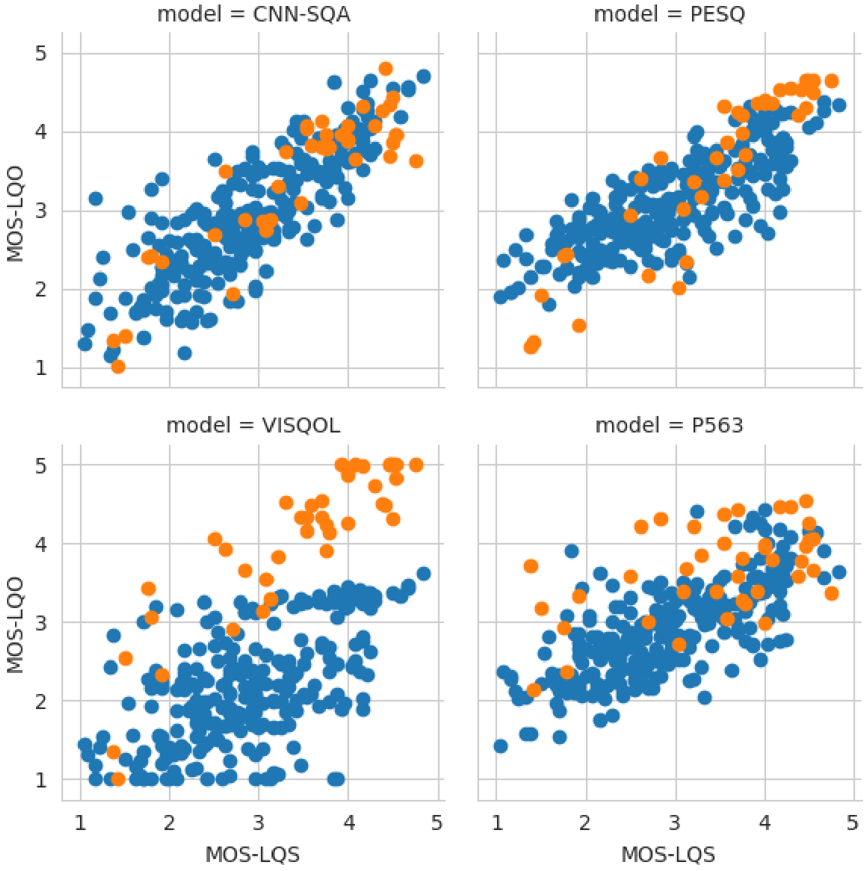


Figure 30 – Scatter results (ITU-T Suppl. 23 and TCD-VoIP). Objective measures against subjective MOS scores using ITU-T Suppl. 23 (Experiments 1 and 3, in blue color) and TCD-VOIP (background noise, clip and competing speakers, in orange color).

Source: Image generated by the author by using the Seaborn visualization library (seaborn.pydata.org/)

Applying a linear regression to the curves of the graphics in Fig.30, and plotting a reference regression ideal line, is shown in the Fig.31. In this graphics is seen that the regression line of the CNN-SQA model results are the most similar with the reference line. The PESQ and the P563 trends to overestimate the samples scores with low MOS-LQS. In another hand the ViSQOL mode trends to underestimate the samples as the MOS-LQS increases.

A table with the Pearson's correlations for each model is shown in the Table 23. As

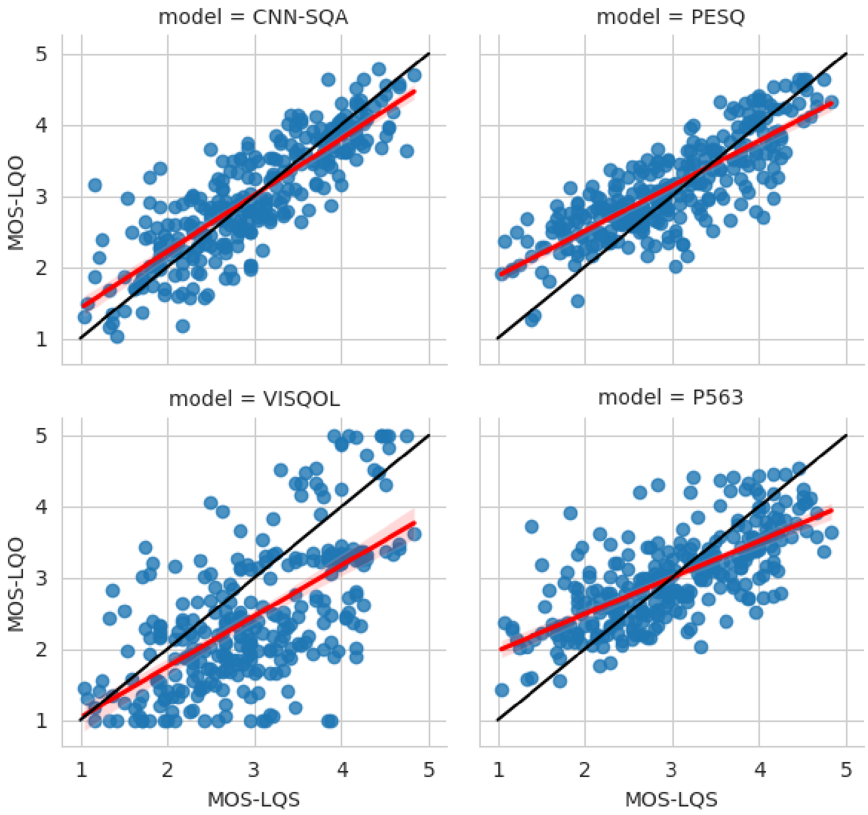


Figure 31 – Scatter results (ITU-T Suppl. 23 and TCD-VoIP). Objective measures against subjective MOS scores using ITU-T Suppl. 23 (Experiments 1 and 3) and TCD-VOIP (background noise, clip and competing speakers). The red lines are the regression lines generated and the black lines are the ideal reference lines.

