FAKULTA MECHATRONIKY,
INFORMATIKY A MEZIOBOROVÝCH
STUDIÍ TUL

Bakalářská práce

Non-intrusive speech quality assessment

Studijní program:	B0613A140005 Informační technologie
Studijní obor:	Inteligentní systémy
Autor práce:	**Viktoriia Sergeeva**
Vedoucí práce:	doc. Ing. Jiří Málek, Ph.D. Ústav informačních technologií a elektroniky

Liberec 2024

FAKULTA MECHATRONIKY,
INFORMATIKY A MEZIOBOROVÝCH
STUDIÍ TUL

Zadání bakalářské práce

Non-intrusive speech quality assessment

Jméno a příjmení:	**Viktoriia Sergeeva**
Osobní číslo:	M20000072
Studijní program:	B0613A140005 Informační technologie
Specializace:	Inteligentní systémy
Zadávající katedra:	Ústav informačních technologií a elektroniky
Akademický rok:	2022/2023

Zásady pro vypracování:

The bachelor thesis aims to develop a method for the estimation of selected speech signal quality metrics. The method must be applicable without any reference non-distorted signal (i.e., evaluation in a non-intrusive manner).

1. Determine the acoustic conditions and distortions that the work will focus on (e.g., low reverberation noisy environment for estimation of noise levels)
2. Determine the metrics to be estimated (e.g., SNR, STOI – for noise measurement).
3. Propose a method for the estimation of these metrics based on machine learning principles.
4. On a suitable test dataset, compare the accuracy of the non-intrusive estimation versus the conventional evaluation with reference.
5. Analyze the robustness of the estimator with respect to some selected design parameters (architecture or hyperparameters of the neural network) or acoustic environment (e.g., noise level, noise type, reverberation level).

Rozsah grafických prací:	Dle potřeby dokumentace
Rozsah pracovní zprávy:	30-40 stran
Forma zpracování práce:	tištěná/elektronická
Jazyk práce:	angličtina

Seznam odborné literatury:

[1] SEETHARAMAN, Prem, Gautham J. MYSORE, Paris SMARAGDIS a Bryan PARDO. Blind Estimation of the Speech Transmission Index for Speech Quality Prediction. In: 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) [online]. IEEE, 2018, 2018, s. 591-595 [cit. 2022-09-22]. ISBN 978-1-5386-4658-8. Dostupné z: doi:10.1109/ICASSP.2018.8461827

[2] SHARMA, Dushyant, Lucia BERGER, Carl QUILLEN a Patrick A. NAYLOR. Non-Intrusive Estimation of Speech Signal Parameters using a Frame-based Machine Learning Approach. In: 2020 28th European Signal Processing Conference (EUSIPCO) [online]. IEEE, 2021, 2021-1-24, s. 446-450 [cit. 2022-09-22]. ISBN 978-9-0827-9705-3. Dostupné z: doi:10.23919/Eusipco47968.2020.9287785

[3] AVILA, Anderson R., Hannes GAMPER, Chandan REDDY, Ross CUTLER, Ivan TASHEV a Johannes GEHRKE. Non-intrusive Speech Quality Assessment Using Neural Networks. In: ICASSP 2019 – 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) [online]. IEEE, 2019, 2019, s. 631-635 [cit. 2022-09-22]. ISBN 978-1-4799-8131-1. Dostupné z: doi:10.1109/ICASSP.2019.8683175

[4] GAMPER, Hannes, Chandan K A REDDY, Ross CUTLER, Ivan J. TASHEV a Johannes GEHRKE. Intrusive and Non-Intrusive Perceptual Speech Quality Assessment Using a Convolutional Neural Network. In: 2019 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA) [online]. IEEE, 2019, 2019, s. 85-89 [cit. 2022-09-22]. ISBN 978-1-7281-1123-0. Dostupné z: doi:10.1109/WASPAA.2019.8937202

Vedoucí práce:	doc. Ing. Jiří Málek, Ph.D. Ústav informačních technologií a elektroniky
Datum zadání práce:	24. října 2022
Předpokládaný termín odevzdání:	14. května 2024

L.S.

prof. Ing. Zdeněk Plíva, Ph.D.
děkan

doc. Ing. Josef Chaloupka, Ph.D.
garant studijního programu

V Liberci dne 24. října 2022

Prohlášení

Prohlašuji, že svou bakalářskou práci jsem vypracovala samostatně jako původní dílo s použitím uvedené literatury a na základě konzultací s vedoucím mé bakalářské práce a konzultantem.

Jsem si vědoma toho, že na mou bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci nezasahuje do mých autorských práv užitím mé bakalářské práce pro vnitřní potřebu Technické univerzity v Liberci.

Užiji-li bakalářskou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědoma povinnosti informovat o této skutečnosti Technickou univerzitu v Liberci; v tomto případě má Technická univerzita v Liberci právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Současně čestně prohlašuji, že text elektronické podoby práce vložený do IS/STAG se shoduje s textem tištěné podoby práce.

Beru na vědomí, že má bakalářská práce bude zveřejněna Technickou univerzitou v Liberci v souladu s § 47b zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů.

Jsem si vědoma následků, které podle zákona o vysokých školách mohou vyplývat z porušení tohoto prohlášení.

7. května 2024 Viktoriia Sergeeva

Vyhodnocení kvality řečového signálu bez reference

Abstrakt

Tato bakalářská práce se zabývá odhadem metrik pro kvalitu řečového signálu bez reference. Odhad je založen na realných datech a je postaven na vytvoření neuronové sítě. Konkrétně odhadujeme odstup signálu od šumu, čas dozvuku T60 a srozumitelnost řeči metrikou STOI (Short-Time Objective Intelligibility). Práce porovnává odhad těchto metrik s a bez reference. Model je testován na různých datových sadách a je diskutována přesnost odhadu. Odhad kvality řečového signálu bez reference umožňuje přenést měření kvality řečového signálu do praktických situací mimo umělé laboratorní podmínky.

Klíčová slova: odhad metrik bez reference, kvalita řeči, neuronové sítě, zpracování signálu, metriky kvality

Non-intrusive speech quality assessment

Abstract

This bachelor thesis deals with the estimation of metrics for speech signal quality without reference. Estimation is based on real data and implemented through a neural network. Specifically, we estimate signal-to-noise ratio, T60 reverberation time, and speech intelligibility using the STOI (Short-Time Objective Intelligibility) metric. The work compares estimation of the metrics with and without reference. The model is tested on different datasets, and the estimation accuracy is discussed. The estimation of speech signal quality without reference enables to transfer the measurement of speech signal quality to practical situations outside artificial laboratory conditions.

Keywords: estimation of metrics without reference, speech quality, neural networks, signal processing, quality metrics

Acknowledgements

I am heartfelt grateful to my supervisor Ing. Jiri Malek, for his patience, thorough explanation of the materials and all the help with perfecting this bachelor thesis.

Contents

List of abbreviations

SNR	Signal To Noise Ratio
STOI	Short-Time Objective Intelligibility
WADA	Waveform Amplitude Distribution Analysis
DSP	Digital signal processing
FFT	Fast Fourier transform
STFT	Short-Time Fourier Transform
RMSE	Root Mean Squared Error
MSE	Mean squared error
Noisy-En	Noisy English dataset
Noisy-Ge	Noisy German dataset
Noisy-Ko	Noisy Korean dataset
Reverb-En	Reverberated English dataset
Combi-En	Combinated English dataset

List of Figures

List of Tables

1 Introduction and motivation

The world has changed quickly; people become surrounded by more than just computers. Today, artificial intelligence does not just imitate human thinking—it expands its capabilities. It analyses problems and comes up with a solution that a person could offer. Thanks to artificial intelligence, previously time-consuming and challenging tasks have been accelerated and simplified. This has its positive and negative sides, let me focus on the positive ones mainly in this work. The importance of audio signals cannot be overestimated — they inform about the world around, provide entertainment, provide security, and help with communication. The variety of sounds encounter daily needs complex processing to get helpful information and filter out the noise.

A signal is a function of one or more independent variables. Digital signal processing (DSP) is a way of processing signals with a representation of a sequence of numbers based on digital methods using digital computing technology. DSP prefabricated concept for rearranging signals using operations such as analysing, processing, modifying and synthesising to improve their usefulness. The frequent tasks of audio processing are for example speech recognition, which aims to transcribe speech recording into text or event detection, which aims to find some essential sounds such as a siren or a gun shot. In the context of speech processing, it is useful to evaluate the quality of the signal. Processing of undistorted signals provides higher confidence about the results and might decrease complexity/cost of any involved algorithms. The algorithmic evaluation of speech quality is carried out through metrics such as Signal-to-Noise Ratio (SNR), Intelligibility of Speech recognition (STOI) and reverberation level (T60).