Source: Image generated by the author by using the Seaborn visualization library (seaborn.pydata.org/)

suspected, the CNN-SQA model achieves the best performance in the mixed database. However, in the TCD-VoIP database, the PESQ correlation is bigger than CNN-SQA correlation by 0.02. This is explained by the amount of samples used from each database, in the validation set the percentage of samples used from each database is the same, however the total number of samples in database ITU-T Suppl. 23 is much greater than in database TCD-VoIP, making many more samples from the database ITU-T Suppl. 23 appear in the validation set.

Model	Mixed	Suppl23	TCD-VOIP
CNN-SQA	0.84	0.82	0.90
PESQ	0.82	0.79	0.92
ViSQOL	0.62	0.59	0.89
P563	0.70	0.71	0.52

Table 23 – Pearson correlation for all models on validation set using the database composed by samples from the ITU-T Suppl. 23 and TCD-VOIP datasets.

A table with the standard deviation for each model is shown in the Table 24. Again, a similar result, to that for Person's correlation, is observed; the CNN-SQA model achieves the best result in the mixed database, while in the TCD-VoIP database the PESQ standard deviation is lower than CNN-SQA result by 0.05.

Model	Mixed	Suppl23	TCD-VOIP
CNN-SQA	0.47	0.47	0.43
PESQ	0.50	0.51	0.38
ViSQOL	0.68	0.67	0.45
P563	0.62	0.58	0.83

Table 24 – Standard deviation for all models on validation set using the database composed by samples from the ITU-T Suppl. 23 and TCD-VOIP datasets.

To obtain the presented results the CNN-SQA was trained a few times showing little variation in each step.

5.3.3 SNDSuppl23

In this subsection, the scores generated by PESQ, treated as ground truth in the SNDSuppl23 database, and the scores predicted by the other models are compared using statistical tools. Because the existence of many conditions, generated by the permutation of noise type, SNR level and the quantity of speech samples available in database SNDSuppl23, a further analysis is conducted from a general overview to a categorized statistical description.

5.3.3.1 General Description

In the Fig.32 is presented the scatter plot for CNN-SQA model results. It can be seen the almost linear relation between objective and subjective MOS. Besides, because the subjective MOS are based on the PESQ scores, the maximum score in the graphic is limited by almost 4.

In Fig.33, it was applied a linear regression to the points of the graphic in Fig.32, and plotting a reference regression ideal line. In this graphic, it is seen that the regression line of the CNN-SQA model results are almost equal to the reference line, indicating a good fit of the model in the database SNDSuppl23.

Table 25 shows Pearson's correlation (ρ) and the standard deviation of the error ($\widehat{\sigma_e}$) for the models CNN-SQA, ViSQOL and P563. As we have suspected, the CNN-SQA model achieves the best performance in the SNDSuppl23 database. The results for ViSQOL and P563

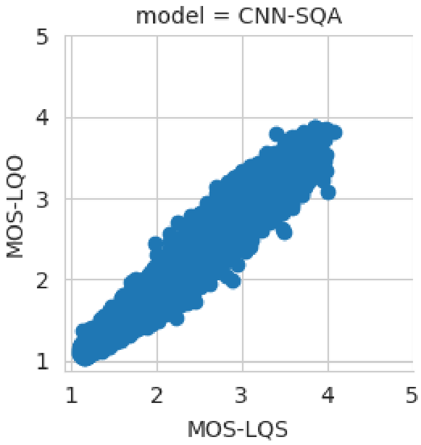


Figure 32 – Scatter plot of the scores for the CNN-SQA model when training with SND-Suppl23. Objective MOS (vertical) against subjective MOS (horizontal).

Source: Image generated by the author by using the Seaborn visualization library (seaborn.pydata.org/)

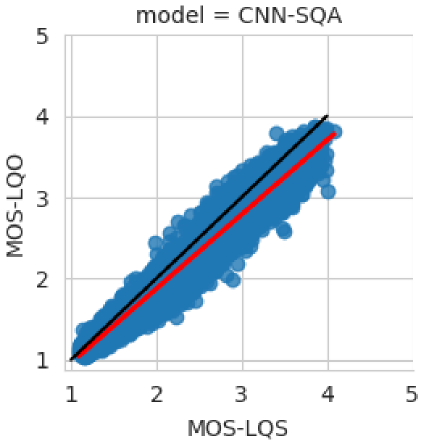


Figure 33 – Scatter plot of the scores for the CNN-SQA model when training with SND-Suppl23. Objective MOS (vertical) against subjective MOS (horizontal). The red line is a linear regression generated for shows the scores set, and the black line a reference line of a ideal linear regression result.

Source: Image generated by the author by using the Seaborn visualization library (seaborn.pydata.org/)

are presented just for reference, since this database was projected for training and analysis of the CNN-SQA model.

5.3.3.2 Categorized Description

When plotting the results in a scatter plot, Fig. 34, besides the visualization of an almost linear relation between the distributions, it is presented the color change with respect

Model	ρ	$\widehat{\sigma_e}$
CNN-SQA	0.98	0.15
ViSQOL	0.85	0.43
P563	0.65	0.62

Table 25 – Pearson correlation for all models on validation set using the SNDSuppl23.

to the SNR levels. It is notable the scores increasing when the SNR levels grows; an expected result since the active level of degradations is going down.

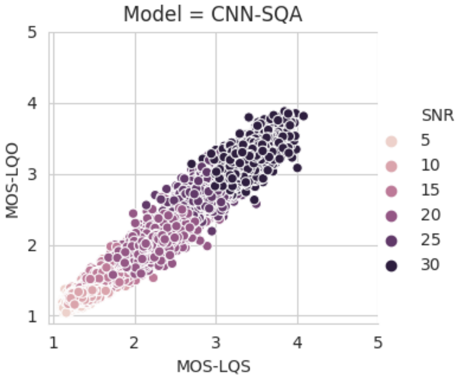


Figure 34 – The scatter plot with the samples colored by SNR.

Source: Image generated by the author by using the Seaborn visualization library (seaborn.pydata.org/)

As illustrated in Fig. 35, for almost all types of noise, the relation between predicted and Ground Truth (GT) scores seems linear and similar to the general overview. In the graphics, it is possible to visualize the increase of scores according to the increasing of SNR.

To get more insights about the underlying distributions, a KDE plot is used, comparing the ground truth (MOS-LQS) vs predicted scores (MOS-LQO), grouped by noise, as shown in Fig. 36. It can be seen that similar patterns are presented for both distributions. Between the scores range of 1 and 2 is found a peak for all noise types, this is explained by the low objective quality scores predicted by PESQ for low SNR.

The results indicate that the CNN-SQA model achieved better results compared to almost all other models evaluated. In comparison, in the first experiment, the PESQ methodology achieved the best results among all models in the subset of the TCD-VOIP database. On the other hand, when using the ITU-T Recommendation Supplement 23 database it was found that the CNN-SQA model achieved the best correlation rates. Finally, in the overall result the CNN-SQA model obtained the higher correlation results. Therefore, the results indicated a better fit of the CNN-SQA model when a more variable set of samples is available for training.

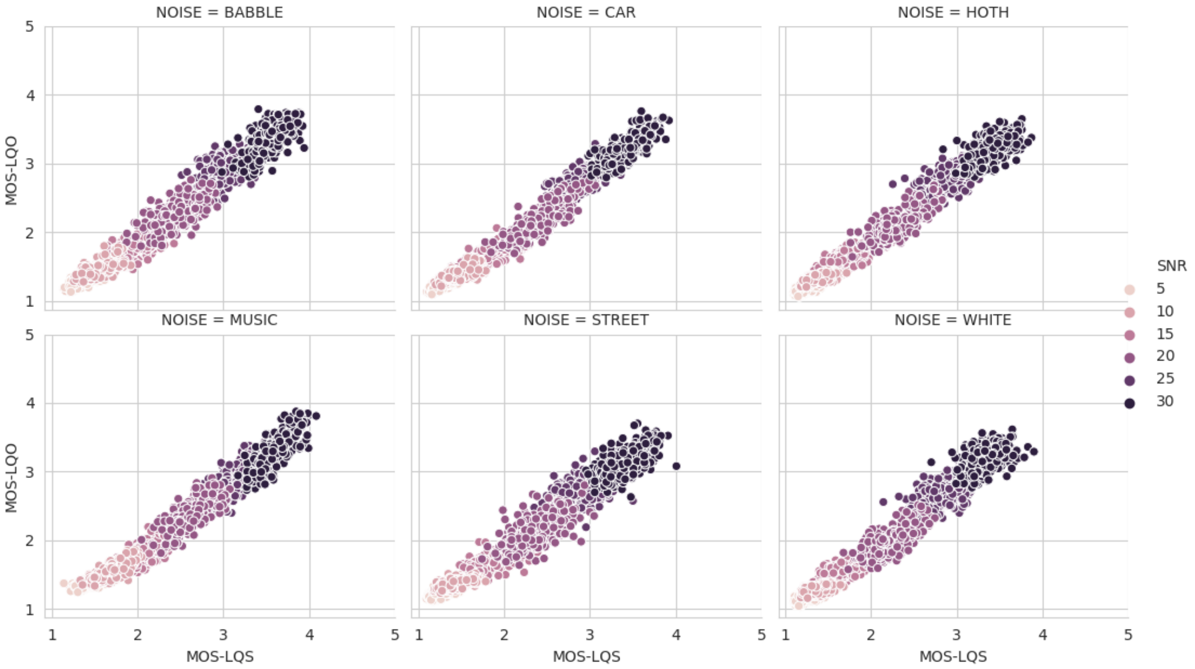


Figure 35 – The scatter plot of predictions by Ground Truth (GT) grouped by noise type and colored conform SNR increase.

Source: Image generated by the author by using the Seaborn visualization library (seaborn.pydata.org/)

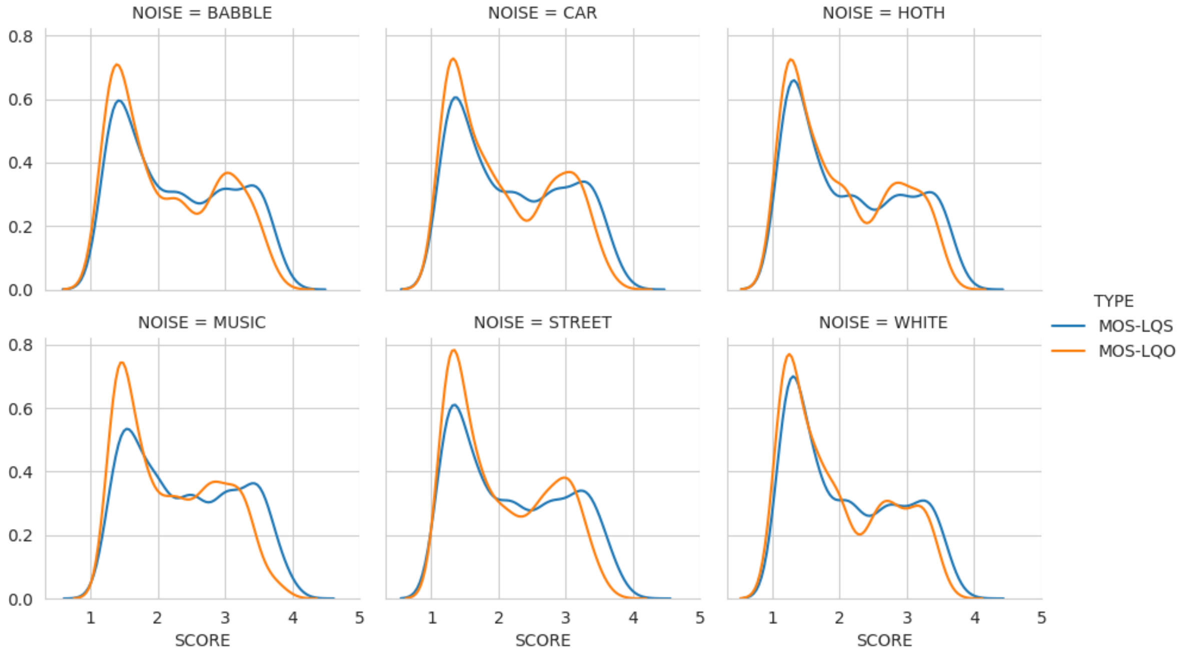


Figure 36 – The KDE plot of predictions by ground truth (GT) grouped by noise type.