Traditionally, these metrics require a reference undistorted speech. This signal is used to compare and estimate distortion in the processed audio. In practice, a reference signal is often unavailable, which limits these metrics usability to laboratory conditions where artificial mixtures of sounds can be accurately controlled and reproduced. Moreover, non-intrusive solution does not require a reference signal, so it is independent on artificial and laboratory-created signals. The creation and subsequent expansion of neural network-based estimator may facilitate evaluation of speech signals under wide range of conditions.

The absence of a reference signal offers both challenges and opportunities. The obvious benefit is the cessation of dependence on practically unavailable speech and noise components, which will help to increase the usage of these metrics in practical applications. The lack of reference signals means that the quality of system will depend on an initial set of training and evaluation data - the vulnerability of the

data in training may lead to a tendency to estimate metrics in practice incorrectly.

In the first section of work, there is a discussion about the used metrics from the general point of view and comparison of intrusive and non-intrusive methods for their measurement. It also includes briefly discussion about existing methods for non-intrusive metric estimation and introduce the solution. The second section describes augmentation of training data for chosen data-driven solution and proposes the network architecture of the estimator. The third part of the thesis describes origin of the speech and noise data used, design of the experiments and methods and metrics used for their evaluation. The fourth part concludes the thesis.

1.1 Metrics

- **SNR**

  Signal-to-noise Ratio is an objective measure for speech quality assessment in noisy conditions. It is an objective measure, which does not take any human perception into consideration. This dimensionless value is equal to the ratio of the useful signal power to the noise power. We measure and know the signal x, which consists of the (in practice unknown) components s and v according to the formula $x = s + v$, where s is the useful speech signal and v is the unwanted noise.

  Based on these components SNR is calculated by the following formula:

  $$S = 10 \cdot log_{10} \frac{\sigma_s^2}{\sigma_v^2}, \tag{1.1}$$

  where σ_s^2 is an average useful signal power, σ_v^2 is an average noise signal power. In this form it is expressed in decibels.

  It's important to mention that that the aforementioned components are available only under laboratory conditions. In practical applications, the components are unavailable and must be estimated.

- **STOI** Short-Time Objective Intelligibility (STOI) is an objective metric for measuring speech quality under noisy and reverberant conditions. It approximate speech intelligibility. It attempts to compare clean and distorted/noisy speech signal and quantifies their similarity into range 0 to 1. High STOI corresponds to intelligible undistorted speech signal and vice versa.

  The computation of STOI is based on the following steps [1]. The analysed speech is first decomponsed into DFT-based one-third octave bands. The next step is to compare the short-time (384 ms) temporal envelope segments of the clean and degraded speech using a correlation coefficient. Before comparison, the short-time envelopes of the degraded speech are first normalised and clipped. These short-time intermediate measures of speech intelligibility are then averaged to produce a scalar value that is expected to have a monotonically increasing relationship with the intelligibility of the speech.

- **T60** Metric T60 is a measurement of reverberation. Reverberation is the continuation of sound in a space after a sound source has stopped due to reflected sound waves mixing with the direct sound. High reverberation interferes with speech intelligibility and clarity in speech processing. The amount of reverberation is generally quantified using Reverberation Time, which is the time it takes for the sound to decrease by 60 dB after the sound activity stopped. It is affected by [2] the room size, the total surface area, and the type of materials on these surfaces. Soft, rough materials absorb sound better, reducing reverberation time and making speech clearer.

  The effect of reverberation is not only due to its attenuation of sound. It also modifies temporal and spectral characteristics of speech, which causes problems with performance in a reverberant environment when using speech recognition systems or hearing aids. The understanding and adaptation to reverberation are necessary to improve speech procession.

1.2 Intrusive vs non intrusive assessments

Intrusive assessments [3] involve a direct interaction with the system by introducing a known reference signal alongside the test signal. It allows a detailed analysis of the sound quality through direct comparisons. This method calculates a distance between the clean (reference) and the contaminated (test) signals to estimate sound quality. The calculation can be objective as in the case of SNR, or perceptual as in the case of STOI. The accuracy of intrusive methods generally achieves higher correlation with subjective evaluations due to the availability of the reference signal, which provides a direct measure against distortions, noise, and other degradation. However, this approach requires access to original, undistorted signals, which are not usually available, thus limiting its application in real-world environments.

Non intrusive assessment [3] does not interact directly with the system or the signals being evaluated. Instead, it relies on the analysis of the contaminated signal alone. This method employs advanced algorithms and models, such as neural networks, to predict the quality of speech without the need of a reference signal. For instance, the ITU-T Recommendation P.563 [4] outlines a standard for non-intrusive perceptual speech quality assessment, which is designed for practical applications where direct signal comparison is not feasible. Non-intrusive methods are particularly beneficial in situations where continuous system operation is critical, such as in real-time communication and network monitoring, offering a balance between performance and practicality without the disruption of the system functioning.

1.3 Research of competing methods

Non-intrusive estimation of signal-to-noise ratio, speech intelligibility, and reverberation time are important for systems where the undistorted signal is unavailable.

Over the years, various methodologies have been explored, each with its own assumptions and computational bases.

Historical Approaches Let me focus on historical SNR estimation. Conventional methods are based on statistical models of speech and noise components. For example WADA-SNR method [5] (the Waveform Amplitude Distribution Analysis), bases its calculations on the waveform amplitude distribution analysis. It assumes that the amplitude distribution of clean speech can be approximated by a Gamma distribution, which contrasts with the Gaussian assumption about background noise.

Modern Approaches In contrast, modern approaches increasingly leverage neural networks because they can model complex non-linear relationships without explicit programming for specific noise types. For instance, convolutional neural networks (CNNs) have been utilised effectively for non-intrusive speech intelligibility prediction, showcasing their robustness across varying acoustic conditions [6].

Non-intrusive methods have been developed also for the assessment of reverberation through T60. One approach to blind T60 estimation [7] involves model-based methods that utilise statistical models of signal decay to estimate the reverberation time. These methods typically identify speech decay intervals during silent pauses and employ algorithms to estimate the decay rates, which are then used to calculate T60. The maximum likelihood (ML) estimators represent a significant advancement, balancing estimation accuracy and computational efficiency. These estimators operate by fitting a model decay to the observed speech signal.

Recurrent and Attention Networks Most recent advancements in neural approaches include recurrent (RNNs) and attention networks. In addition to convolutional networks, which often work in short-time frequency domains, the RNNs allow processing of speech in time-domain or some combination of approaches.

Recently, the STOI-Net [8] model effectively works without needing a clean speech reference. This model integrates a convolutional neural network (CNN) and bidirectional long short-term memory (BLSTM) with a multiplicative attention mechanism. It performs exceptionally well in real-world conditions where the original, undistorted speech is inaccessible, making it highly relevant for non-intrusive settings. The model effectiveness is demonstrated through its high correlation values of 97% and 83% for seen and unseen test conditions, respectively, showcasing its robustness and adaptability across different noise environments.

Moreover, attention mechanisms have changed how models handle sequence-to-sequence tasks. Self-attention allows the model to weigh the importance of different words in a sentence irrespective of their positional distance from each other, enhancing the model ability to focus on relevant parts of the input sequence when performing tasks like speech recognition. This capability could improve the prediction of temporal patterns in speech signals.

1.4 Formal description of the problem

This thesis mainly focuses on developing a reliable estimator for non-intrusive speech quality assessment. This will be done experimentally, generating various datasets to simulate noise and reverberation conditions as described in the introductory sections of this thesis. These datasets will be used to train and validate models, which will predict SNR, STOI, and T60 metrics without reference signals.

The task was formulated as a regression problem, i.e., the model attempts to estimate a precise value of the metrics. Additionally, to analyse the structure of errors, the evaluation of the models from the classification perspective was also done. It was measured, whether the model does not confuse undistorted and highly distorted cases.