Source: Image generated by the author by using the Seaborn visualization library (seaborn.pydata.org/)

This was observed in the analysis of the experiments, when comparing the MOS-LQO with the MOS-LQS, in which a better fit was observed for the model CNN-SQA. It was observed an almost linear relation between the predicted scores of of the CNN-SQA model and the subjective scores. Although, the predicted scores of the PESQ model have overestimated some subjective scores.

In the experiments with the SNDSuppl23 database, it was verified the behavior of the proposed model in a situation in which a huge quantity of samples was available for training. Besides, to evaluate the new model in the context of background noise, with variable SNR, it were used six different types of noises combined with six different values of SNR levels. Analysing the results, it was noted that an even higher disparity between results was reached, the proposed model outperformed the other methodologies with high correlated results. This can be assigned to the quantity and diversity of samples available, besides that the others models were trained with such database.

6 CONCLUSIONS

Speech understanding is a complex mechanism which is highly dependent on several aspects as speech quality. This is a subjective matter and it is difficult to measure, being necessary high demands of cost and time to do it properly. Although its complexity, it is of very important as many communication systems could use such measures to verify the quality of their services. There are different types of approaches to solve that problem in an automatic manner, including international standardized methodologies which in some scenarios get poor accuracy.

In this work, automatic speech quality assessment is investigated. Many of the existing concepts and some of the main methodologies of the area are presented throughout the work. As a matter of importance for many speech systems processing, the application of automatic speech quality assessment resulted in many standardized models which were defined as international recommendations, some of them being presented in this work. In the research context, this is an area in continuous evolution in which many studies are constantly elaborated.

Since the proposed model is formulated over the CNN architecture, this work get in touch with the advances in the DL area. More specifically, the main concepts involving the structure of CNN architectures are presented. Besides that, the classical convolutional architectures that were part of the evolution of the area are summarized.

One of the main problems when addressing speech quality, in a non-reference approach, is concerned on the proper characterization of speech and the underlying noise. Many of the current tools focus on solutions involving the psychoacoustic modeling, or in parameters extracted from the analysed system. Although there are great solutions to solve such problems, many of them generate bad results under noise conditions. In another direction, many of the problems involving speech processing are being modeled by DL techniques. Many of them employing the use of CNN architectures as their main feature extractors, sometimes serving to characterize speech and noise. Thus, it would be expected that the use of CNN architectures could be useful as solutions for problems involving speech quality modeling.

6.1 CONTRIBUTIONS

To evaluate the CNN architecture as a solution for speech quality assessment, it was proposed a new methodology called CNN-SQA. In this model, convolutional layers were used as feature extractors to address speech quality assessment in a non-reference approach. The new model, that can be trained in a supervisioned manner using speech databases, is publicly available and can be used as a reference model in speech quality experiments.

The problems involving the applicability of subjective tests, as time and cost, limit the creation and availability of new databases. Because of this, in areas as speech quality assessment there is high demand for open labelled databases. Thus, besides the proposal of a new methodology to address speech quality assessment, the present work has contributed with the built of the SNDSuppl23 database. It was built to evaluate the proposed model on conditions of background noise. Their corrupted samples were generated using samples from the ITU-T Recommendation Sumplement 23. Its conditions includes the presence of six different types of background noises mixed with six different types of SNR values. The SNDSuppl23 database was used in the experiments as an alternative to shortage of public datasets. It will be publicly available, serving not only as a reference dataset, to compare the performance of other speech quality assessment approaches with the CNN-SQA model, but as a bigger source of samples to use in experiments involving speech and background noise conditions.

In the first of two experiments, it was evaluated the performance of the proposed model, in comparison to other three models, when training it in a mix of public available databases. The databases conditions included the situation of few samples for training, with a diversified set of degradations, including distortions inserted by using speech CODECs and a subset of the VoIP degradations. It was shown that the use of an CNN based architecture reached better results in comparison with almost all other models evaluated. Although the proposed model has not achieved the best results on the subset of samples with VoIP degradations, the regression analysis, used to compare the real scores with the predicted ones, showed a better fit of the proposed model in an almost linear relation. These results indicates that a higher overall generalization was reached when employing convolutional layers as feature extractors. In this view, the use of convolutional layers, for speech and noise characterization, can be announced as a good choice in the context of speech quality assessment.

In the second experiment, in which it was used the SNDSuppl23 database, it was analysed the behavior of the proposed model in the context of background noise, in which a huge quantity of samples was available for training. It was noted that a higher disparity between results was reached, the proposed model outperformed the other methodologies with higher correlated results. This shows that the proposed model is able to characterize speech and background noise properly. Even for databases where complex degradations are present, such as babble noise in which speech samples are used as distortions, and street noise in which different noise types are mixed together, the use of the same convolutional architecture used in the first experiment can reach higher correlation rates. In any case, it could be noted that the use of a larger dataset was beneficial to increase the model accuracy.

Besides other approaches using CNN to solve speech quality assessment problems, this work is innovative in some ways. Firstly, in the present work were used MFCC coefficients as the main visual representation of the speech and noise as input for CNN layers aiming to solve the speech quality assessment problem. Besides, in this work was presented the use of fully-convolutional layers to extract the main features representing speech quality elements, this approach was beneficial to provide a good characterization of speech and noise. Lastly, the training and evaluation of the proposed model using standardized public databases making it possible to compare other models using the chosen databases.

6.2 FUTURE WORKS

Since the CNN-SQA presented in this work was an initial approach to verify the performance of a CNN architecture in the context of speech quality assessment in a controlled environment it is subject of improvement. Future works should be evaluated with a view of improving the CNN-SQA.

Firstly, the evaluation of other types of features as input for the CNN architecture should be done more deeply. Since the evaluation described in this work was in some view limited, it becomes necessary the evaluation of more features to characterize speech and noisy.

Another future research should be the investigation of the use of windowing analysis over the inputs of the CNN architecture. Instead of deal with full images of features, representing a complete speech sample as in the case of the CNN-SQA model, the use of windowing will force the feature extraction block (CNN layers) acts in a frame level quality assessment resolution. With this approach, it is expected that more localized features, representing speech and noise characteristics, could be extracted. It should be useful as a comparison between the use of full images versus the use of only frames to train the CNN model. Thus it should be useful a later comparison using a hybrid model taking the advantages of the two approaches.

Although the use of windowing is a classical approach in speech processing area, its use bring some difficulties to the training of CNN architectures. Firstly, it would be necessary the addition of some preprocessing steps for the input in order to apply the windowing, dividing the input images into frames of speech. It would be necessary the evaluation of the size in the time and frequency dimensions since there is no previous study appointing the beneficial use of smaller or bigger regions.

Besides the additional steps to apply windowing in the CNN architecture, it could be necessary the investigation of other types of loss functions more appropriated to work with windowing. Thus, it should be evaluated the use of different loss functions that be sensitive to local aspects concerned to speech quality.

Since the present work does not present an investigation on what is learned by the CNN-SQA model, an analysis should be conducted with the use of visualization techniques to investigate the CNN architecture weights. This investigation could be useful to indicate improvement points in the architectural design.

The comparison made between the CNN-SQA and the other models (PESQ, POLQA, ViSQOL and P563) was useful to verify the applicability of the proposed model in relation to the performance of popular models. As a future work, it should be useful the comparison in a broader group of methodologies. Some other models should be used, the works presented in (LO et al., 2019) and (AVILA et al., 2019) should be carefully studied and tested.

REFERENCES

ABDEL-HAMID, O. et al. Convolutional Neural Networks for Speech Recognition. **IEEE/ACM Transactions on Audio, Speech, and Language Processing**, v. 22, n. 10, p. 1533–1545, Oct. 2014. ISSN 2329-9304. DOI: 10.1109/TASLP.2014.2339736.

ABIODUN, Oludare et al. State-of-the-art in artificial neural network applications: A survey. **Heliyon**, v. 4, n. 11, e00938, Nov. 2018. ISSN 2405-8440. DOI: 10.1016/j.heliyon.2018.e00938.

AFFONSO, E. T.; ROSA, R. L.; RODRÍGUEZ, D. Z. Speech Quality Assessment Over Lossy Transmission Channels Using Deep Belief Networks. **IEEE Signal Processing Letters**, v. 25, n. 1, p. 70–74, Jan. 2018. ISSN 1558-2361. DOI: 10.1109/LSP.2017.2773536.

ANDERSEN, A. H. et al. Nonintrusive Speech Intelligibility Prediction Using Convolutional Neural Networks. **IEEE/ACM Transactions on Audio, Speech, and Language Processing**, v. 26, n. 10, p. 1925–1939, July 2018. ISSN 2329-9304. DOI: 10.1109/TASLP.2018.2847459.

ARORA, Sanjeev et al. Provable Bounds for Learning Some Deep Representations. In: XING, Eric P.; JEBARA, Tony (Eds.), 1. **Proceedings of the 31st International Conference on Machine Learning**. PMLR, Oct. 2014. v. 32. (Proceedings of Machine Learning Research, 1), p. 584–592. Available from: <http://proceedings.mlr.press/v32/arora14.html>.

AVILA, A. R. et al. Non-intrusive Speech Quality Assessment Using Neural Networks. In: ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). May 2019. P. 631–635. DOI: 10.1109/ICASSP.2019.8683175.

BENESTY, Jacob; SONDHI, M.; HUANG, Yiteng. **Springer Handbook of Speech Processing**. Springer-Verlag Berlin Heidelberg, Jan. 2008. ISBN 978-3-540-49125-5. DOI: 10.1007/978-3-540-49127-9.

BENGIO, Y.; GLOROT, X. Understanding the difficulty of training deep feedforward neural networks. In: PROCEEDINGS of the Thirteenth International Conference on Artificial Intelligence and Statistics. PMLR, May 2010. v. 9. (Proceedings of Machine Learning Research), p. 249–256. Available from: <http://proceedings.mlr.press/v9/glorot10a.html>.

CHAN, W. et al. Listen, attend and spell: A neural network for large vocabulary conversational speech recognition. In: 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). Mar. 2016. P. 4960–4964. DOI: `10.1109/ICASSP.2016.7472621`.

CHIU, C. et al. State-of-the-Art Speech Recognition with Sequence-to-Sequence Models. In: 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). Apr. 2018. P. 4774–4778. DOI: `10.1109/ICASSP.2018.8462105`.

CÔTÉ, Nicolas et al. An intrusive super-wideband speech quality model: DIAL. In: INTERSPEECH 2010, 11th Annual Conference of the International Speech Communication Association, Makuhari, Chiba, Japan, September 26-30, 2010. ISCA, Sept. 2010. P. 1317–1320. Available from: <`http://www.isca-speech.org/archive/interspeech%5C_2010/i10%5C_1317.html`>.

DAVIS, S.; MERMELSTEIN, P. Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences. **IEEE Transactions on Acoustics, Speech, and Signal Processing**, v. 28, p. 357–366, Jan. 1980. ISSN 0096-3518. DOI: `10.1109/TASSP.1980.1163420`.

DENG, J. et al. ImageNet: A large-scale hierarchical image database. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition. June 2009. P. 248–255. DOI: `10.1109/CVPR.2009.5206848`.