2 Solution description

2.1 Preparation of data

Data were prepared artificially, applying augmentation to clean speech signals. For signal-to-noise ratio prediction, noised signals with different SNR levels were essential. The Noisy scenario was prepared by combining clean speech with noise components. Similarly, the Reverberated and Combined scenarios were prepared using artificial impulse responses and noised components. The Combined scenario included the noised and reverberated transformations.

It is also essential to highlight that, the "standard form of the signal" refers to a signal normalised to have a zero mean and unit variance.

2.1.1 Metrics calculation

SNR calculation

To calculate the SNR metric, it is important to know which part of the signal is of interest and which is unwanted.

The global SNR is calculated for the whole signal:

$$SNR = 10 \cdot log_{10} \frac{\sigma_s^2}{\sigma_v^2}, \tag{2.1}$$

where σ_s^2 is the average signal power and σ_v^2 is the average noise power, SNR is a global SNR value.

The local SNR reference values for each second of the signal were calculated according to the formula:

$$SNR_i = 10 \cdot log_{10} \frac{\sum_{k=1}^{K} s_{i,k}^2}{\sum_{k=1}^{K} v_{i,k}^2}, \tag{2.2}$$

where SNR_i is the local SNR value, K is the signal length in seconds, k is the number of sample, i is a segment index, s is the speech signal and v is the added noise signal.

STOI calculation

To assess the STOI metric, it's essential to differentiate between the intelligible part of the speech and the portions that don't contribute to understanding. Working with

the speech signal x, which is a mix of the components (in practice are unknown, and it has to be estimated somehow): s is an articulate speech and v are background elements following the equation $x = s + v$, where s represents the clear speech component, and v denotes the inarticulate or background noise. In thesis these components were used to calculate the STOI metric using the py.stoi library [9].

T60 calculation

To compute the T60 metric, which estimates the reverberation time, it was analysed how the sound persists in the environment after the source has stopped. The reverberation time, T60 is theoretically defined [2] by the time it takes for the sound to decay by 60 dB from its initial level.

Figure 2.1 shows the scheme of decay of sound.

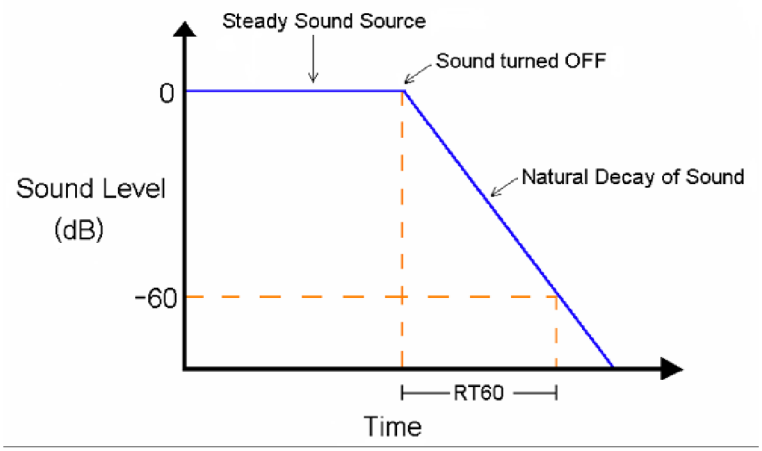

Figure 2.1: Decay of sound [10]

In 1898, Wallace C. Sabine (Sabin) developed the reverb time formula, which remains unchanged and in constant use today.

$$RT_{60} = \frac{k \cdot V}{A} = \frac{0.161 \cdot V}{A}, \tag{2.3}$$

where RT_{60} is the reverberation time, k is the factor calculated as $\frac{24 \cdot ln_{10}}{c20} = 0.161(\text{m})$, $c20$ is a speed of sound is 343 m/s, V is the room volume in m^3 and A is an equivalent absorption surface or area in m^2.

2.1.2 Noisy scenario augmentation

The augmentation of data for Noisy scenario was carried out using the algorithm described in this section. For Noisy scenario, noise with varying loudness to speech was artificially added. This simulates noisy signals with varying SNR/STOI levels. MATLAB was used for writing the programme, which created the noisy mixtures and their labels.

- **Signal normalisation**: take the original clean signal and normalise it to bring it to a standard form.
- **Noise generation**: the length of the original signal is almost always longer than the length of the noise, so several noises are concatenated to achieve equal sizes of the original signal and the combined noise. Each noise was also normalised before processing.
- **Calculating the coefficient** z: for the combination of the original clean signal and the noise mixture, a coefficient z is calculated based on the desired level of global SNR that will be used to amplify or suppress the noise.

  Since the dataset will be created with the addition of background noises, the following formula will be used to calculate the noise gain coefficient according to the desired level of global SNR after combining the clean and noise signals.

$$z = 10^{\frac{-S}{20}} \cdot \sqrt{\frac{\sum_{k=1}^{K} s_k^2}{\sum_{k=1}^{K} v_k^2}}, \tag{2.4}$$

  where z is the resulting noise multiplier, S is the desired SNR level of the resulting signal, K is the signal length, s is the signal with speech, and v is the signal with noise to be added.
- **Modify and store signals**: each noise component is multiplied by the appropriate coefficient and added to the original speech component. Separately created noise combinations are also saved.
- **Calculation of local SNR values**: local SNR values are calculated for each second of the signal mixture using the formula for local SNR metric calculation from the (2.2) equation.
- **Calculation of local STOI values**: local STOI values for each second of the signal are calculated separately. The script in the Python programming language using the pystoi.stoi [9] library was written for this process.

2.1.3 Reverberated scenario augmentation

The augmentation of data for Reverberated scenario was done using the algorithm described in this section. The reverberation is simulated through convolution of the speech component and the room impulse response (RIR).

- **Room Impulse Response (RIR)** is a recording of the acoustic characteristics of a space that captures how sound reflects off surfaces and objects within the room. While it is possible to use measured impulse responses that capture the acoustic properties of actual rooms, impulse responses for this study are generated artificially using RIR-Generator [11].

  This method generates an impulse response based on the following parameters: sound velocity in ms, sample frequency, receiver position $[x, y, z]$ in m, source

position $[x, y, z]$ in m, room dimensions $[x, y, z]$ in m. Figure 2.2 shows a scheme of the simulated room from the top.

The source position was calculated based on the required distance of the source from the receiver and the angle at which it would be positioned. Also the required level of the T60 metric was used as input to RIR generator function.

In determining the location of the sound source, its x and y coordinates were calculated. The sound source is assumed to lie on a circle with a certain distance, d, from a reference point and at an angle, α, relative to the room axis. The coordinates are then found using basic trigonometric relations: for the x coordinate, it is used $x = d\cos(\alpha)$ and for the y coordinate, $y = d\sin(\alpha)$. This approach translates the position of the sound source into a more straightforward geometric problem, leveraging the circle radius for distance and the angle for directional orientation.

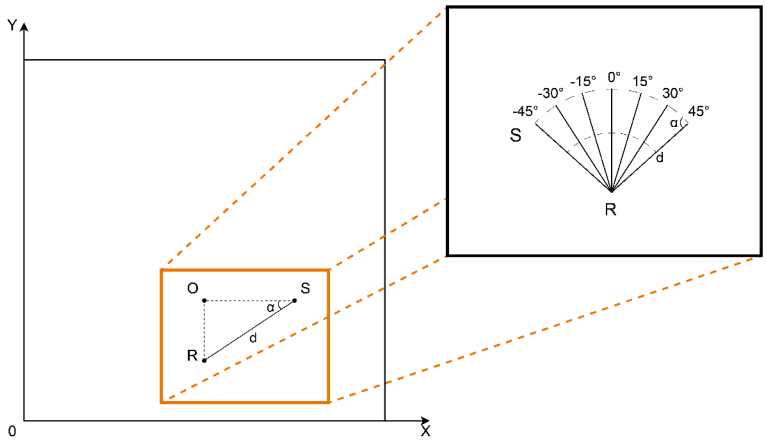

Figure 2.2: Layout of the virtual room used for artificial impulse response generation

- **Signal normalisation**: Take the original clean signal and normalise it to bring it to a standard form.
- **Signal filtration**: The convolution of the clean speech signal with the RIR simulates the reverberation. A script was written in MATLAB programming language to filter the clean signal by impulse response. The *filter* function combines the clean speech signal with the impulse response to model the time for sounds to bounce around the room and create echoes.
- **Calculation of T60 values**: T60 metric did not require additional calculation. The reference values are the same in every second of the signal and are equal to T60 values used when generating the RIRs.
- **Calculation of local STOI values**: Local STOI values for each second of the signal were calculated separately. A script in the Python programming language using the pystoi.stoi [9] library was written for this process.