DUBEY, R. K.; KUMAR, A. Non-intrusive objective speech quality assessment using a combination of MFCC, PLP and LSF features. In: 2013 INTERNATIONAL CONFERENCE ON SIGNAL PROCESSING AND COMMUNICATION (ICSC). Dec. 2013. P. 297–302. DOI: `10.1109/ICSPCom.2013.6719801`.

FALK, T. H.; ZHENG, C.; CHAN, W. A Non-Intrusive Quality and Intelligibility Measure of Reverberant and Dereverberated Speech. **IEEE Transactions on Audio, Speech, and Language Processing**, v. 18, n. 7, p. 1766–1774, Oct. 2010. ISSN 1558-7924. DOI: `10.1109/TASL.2010.2052247`.

FU, Szu-Wei; TSAO, Yu; HWANG, Hsin-Te, et al. Quality-Net: An End-to-End Non-intrusive Speech Quality Assessment Model Based on BLSTM. In: PROC. Interspeech 2018. Sept. 2018. P. 1873–1877. DOI: `10.21437/Interspeech.2018-1802`.

FU, Szu-Wei; TSAO, Yu; LU, Xugang. SNR-Aware Convolutional Neural Network Modeling for Speech Enhancement. In: INTERSPEECH 2016. Sept. 2016. P. 3768–3772. DOI: `10.21437/Interspeech.2016-211`.

GOODFELLOW, Ian; BENGIO, Yoshua; COURVILLE, Aaron. **Deep Learning**. The MIT Press, 2016. ISBN 0262035618. Available from: <`http://www.deeplearningbook.org`>.

HARTE, Naomi; GILLEN, Eoin; HINES, Andrew. TCD-VoIP, a research database of degraded speech for assessing quality in VoIP applications. In: 2015 Seventh International Workshop on Quality of Multimedia Experience (QoMEX). July 2015. P. 1–6. DOI: `10.1109/QoMEX.2015.7148100`.

HE, K.; SUN, J. Convolutional neural networks at constrained time cost. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). June 2015. P. 5353–5360. DOI: `10.1109/CVPR.2015.7299173`.

HE, K.; ZHANG, X., et al. Deep Residual Learning for Image Recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). June 2016. P. 770–778. DOI: `10.1109/CVPR.2016.90`.

HINES, Andrew et al. ViSQOL: an objective speech quality model. **EURASIP Journal on Audio, Speech, and Music Processing**, v. 2015, May 2015. DOI: `10.1186/s13636-015-0054-9`.

HIRSCH, H. G. **FaNT**: Filtering and Noise Adding Tool. Mar. 2005. Available from: <`http://aurora.hsnr.de/download.html`>.

HU, Y.; LOIZOU, P. C. Evaluation of Objective Quality Measures for Speech Enhancement. **IEEE Transactions on Audio, Speech, and Language Processing**, v. 16, n. 1, p. 229–238, Feb. 2008. ISSN 1558-7924. DOI: `10.1109/TASL.2007.911054`.

HUBEL, D.H.; WEISEL, T.N. Receptive Fields and Functional Architecture of Monkey Striate Cortex. **The Journal of Physiology**, v. 195, p. 215–243, Mar. 1968. ISSN 0022-3751. DOI: `10.1113/jphysiol.1968.sp008455`.

HUBEL, David H.; WIESEL, Torsten N. Receptive Fields of Single Neurones in the Cat's Striate Cortex. **The Journal of physiology**, v. 148, p. 574–91, Oct. 1959. ISSN 0022-3751. DOI: `10.1113/jphysiol.1959.sp006308`.

IOFFE, Sergey; SZEGEDY, Christian. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. **CoRR**, abs/1502.03167, Feb. 2015. arXiv: 1502.03167. Available from: <http://arxiv.org/abs/1502.03167>.

ITU-T. **Recommendation P.800.1**: Mean opinion score (MOS) terminology. July 2016. Available from: <https://www.itu.int/rec/T-REC-P.800.1/en>.

ITU-T. **Recommendation P.861**: Objective quality measurement of telephone-band (300- 3400 Hz) speech codecs. Feb. 1998. Available from: <https://www.itu.int/rec/T-REC-P.861/en>.

ITU–T. **Recommendation G.191**: Software tools for speech and audio coding standardization. Jan. 2019. Available from: <https://www.itu.int/rec/T-REC-G.191/en>.

ITU–T. **Recommendation P.563**: Single ended method for objective speech quality assessment in narrowband telephony applications. May 2004. Available from: <https://www.itu.int/rec/T-REC-P.563/en>.

ITU–T. **Recommendation P.800**: Methods for Subjective Determination of Transmission Quality. Aug. 1996. Available from: <https://www.itu.int/rec/T-REC-P.800/en>.

ITU–T. **Recommendation P.830**: Subjective performance assessment of telephone-band and wideband digital codecs. Feb. 1996. Available from: <https://www.itu.int/rec/T-REC-P.830/en>.

ITU–T. **Recommendation 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. Nov. 2005. Available from: <https://www.itu.int/rec/T-REC-P.862>.

ITU–T. **Recommendation P.862.2**: Wideband Extension to Recommendation P.862 for the Assessment of Wideband Telephone Networks and Speech Codecs. Oct. 2017. Available from: <https://www.itu.int/rec/T-REC-P.862.2>.

ITU–T. **Recommendation P.862.3**: Application guide for objective quality measurement based on Recommendations P.862, P.862.1 and P.862.2. Nov. 2007. Available from: <https://www.itu.int/rec/T-REC-P.862.3/en>.

ITU–T. **Recommendation P.863**: Perceptual Objective Listening Quality Assessment. Jan. 2011. Available from: <https://www.itu.int/rec/T-REC-P.863>.

ITU–T. **Recommendation P.863**: Perceptual Objective Listening Quality Prediction. Jan. 2011. Available from: <https://www.itu.int/rec/T-REC-P.863>.

ITU–T. **Recommendation Supplement 23**: P.Sup23 : ITU-T coded-speech database. Feb. 1998. Available from: <`https://www.itu.int/rec/T-REC-P.Sup23/en`>.

JAUK, Igor et al. Expressive Speech Synthesis Using Sentiment Embeddings. In: INTERSPEECH 2018. International Speech Communication Association (ISCA), Sept. 2018. P. 3062–3066. DOI: `10.21437/Interspeech.2018-2467`.

JEKOSCH, Ute. **Voice and Speech Quality Perception: Assessment and Evaluation**. Berlin, Heidelberg: Springer-Verlag Berlin Heidelberg, 2005. ISBN 978-3-540-24095-2. DOI: `10.1007/3-540-28860-0`.

JELASSI, S. et al. Quality of Experience of VoIP Service: A Survey of Assessment Approaches and Open Issues. **IEEE Communications Surveys Tutorials**, v. 14, n. 2, p. 491–513, Feb. 2012. ISSN 1553-877X. DOI: `10.1109/SURV.2011.120811.00063`.

JUNIOR, Franciscone; ROSA, Renata; RODRIGUEZ, Demostenes Zegarra. Voice Quality Assessment in Communication Services using Deep Learning. In: 2018 15th International Symposium on Wireless Communication Systems (ISWCS). Aug. 2018. P. 1–6. DOI: `10.1109/ISWCS.2018.8491055`.

KIM, Chanwoo; STERN, Richard. Power-Normalized Cepstral Coefficients (PNCC) for Robust Speech Recognition. **IEEE/ACM Transactions on Audio, Speech, and Language Processing**, v. 24, n. 7, p. 1315–1329, July 2016. ISSN 2329-9304. DOI: `10.1109/TASLP.2016.2545928`.

KIM, Doh-Suk. A cue for objective speech quality estimation in temporal envelope representations. **IEEE Signal Processing Letters**, v. 11, n. 10, p. 849–852, Nov. 2004. ISSN 1558-2361. DOI: `10.1109/LSP.2004.835466`.

KIM, Doh-Suk. ANIQUE: An auditory model for single-ended speech quality estimation. **IEEE Transactions on Speech and Audio Processing**, v. 13, n. 5, p. 821–831, Sept. 2005. ISSN 1558-2353. DOI: `10.1109/TSA.2005.851924`.

KRIZHEVSKY, Alex; SUTSKEVER, Ilya; HINTON, Geoffrey E. ImageNet Classification with Deep Convolutional Neural Networks. In: PEREIRA, F. et al. (Eds.). **Advances in Neural Information Processing Systems 25**. Curran Associates, Inc., Jan. 2012. P. 1097–1105. DOI: `10.1145/3065386`. Available from: <`http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf`>.

LAPIDUS, M.; SHALLOM, I. Enhanced intrusive Voice Quality Estimation (EVQE), p. 000476–000480, Nov. 2010. DOI: 10.1109/EEEI.2010.5662174.

LECUN, Y. et al. Gradient-based learning applied to document recognition. **Proceedings of the IEEE**, v. 86, n. 11, p. 2278–2324, Nov. 1998. ISSN 1558-2256. DOI: 10.1109/5.726791.

LIN, Haoning; SHI, Zhenwei; ZOU, Zhengxia. Maritime Semantic Labeling of Optical Remote Sensing Images with Multi-Scale Fully Convolutional Network. **Remote Sensing**, v. 9, p. 480, May 2017. DOI: 10.3390/rs9050480.

LIN, Min; CHEN, Qiang; YAN, Shuicheng. **Network In Network**. Dec. 2013. arXiv: 1312.4400. Available from: <http://arxiv.org/abs/1312.4400>.

LO, Chen - Chou et al. MOSNet: Deep Learning-Based Objective Assessment for Voice Conversion. In: abs/1904.08352, p. 1541–1545. DOI: 10.21437/Interspeech.2019-2003. arXiv: 1904.08352. Available from: <http://arxiv.org/abs/1904.08352>.

MCLOUGHLIN, I. **Applied Speech and Audio Processing With Matlab Examples**. Cambridge University Press, Jan. 2009. ISBN 9780521519540. DOI: 10.1017/CBO9780511609640.

MELVIN, H.; MURPHY, L. Time synchronization for VoIP quality of service. **IEEE Internet Computing**, v. 6, n. 3, p. 57–63, May 2002. ISSN 1941-0131. DOI: 10.1109/MIC.2002.1003132.

MIKOŁAJCZYK, A.; GROCHOWSKI, M. Data augmentation for improving deep learning in image classification problem. In: 2018 International Interdisciplinary PhD Workshop (IIPhDW). May 2018. P. 117–122. DOI: 10.1109/IIPHDW.2018.8388338.

MÖLLER, S. et al. Speech Quality Estimation: Models and Trends. **IEEE Signal Processing Magazine**, v. 28, n. 6, p. 18–28, Nov. 2011. ISSN 1558-0792. DOI: 10.1109/MSP.2011.942469.

MÖLLER, Sebastian. **Assessment and Prediction of Speech Quality in Telecommunications**. Springer US, 2000. ISBN 9780792378945. Available from: <https://books.google.com.br/books?id=p7AXOFTSNu4C>.

MÖLLER, Sebastian. **Quality of Telephone-Based Spoken Dialogue Systems**. Springer, Jan. 2005. ISBN 978-0-387-23190-7. DOI: 10.1007/b100796.

NAIR, Vinod; HINTON, Geoffrey. Rectified Linear Units Improve Restricted Boltzmann Machines. In: PROCEEDINGS of the 27th International Conference on International Conference on Machine Learning. Omnipress, June 2010. v. 27. (ICML'10), p. 807–814. ISBN 9781605589077.

NORSKOG, L. **SoX**: Sound eXchange. Dec. 2014. Available from: <`http://sox.sourceforge.net/Docs/Documentation`>.

PARK, Daniel S.; CHAN, William, et al. SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition. **Interspeech 2019**, ISCA, Sept. 2019. DOI: `10.21437/interspeech.2019-2680`.