2.1.4 Combined scenario augmentation

The augmentation of data for Combined scenario was done by applying noise to the previously reverberated signal. This process effectively simulates environments with both noise and reverberation, making the data realistic for testing speech processing algorithms, where both noise and reverberation are present. The methodology for augmenting builds on the techniques detailed in the Noisy scenario augmentation and the Reverberated scenario augmentation.

- **Reverberation addition**: Reverberation was applied using augmentation of the the clean signal from Section 2.1.3.
- **Noise addition**: Each reverberated signal is then noised using augmentation of noisy scenario (see Section 2.1.2). This involves: noise generation, coefficient calculation and signal combination.
- **Clone T60 Metrics**: Reverberation is unaffected by addition of the noise, thus the references are duplicated.
- **Calculate Local SNR**: The local SNR values for each second of the noise-added reverberated signal mixture are calculated using the equation (2.2) for local SNR calculation.
- **New STOI Calculation**: Values are computed to reflect the intelligibility of the combined noise and reverberation signal. The reference values of STOI are computed using the original speech component (without reverberation) and the mixture containing both noise and reverberation.

2.1.5 Spectrogram

Sound is often analysed in the short-term frequency domain, thus using a spectrogram. A Mel spectrogram was used for reduction of dimensionality, which essentially has the same format as an image.

Mel spectrogram calculation includes next steps:

- Short-time Fourier transform (STFT): the audio signal is decomposed into its frequency components, represented as a standard spectrogram
- Frequencies are converted to the MEL scale according to the formula

$$m = 2595 \cdot log_{10}(1 + \frac{f}{700}), \tag{2.5}$$

  where f is a frequency and m is a calculated MEL value.

  The MEL scale assigns a higher value to low frequencies and a lower value to high frequencies. It attempts to approximate human hearing.
- After converting frequencies to Mels, the spectrogram changes - now low frequencies take up more space and high frequencies are compressed

- In the final version of the transformed spectrogram, different frequencies are represented by different colours, and the intensity of the colour reflects the amplitude at the corresponding frequency

Spectrograms were generated with the following parameters: the size of fast Fourier transform (FFT) was 512, the window size was 512, the length of hop between STFT windows was 256, the number of Mel filterbanks was 64, and the sample rate of the audio signal was 16 kHz. Due to the unique way signals are transformed into spectrograms, one second was calculated as 62 frames represented in the resulting spectrogram. The following formula was used to calculate how many samples one second of the signal contains in the spectrogram.

$$d = n_m \cdot h - n_f, \tag{2.6}$$

where d is second length, n_m is number of Mel filterbanks, h is length of hop between STFT windows and n_f is size of FFT.

The reason of consideration less frames as one second was necessary because of movement during the creation of spectrogram. If one second would corresponded to 16000 frames, then with each movement of the spectrogram, the references would move 256 frames away from their real positions, leading to poor network training and performance.

2.2 Network architecture

The choice of neural network architecture was based on several points: type of data and type of task. The input data are speech signals transformed into Mel spectrograms. The goal is to predict the maximum possible accurate SNR, STOI and T60 values, respectively. This is a regression task.

Figure 2.3 shows the proposed architecture for solving the metrics prediction problem. The network design follows a hierarchical convolutional structure. The hyperparameters build on findings from a previous project [12], optimising the network to have enough parameters to learn effectively without overtraining.

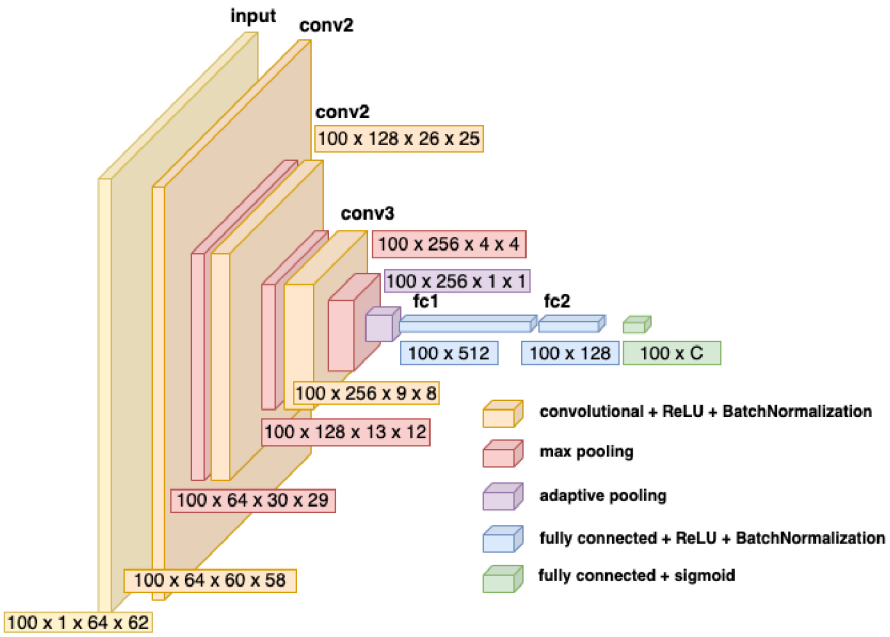

Figure 2.3: The convolutional neural network architecture [13] where C is the count of outputs according to a specific task.

The network accepts input data of shape $100 \times 1 \times 64 \times 62$, where 100 represents the batch size, and 64×62 corresponds to the dimensions of the spectrogram frames. The initial convolutional layer comprises 64 filters, which is also the batch size, followed by batch normalisation. Subsequent convolutional layers expand the feature space to 128 and then 256 filters, each time applying batch normalisation and max pooling to condense the feature maps. Global average pooling is implemented post the final convolutional stage to prepare the data for the fully connected layers.

2.2.1 Loss Function

The mean squared error (MSE), which is the mean square of the difference between actual and predicted values, was chosen to assess the losses qualitatively. This indicator shows the deviation from the desired value and indicates the model current state. MSE is not difficult to calculate, so frequent use does not slow down training.

$$MSE = \frac{1}{N} \cdot \sum_{i=1}^{N} (Y_i - \widehat{Y}_i)^2, \tag{2.7}$$

where Y_i is the ith observed value, $\widehat{Y}_i$ is the corresponding predicted value and N is the number of observations.

2.2.2 Scaling of output values

A convolutional network was chosen based on the matrix form of the spectrogram input. Since the MSE loss function is sensitive to the difference in the range of

individual inputs, it was necessary to normalise them. Without normalisation, only the metric with the largest range would be trained; in this case, it would be T60. In preprocessing phase, all the input labels were scaled to fall within the range of $\langle 0, 1 \rangle$. SNR labels, which originally placed in an infinite range, were mapped from $\langle -\infty, \infty \rangle$ to a range of $\langle 0, 1 \rangle$ before sending it into the network. Similarly, T60 labels were scaled down by a factor of 0.001 to ensure consistency in the range. In postrocessing phase, the outputs were converted back to their original scales. SNR outputs were inversely mapped to the original infinite interval and T60 outputs were multiplied by 1,000. Unlike these metrics, the STOI metric, which is already calculated within the required interval, was not preprocessed.

2.2.3 Optimiser and Scheduler

The Adam optimiser, with an initial learning rate of 0.0001, is employed for adjusting the network weights. To further refine the training, a learning rate scheduler is utilised, which reduces the learning rate when the validation loss plateaus, ensuring steady and controlled updates to the model parameters.

The schedulers for relevant models were configured with reduction factors of 0.95 and 0.1, patiences of 3 and 12 epochs, and a threshold for measuring the new optimum of 0.0015. The schedulers are responsible for decreasing the learning rate when the validation loss plateaus. Due to memory constraints, the entire dataset could not be loaded into memory at once; instead, dividing it into five parts was necessary, with only one-fifth of the dataset being loaded at a time. Consequently, what constitutes one training epoch in setup is equivalent to five loadings of these dataset partitions. This approach differs with smaller datasets, where one epoch loads the entire dataset simultaneously. This strategy was crucial to accommodate computational resources.

2.2.4 Practical Usage

The computational experiments were conducted using remote GPUs provided by Google Colab. The Google Colab Pro version was purchased to access enhanced computational resources. Specifically, the models were trained on Tesla T4 GPUs equipped with high-RAM settings, which included 51.0 GB of system RAM, 15.0 GB of GPU RAM, and 201.2 GB of disk space.