PARK, Serim; LEE, Jin. A Fully Convolutional Neural Network for Speech Enhancement. In: PROC. Interspeech 2017. Aug. 2017. P. 1993–1997. DOI: `10.21437/Interspeech.2017-1465`.

PAVLOVIC, C. SII—Speech intelligibility index standard: ANSI S3.5 1997. **The Journal of the Acoustical Society of America**, v. 143, n. 3, p. 1906–1906, Mar. 2018. DOI: `10.1121/1.5036206`.

QIAO, Z.; SUN, L.; IFEACHOR, E. Case study of PESQ performance in live wireless mobile VoIP environment. In: 2008 IEEE 19th International Symposium on Personal, Indoor and Mobile Radio Communications. Sept. 2008. P. 1–6. DOI: `10.1109/PIMRC.2008.4699880`.

RAAKE, A. **Speech Quality of VoIP: Assessment and Prediction**. Nov. 2006. P. 1–309. ISBN 9780470033005. DOI: `10.1002/9780470033005`.

ROZHON, J. et al. A new approach to speech quality assessment based on back-propagation neural networks. **International Journal of Circuits, Systems and Signal Processing**, v. 10, 2-s2.0-84957017651, p. 52–61, Jan. 2016. ISSN 1998-4464. Available from: <`https://www.scopus.com/record/display.uri?eid=2-s2.0-84957017651&origin=inward&txGid=51a1f1a3b0edb0dab058c9211f25cbe5`>.

RUDER, S. An overview of gradient descent optimization algorithms. **CoRR**, abs/1609.04747, 2016. arXiv: `1609.04747`. Available from: <`http://arxiv.org/abs/1609.04747`>.

SERRE, T. et al. Robust Object Recognition with Cortex-Like Mechanisms. **IEEE transactions on pattern analysis and machine intelligence**, v. 29, n. 3, p. 411–26, Mar. 2007. ISSN 1939-3539. DOI: `10.1109/TPAMI.2007.56`.

SONI, M. H.; PATIL, H. A. Novel deep autoencoder features for non-intrusive speech quality assessment. In: 2016 24th European Signal Processing Conference (EUSIPCO). Aug. 2016. P. 2315–2319. DOI: `10.1109/EUSIPCO.2016.7760662`.

SRIVASTAVA, Nitish et al. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. **Journal of Machine Learning Research**, JMLR.org, v. 15, n. 56, p. 1929–1958, June 2014. ISSN 1532-4435. Available from: <`http://jmlr.org/papers/v15/srivastava14a.html`>.

STEVENS, S. S.; VOLKMANN, J.; NEWMAN, E. B. A scale for the measurement of the psychological magnitude pitch. **Journal of the Acoustical Society of America**, v. 8, n. 3, p. 185–190, Jan. 1937. DOI: `10.1121/1.1915893`.

SZEGEDY, C. et al. Going deeper with convolutions. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). June 2015. P. 1–9. DOI: `10.1109/CVPR.2015.7298594`.

TAAL, C. H. et al. An Algorithm for Intelligibility Prediction of Time–Frequency Weighted Noisy Speech. **IEEE Transactions on Audio, Speech, and Language Processing**, v. 19, n. 7, p. 2125–2136, Oct. 2011. ISSN 1558-7924. DOI: `10.1109/TASL.2011.2114881`.

TORAL-CRUZ, H. et al. **An Introduction to VoIP: End-to-End Elements and QoS Parameters**. Ed. by Shigeru Kashihara. Rijeka: IntechOpen, Feb. 2011. chap. 4. DOI: `10.5772/13520`. Available from: <`https://doi.org/10.5772/13520`>.

UPADHYAY, Navneet; KARMAKAR, Abhijit. Speech Enhancement using Spectral Subtraction-type Algorithms: A Comparison and Simulation Study. **Procedia Computer Science**, v. 54, p. 574–584, Dec. 2015. ISSN 1877-0509. DOI: `10.1016/j.procs.2015.06.066`.

VÉSTIAS, Mário. A Survey of Convolutional Neural Networks on Edge with Reconfigurable Computing. **Algorithms**, v. 12, p. 154, July 2019. DOI: `10.3390/a12080154`.

WANG, R. et al. No-reference Speech Quality Assessment of SWB Signal Based on Machine Learning. In: 2018 15th International Symposium on Wireless Communication Systems (ISWCS). Aug. 2018. P. 1–5. DOI: `10.1109/ISWCS.2018.8491197`.

WEI, Zhili et al. Characterizing Rock Facies Using Machine Learning Algorithm Based on a Convolutional Neural Network and Data Padding Strategy. **Pure and Applied Geophysics**, v. 176, p. 3593–3605, Aug. 2019. ISSN 1420-9136. DOI: `10.1007/s00024-019-02152-0`.

WU, Z.; CAO, Z. Improved MFCC-based feature for robust speaker identification. **Tsinghua Science and Technology**, v. 10, n. 2, p. 158–161, Apr. 2005. ISSN 1007-0214. DOI: `10.1016/S1007-0214(05)70048-1`.

XIE, S. et al. Deep Neural Networks for Voice Quality Assessment Based on the GRBAS Scale. In: INTERSPEECH 2016. ISCA, Sept. 2016. P. 2656–2660. DOI: `10.21437/Interspeech.2016-986`.

YANG, H. et al. Parametric-based non-intrusive speech quality assessment by deep neural network. In: 2016 IEEE International Conference on Digital Signal Processing (DSP). Oct. 2016. P. 99–103. DOI: `10.1109/ICDSP.2016.7868524`.

ZEILER, M. D.; FERGUS, R. Visualizing and Understanding Convolutional Networks. In: COMPUTER Vision, ECCV 2014 - 13th European Conference, Proceedings. Springer Verlag, Jan. 2014. v. 8689. (Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)), p. 818–833. ISBN 9783319105895. DOI: `10.1007/978-3-319-10590-1_53`.