Our datasets were primarily hosted on Kaggle and Google Drive. Before processing, the datasets were cloned into the Colab environment to ensure efficient data handling and accessibility. This setup facilitated larger datasets that require significant memory and processing power.

Regarding software, the project leveraged PyTorch for model development and torchaudio for audio processing tasks. The essential Python libraries utilised in the used scripts included:

- Visualisation: `matplotlib.pyplot`

- Google Colab specific functions: `files`, `drive`. These libraries were essential for datasets importing to Google Colab
- PyTorch specific utilities and classes: `torch` was the main library, which helped working with tensors, `Dataset` helped write an UploadArtificialDataset dataset processing class. This class handles the initial data ingestion, prepares it for model training, and facilitates retrieval of specific instances of distorted signals for analysis. `DataLoader` was used for creation of UploadArtificial-DataModule, which was responsible for loading parts of dataset
- PyTorch Lightning modules and utilities: `pytorch_lightning` was used for writing LitNet class, which describes behaviour of the model, `TensorBoardLogger`, `ModelCheckpoint` and `SimpleProfiler` were essential for saving metrics while training, validation and testing the models

3 Experiments

3.1 Datasets preparation

To train and evaluate the models, datasets had to be prepared. It was required to test the prediction of metrics in the presence of noise, reverberation and their combination. A corresponding dataset was created to evaluate each ability. In addition, when evaluating the quality of the noise model, language dependency was also evaluated. This added the creation of two additional noise datasets with different languages from the main one.

Three datasets were created to train and evaluate prediction in noisy conditions: English (Noisy-En), German (Noisy-Ge) and Korean (Noisy-Ko). The training, validation and test parts were in the English and German datasets, and the Korean one was used only for testing. The clean datasets were divided into training, validation, and test parts in the ratio 88:6:6(%). The sizes of the datasets were selected to be approximately the same for all the language variants to make the results of the experiments comparable. It is important to note that when creating the Korean dataset, finding high-quality, clean data was difficult, and this was the reason for the smaller number of tested seconds in the dataset.

One dataset was created to evaluate estimation in noiseless reverberant conditions (Reveb-En). See section 3.1.2 for detailed description.

One dataset was created to evaluate the estimation of SNR, STOI and T60 in noisy reverberated conditions. This dataset will be called a Combination English dataset; see section 3.1.3 for details.

The 3.1 table shows the size of each part of the datasets in seconds.

Table 3.1: Datasets: duration in seconds

Dataset	**Train**	**Validation**	**Test**
Noisy English	1,845,414	127,902	130,812
Noisy German	1,829,796	230,088	126,642
Noisy Korean	-	-	77,580
Reverberated English	307,569	21,317	21,802
Combined English	1,845,414	127,902	130,812

3.1.1 Noisy datasets

The clean speech augmentation with noise is described in section 2.1.2. The clean speech data originates from several databases that differ in language. To each clean file a noise was added from the DCASE [14] database from the public square and street subparts.

The noisy English dataset (Noisy-En) utilises clean speech data from the Librispeech [15] dataset. It contains 100 hours of recordings uttered by various speakers. Book excerpts are read by different people - the data is independent of the speaker's gender and age. The textual information is non-repeating. It is officially confirmed that there is almost no noise in the data, and it can be considered as clean speech.

When creating the Noisy-En dataset, the size of the original, clean dataset was artificially increased. The original LibreSpeech dataset was replicated six times, where each replica corresponded to a different level of global SNR. The main idea is for the model to see the same utterances with different noise levels, which should simplify SNR estimation. Clean speech and noise were combined according to six levels of global SNR: -5 dB, 0 dB, 5 dB, 10 dB, 20 dB and infinite (expressed as 50 dB due to technical reasons).

The chosen SNR levels cover many scenarios, from very noisy to noise-free environments. By selecting a balanced range of SNR levels, there is an insurance that the neural network is nether under-tested in overly simple scenarios nor over-tested in excessively harsh conditions. The attention was paid to challenging noisy conditions, where SNR is -5 - 0 dB, moderate conditions with 5 and 10 dB SNR, high clarity with 20dB and ideal conditions without the addition of any noise.

Several experiments were performed to test the neural network functionality to new data. The main and most representative test was a test on data prepared in the same way and using the same language as the main training part - the Noisy English test dataset. Then, two secondary experiments were prepared on datasets created similarly but based on different languages. There is a wish to test the language dependence of the model. Since the initial training dataset was created with English only, it was interesting how the model performs when processing an unseen language. The mail point was the division of the world languages into related groups. The structure and logic of languages from the same group are close, which means that the first test should be performed on a language belonging to the same group as English: so choosing narrowed down the search group to 6 variants. It was decided to test on the German dataset.

For the next experiment, a language from a very distant language group had to be choosen, as well as one that was unique in its structure, speech nature and patterns. According to these criteria, it was chosen from isolated languages, which are languages that have been scientifically proven not to belong to any known language family and thus form their own. Such languages include Sumerian, Korean and Hadza. Korean was chosen because of its familiarity.

When creating test datasets for language dependency tests, German and Korean open datasets were chosen. Although their quality is not as high as that of Librispeech, they could also be considered clean, which means that the tests will not

be significantly distorted.

To create `Noisy German dataset` (Noisy-Ge), several clean speech datasets were used from the HUI Audio Corpus German [16]. This dataset presents a smaller variety of speakers. In contrast to the initial English dataset, there are only five speakers. The complete clean dataset was compiled as a combination of the clean sets of each speaker. The benefit of this dataset is that the speakers talk about various topics. The final version of the prepared dataset contains training, validation and test parts. The dataset was used to train the German language-dependent model in section 3.4.2.

The Common Voice Korean [17] was used to create the `Noisy Korean test set` (Noisy-Ko). This source cannot guarantee the clarity of the data, as anyone can upload their audio there. Common Voice validates audio recordings before publishing but is not responsible for their clarity. Only a test part of the dataset was created because this dataset is only used for testing noisy models.

All tests were performed with previously unused noise instances, but the noises originated from the same environment as in the noisy training set.

3.1.2 Reverberated dataset

The reverberated dataset (Reverb-En) was created based on the undistorted Librispeech [15] dataset. The detailed algorithm for creating the training, validation and test part is presented in the section 2.1.3.

The size of the original clean dataset was not extended when creating Reverb-En. Different impulse responses were generated for the training/test part of the dataset. To generate the impulse responses, an artificial RIR generator ([11]) was used. The free parameters of the RIR generator were set to the following values: sound velocity was 340 m/s, the frequency was chosen the same as in the original clean files 16000 Hz, receiver position was 3x1x1.25 m, and the simulated room size was 6x6x2.5 m. Each reverberated file featured a different sound source position in the virtual room. The source-microphone distance ranged within the interval 0.5-2.5 m with increments of 0.25 m (i.e., 9 distances). There were 7 angular positions available, ranging from -45° to 45° degrees with 15° increment. The possible positions are depicted in Figure 2.2. Test-generated impulse responses were generated with slightly different conditions: receiver position was 2.5x2.5x1.25 m, the simulated room size was 5x5x3 m, and distances were from 1 m to 3 m with increments of 0.25 metres.

The impulse response for each resulting source position was generated based on nine selected T60 levels (from 160 ms to 800 ms with increments of 80 ms). Considering all combinations of distance, angle and T60, 567 impulse responses were generated. The clean speech dataset was thus divided into 568 parts, where each part was either augmented with one of the RIRs or kept unmodified.

The Reverb-En dataset was used as the basis for the Combi-En dataset to match the size of the Noisy-En dataset.

3.1.3 Combined dataset

The utterances of the combined dataset are distorted by both the reverberation and the background noise. The process of creating this dataset was divided into several stages. To include the reverberation, the noiseless data are taken from the Reverb-En (3.1.2) dataset instead of the original Librispeech. From Reverb-En, the reference values of the T60 metric were used. Then the size of the Combi-En dataset was artificially increased similarly to the creation of the Noisy-En (3.1.1) dataset. The original Reverb-En dataset was replicated six times, where each replica corresponded to a different level of global SNR (-5 dB up to infinite). The reverberated utterances were used to compute the SNR labels. However, the STOI labels were created using the undistorted utterances from the Librispeech [15]. In this way, T60 labels express only reverberation, SNR represents the noise level over reverberation, and the STOI metric indicates the overall level of speech intelligibility after reverberation and noise addition.

This dataset is used to train and evaluate the most complex model, which allows estimation of all three metrics.

3.2 Transformation into classification problem

When evaluating a regression problem, the key metric is RMSE. A regression problem involves predicting a continuous variable. It is important that the model gives an answer as close as possible to the expected value. The values that the model predicts can take any value on a numerical scale.

In turn, the classification task is to predict the category of a variable. The output is a class label or a probability of belonging to a particular class. It was important to look at the problem from a classification point of view, as this helps to analyse the distribution of errors in the predictions of the models. Small errors are acceptable, and the model should never confuse very high-quality signals with very low-quality ones. To achieve this, regression task was converted into a classification task in the way described in Section 3.2.1. Subsequently, the confusion matrix was used to reveal the error distribution.

The SNR and T60 within the classification problem was assessed using a confusion matrix.

3.2.1 Definition of classes

Six categories corresponding to the SNR intervals used during training were defined, as detailed in the 3.1.1 subsection. These intervals are: $(-\infty, -5\rangle$, $(-5, 0\rangle$, $(0, 5\rangle$, $(5, 10\rangle$, $(10, 20\rangle$, and $(20, \infty)$. Concerning the T60 metric, six categories based on T60 levels outlined in the 3.1.2 subsection were also defined. These categories are delineated by the following intervals: $\langle 0, 160)$, $\langle 160, 288)$, $\langle 288, 416)$, $\langle 416, 544)$, $\langle 544, 672)$, $\langle 672, 800)$.

Additionally, a safety margin was defined, which allows to count as a classification hit estimates that lie at the boundaries of the defined intervals/classes and exhibit

an acceptable error. The margin for the SNR was set at half the smallest interval - 2.5dB and the margin for the T60 metric was set to 60 ms. For example, let me consider a sample with a true SNR belonging to interval $\langle 0, 5\rangle$dB. If the network estimates that this sample corresponds to any SNR in the interval $\langle -2.5, 7.5\rangle$ dB, this is considered as a hit.

3.2.2 Calculation of useful intervals

Boundaries were established for the predicted metrics by defining useful intervals. This approach stems from an understanding that errors within extreme ranges (i.e., those approaching +/- infinity) can be disproportionately large, thereby skewing the overall accuracy of the model estimates. Specifically, in the calculation of the Root Mean Square Error (RMSE), deviations from the mean metric value are weighted more heavily, which could lead to an underestimation of the model precision (see Section 3.3). To mitigate this, there is a focus on a narrower range—the useful intervals—where deviations are more meaningful and indicate the model performance. By evaluating the metrics within these defined intervals and the full range of values, there is a gain a more nuanced understanding of the model quality.

The choice of a useful interval for SNR was based on the fact that signals with SNR higher than 20 dB can be considered undistorted, and for SNR lower than -5 dB the signal is completely unintelligible. In this regard, errors in the intervals $(-\infty, -5\rangle$ and $(20, \infty)$ are assumed less relevant with respect to the performance of the model.

Useful interval for STOI is based on observation that about 3% of lowest STOI values correspond to highly noisy speech or noise only intervals of recordings.

Non-speech intervals when evaluating reverberation

The T60 metric is estimated based on the amount of reverberation present in the speech signal. It cannot be consistently estimated in non-speech intervals. The evaluation of T60 metric thus can be biased, when the test signal contains a long period of silence. To analyse the influence of such errors on T60 estimation, the outputs of the model were also evaluated when limited to speech-only intervals. It was necessary to exclude seconds with insufficient quality from the estimation. The decision about the quality of a second was based on the STOI metric: a second outside the useful interval of this metric is considered to be of non-speech. If a second is considered to be non-speech, it is not included in the T60 interval. The non-speech intervals were determined in two ways. An ideal way when the true STOI labels were used to determine non-speech. Or the practical way, when STOI estimated by model was used to this end. In other words, the trained model detects non-speech intervals with no guarantee of correctness.

3.3 Evaluation metrics

To evaluate the model, the following two metrics were used: RMSE and accuracy. When evaluating from the regression perspective, the RMSE is used. To analyse the classification point of view, the accuracy is calculated.

- RMSE calculation

  The most representative metric to present the results is Root Mean Square Error (RMSE). This metric will measure the average deviation between the estimated and the actual value of a particular metric.

$$RMSE = \sqrt{\frac{1}{K}\sum_{i=0}^{K}(\hat{u_k} - u_k)^2}, \tag{3.1}$$

  where $RMSE$ is calculated in the same units as the data used for the calculation. For example, if the data is measured in meters, it will also be in meters. $\hat{u_k}$ is a predicted value in kth second and u_k is the corresponding expected value.

- Accuracy calculation

  The accuracy of the model, denoted by a, is calculated using the following formula:

$$a = \frac{y}{y + n_i} \cdot 100, \tag{3.2}$$

  where y is the number of correct predictions, including those within an acceptable safety margin and n_i is the total number of incorrect predictions.

3.3.1 Confusion matrix for SNR and T60 evaluation

The confusion matrix in a classification task has size $C \times C$, where C is the number of classes. Each column of this matrix represents a correct class, and each row represents a predicted class. Values on the main diagonal of this matrix indicate the number of correctly classified examples for each class, while values off the main diagonal indicate classification errors. Classification errors not only indicate the existence of an error in prediction, but also show how far the predicted class is from the correct class. The confusion matrix is diagonal for a perfect classifier. Errors around the diagonal indicate confusion for close cases.

3.4 Noisy scenario

This Section evaluates precision of SNR and STOI estimates obtained on noisy speech recordings without reverberation. The noisy model will be the baseline for

other models in this thesis. It was used to select appropriate parameters and network architecture for training.

3.4.1 The Noisy-En model

While training the network, the data were evaluated on the Noisy-En validation dataset 5 times per epoch. More information about the Noisy-En dataset can be found in the section 3.1.1. The more frequent estimation of the loss function on the validation dataset is due to the large amount of training data: frequent estimation on validation data unseen by the network prevents over-training.

The graph of the loss function during model training tends to be a monotonically decreasing function, albeit with minimal deviation. This indicates a gradual training of the model.

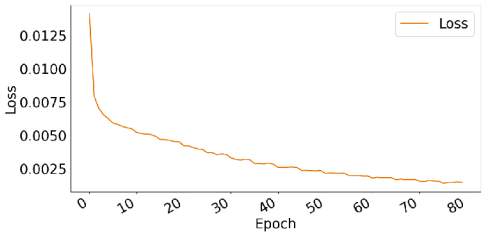

Figure 3.1: Noisy-En model: training loss function

The loss function on validation data does not run as smoothly as when it is evaluated on training data. The validation loss exhibits decreasing trend with random fluctuations.

When looking in detail at the loss functions, around epoch 37, the onset is observed of model over-training, increasing in frequency of loss oscillations and increasing in model error on the validation data.

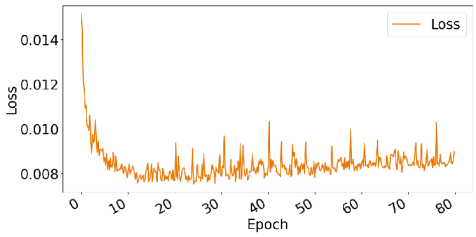

Figure 3.2: Noisy-En model: validation loss function

Despite the fact that the training took multiple epochs, the validation evaluation indicates the best model after 37 epochs.

Table 3.2: Noisy-En model: Average RMSE Values for English test dataset

	English
SNR (dB)	3.44
STOI	0.11
SNR in useful interval (dB)	1.85
STOI in useful interval	0.09

Table 3.2 presents the results of RMSE estimation for Noisy-En test dataset either for full range of data and for data corresponding to the usefull intervals. The results indicate that the RMSE of the SNR metric is lower than 2 dB when considering the useful intervals only. This suggests a reliable performance of the proposed estimator.

The following Table 3.3 presents the performance of the English model when evaluating the SNR from the classification point of view. It was wanted to measure the accuracy of the estimator as a classifier. The accuracy and RMSE in each interval were calculated.

The results showed that most of the errors are around the diagonal, so the network confuses mostly close cases. The confusion of mainly close cases indicates a well trained network.

Table 3.3: Noisy-En model on English test dataset: Confusion and classification accuracy for SNR

	$(-\infty, -5\rangle$	$(-5, 0\rangle$	$(0, 5\rangle$	$(5, 10\rangle$	$(10, 20\rangle$	$(20, \infty)$
$(-\infty, -5\rangle$	12142	381	58	10	1	0
$(-5, 0\rangle$	3143	18393	430	24	1	0
$(0, 5\rangle$	232	1009	19675	708	20	0
$(5, 10\rangle$	5	68	615	18210	1027	1
$(10, 20\rangle$	0	0	22	386	20187	1010
$(20, \infty)$	0	2	2	1	688	31249
Accuracy (%)	78.2	92.6	94.6	94.2	92.0	96.9
RMSE (dB)	7.52	1.9	1.64	1.62	2.15	3.43

This experiment represents a matched training-test scenario, where both training and test sets share the same language. The SNR prediction accuracy exceeded 90% on all evaluated intervals except $\langle -5, -\infty)$, whose accuracy reached 78.2%. This behaviour of the model can be explained by two factors:

- the smaller number of data, in contrast to the other classes,
- the complicating evaluation of highly noisy signals

It is also important to note that signals with SNR values less than -5 dB are considered highly distorted, and precise estimations of noise levels become less critical in such conditions. Excellent prediction results for the SNR metric are presented

for the $(20, \infty)$ interval. In general, high prediction performance for higher quality signals is a distinctive feature of the model. The model must accurately identify signals of significantly differing qualities to avoid extremely incorrect predictions, such as mistaking a signal with a quality of 25 dB for one at -5 dB.

3.4.2 Language dependency

Language dependency of Noisy-En model There is an eager to find out, if the Noisy-En model would be functional on datasets, its language wasn't part of training. So, the performance of a Noisy-En model was explored, when applied to datasets in languages different from the training language. While testing the Noisy-En model on Noisy-Ge and Noisy-Ko test datasets in Table 3.4, there was an observation that on related datasets, Noisy-En and Noisy-Ge, the model performed similarly. Still, it performs significantly worse on the Noisy-Ko dataset.

Table 3.4: Noisy-En model: Average RMSE Values for English, German, and Korean test datasets

	SNR (dB)	**STOI**	**SNR in useful interval (dB)**	**STOI in useful interval**
English test	3.44	0.11	1.85	0.09
German test	3.25	0.12	2.01	0.09
Korean test	4.94	0.17	2.68	0.14

In the analysis, the Noisy-En model performance on Noisy-Ge test data is documented in Table 3.5. The model achieved a classification accuracy of 78.43% for the $(-\infty, -5\rangle$ SNR category, reaching up to 96.67% in the $(20, \infty)$ range, with corresponding RMSE values from 7.61 dB to 2.81 dB across the SNR intervals. The results are similar to Noisy-En performance on the Noisy-En test dataset in Table 3.3, where accuracy for the $(-\infty, -5\rangle$ category reaches up 78.2% with 7.52 RMSE value and 96.9% with 3.43 RMSE value in the $(20, \infty)$.

Table 3.5: Noisy-En model on German test dataset: Confusion and classification accuracy for SNR

	$(-\infty, -5\rangle$	$(-5, 0\rangle$	$(0, 5\rangle$	$(5, 10\rangle$	$(10, 20\rangle$	$(20, \infty)$
$(-\infty, -5\rangle$	11840	513	76	16	5	0
$(-5, 0\rangle$	3018	18187	608	26	4	0
$(0, 5\rangle$	230	865	19063	1082	33	0
$(5, 10\rangle$	3	55	480	17524	1417	1
$(10, 20\rangle$	3	1	15	342	19081	1048
$(20, \infty)$	2	0	2	3	580	30477
Accuracy	78.43%	92.69%	94.17%	92.27%	90.35%	96.67%
RMSE (dB)	7.61	1.96	1.67	1.69	2.29	2.81

The following Table 3.6 presents the performance of the Noisy-En model when estimating SNR on the Noisy-Ko test dataset from the classification point of view. The model predictions on the Noisy-Ko test dataset were highly effective with the highest accuracy of 93.67% in the $(20, \infty)$ SNR interval and RMSE of 3.56 dB, while the lowest accuracy reached out 82.63% with a higher RMSE value 8.12 dB in the interval $(-\infty, -5\rangle$. Almost for each category the accuracy decreased on 5-8% against Noisy-En testing in Table 3.3, but improvement of accuracy for interval $(-\infty, -5\rangle$ was detected. It increased from 78.2% on Noisy-En to 82.63% on Noisy-Ko dataset. The overall degradation in the quality of the predictions was caused by significant language difference between english and korean. Testing of other mismatched conditions, such as different noise types, is beyond the scope of the thesis.

Table 3.6: Noisy-En model on Korean test dataset: Confusion and classification accuracy for SNR

	$(-\infty, -5\rangle$	$(-5, 0\rangle$	$(0, 5\rangle$	$(5, 10\rangle$	$(10, 20\rangle$	$(20, \infty)$
$(-\infty, -5\rangle$	9579	746	143	34	8	61
$(-5, 0\rangle$	1834	10127	744	97	11	4
$(0, 5\rangle$	173	690	10569	1061	108	5
$(5, 10\rangle$	5	48	408	9458	1179	17
$(10, 20\rangle$	0	5	26	284	10610	1126
$(20, \infty)$	1	3	3	13	383	17937
Accuracy (%)	82.63	87.16	88.87	86.40	86.27	93.67
RMSE (dB)	8.12	2.18	1.85	1.91	2.44	3.56

Noisy-Ge model training motivation Given the results of Noisy-En language dependency testing, there is an aim to verify if a model trained specifically for another language, in this case German, termed Noisy-Ge, would show improved performance on German datasets. It was compared to Noisy-En model performance on English

datasets. This step was crucial to determine if targeted training could mitigate the effects of language variation.

Noisy-Ge model training When observing the loss function on the training dataset (3.3), a decreasing trend of the function is visible. This reflects that the training process is proceeding correctly.

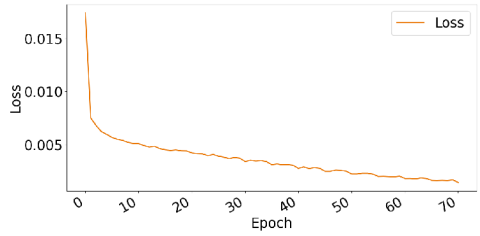

Figure 3.3: Noisy-Ge model: training loss function

There are small oscillations in the validation loss function, but there is a decreasing trend over-training manifests. This indicates that training is successful and the model can see similar patterns in the training and validation data.

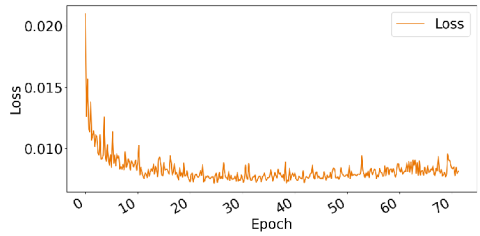

Figure 3.4: Noisy-Ge model: validation loss function

Noisy-Ge model experiments The Table (3.7) shows RMSE on the full sets of tested data as well as on useful intervals during the testing of both: Noisy-En and Noisy-Ge models.

There is a tendency for lower quality predictions on the tested datasets with languages that were not part of the training. Predictions for the datasets with close languages came out better than those with distant languages. So it could be said that language has some effect on the accuracy of the results, but the estimation also works with unseen languages.

Table 3.7: Noisy-En and Noisy-Ge models: Average RMSE Values for English, German, and Korean test datasets

	SNR (dB)	STOI	SNR in useful interval (dB)	STOI in useful interval
English test				
Noisy-Ge	4.00	0.12	2.45	0.10
Noisy-En	3.44	0.11	1.85	0.09
German test				
Noisy-Ge	2.97	0.10	1.97	0.08
Noisy-En	3.25	0.12	2.01	0.09
Korean test				
Noisy-Ge	5.03	0.17	2.89	0.15
Noisy-En	4.94	0.17	2.68	0.14

In the useful interval, the best results are achieved on Noisy-En test dataset, followed by a closely related language, and the poorest on a linguistically distant language. This progression aligns with the expectations based on linguistic similarities.

3.4.3 Case study: analysis of a single sentence

In order to understand and present the quality of the model predictions, this section discusses specific example of prediction when analysing a single test utterance. Along with the analysis, the creation of the test sample, as described in section 2.1, is demonstrated as well. The output will be discussed for the signal with filename "8465-246943-0010.flac". This sample signal lasts 14 seconds and has a sampling frequency of 16kHz.

Creation of an artificial mixture The first step was to load and normalise the signal. For precise evaluation of the experiments, it was necessary to create combinations of clean speech and noise in the same way as in the training part of dataset. Our network is designed to process 62 frames of a spectrogram as input i.e., aproximately 1s of the signal. To align the input signal with this requirement, it was trimmed to have a sample count that is an exact multiple of this frame number. For the signal in question, this results in 14 seconds.

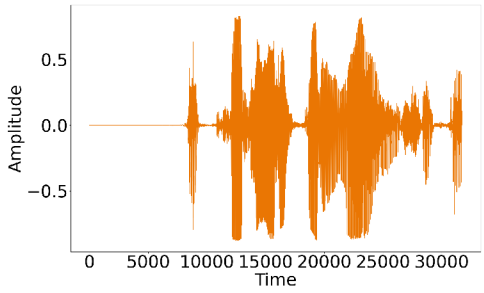

Figure 3.5: Clean normalised signal

In the next step, a noise component of the mixture is selected from the dataset of available noises. Should the noise component be shorter than the speech, the noise component is created by concatenation of several normalised noises.

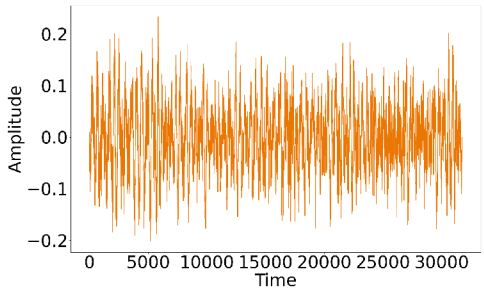

Figure 3.6: Normalised noise component of the same length as the speech

Then the coefficient z was calculated same way as it is described in section 2.1.2 according to the global SNR equal to 10 dB. The noise was amplified by this coefficient and summed with the speech signal.

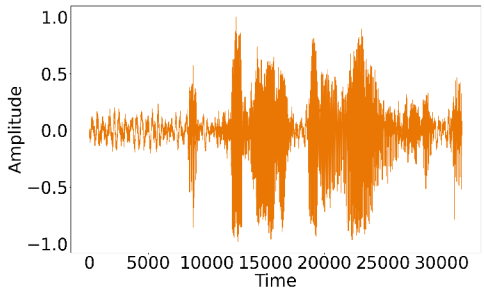

Figure 3.7: Combination of clean signal and noise

Using the known components of the noisy mixture, the reference values of SNR and STOI were computed. These serve as ground truth to compare with the output of the network.

Processing of the mixture The time-domain mixture is subsequently transformed into the MEL spectrogram before sending it to the model for evaluation.

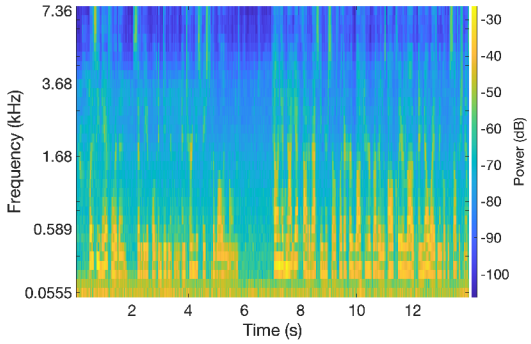

Figure 3.8: Noisy mixture transformed into a MEL spectrogram

The graphs bellow show that the network is able to accurately estimate the metrics during the course of the whole signal. It is also important to note that the graphs indicate that the model is able to compute the non-speech segment of the signal successfully. Notably, the network capability to identify non-speech segments is evidenced by its return of significantly negative SNR values and STOI figures approaching zero. The reference global SNR is 10 dB. Speech is a non-stationary signal, meaning it changes its energy/variance over time; hence, the actual SNR oscillates around 10 dB throughout the signal's duration.

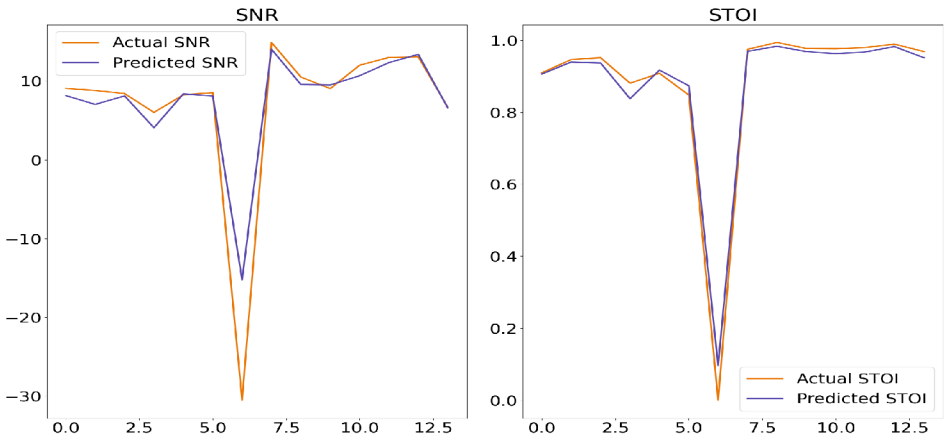

Figure 3.9: Prediction graphs

Tables 3.8 and 3.9 provide illustrative examples of the predictions of the SNR and STOI metrics for the tested audio file. Metrics values were rounded to integers in the case of SNR estimation and to 1 decimal point in the case of STOI estimation.

Table 3.8: Comparison of expected and predicted SNRs

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**
Expected (dB)	9	9	8	6	8	9	-31	15	10	9	12	13	13	7
Predicted (dB)	8	7	8	4	8	8	-15	14	10	10	11	12	13	7

Table 3.9: Comparison of expected and predicted STOIs

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**
Expected	0.9	0.9	0.9	0.9	0.9	0.8	0.0	1	1	1	1	1	1.0	1
Predicted	0.9	0.9	0.9	0.8	0.9	0.9	0.1	1	1	1	1	1	0.9	1

3.5 Noiseless scenario with reverberation

This part of the work presents results in predicting T60 a STOI metrics using noiseless reverberated utterances. The model Reverb-En is trained; training and validation utilize the Reverb-En dataset from Section 3.1.2. For estimation of T60 in noisy conditions see the following Section 3.6.

The loss function on the training dataset is decreasing, indicating that the model is capable of learning from the training data.

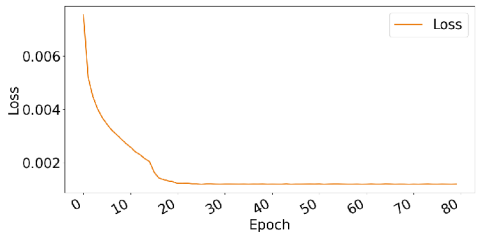

Figure 3.10: Loss function while training on reverberated dataset

The loss function on the validation data tends to oscillate, although it has a decreasing trend. The application of a learning rate scheduler has introduced a significant modification to the validation loss curve, resulting in attenuation of the oscillations.

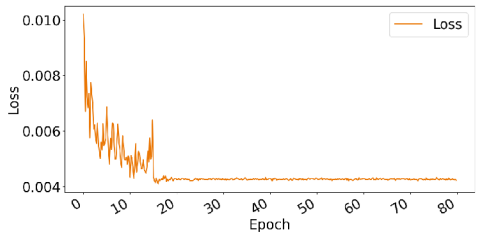

Figure 3.11: Loss function while validation on reverberated dataset

The evaluation of the Reverb-En model via RMSE of the T60 a STOI estimates is presented in Table 3.10. Process of speech/non-speech intervals selection is described in the 3.2.2 section.

Table 3.10: Reverb-En model: Average RMSE Values for test Reverberated dataset

RMSE parameter	**Value**
STOI	0.10
T60 (ms)	61.92
T60 in speech intervals (Ideal) (ms)	56.80
T60 in speech intervals (Practical) (ms)	58.13

A slight improvement was observed in the quality of predictions at speech intervals in the table 3.10. The values at speech intervals of the T60 metric can be considered as equal, due to a small difference of 1.33 ms. This indirectly indicates that the model can also serve as a speech activity detector.

The model on the uniformity of RMSE values was also evaluated. The Table 3.11 represents confusion matrix of the classifier. The evaluation was performed by transforming the regression problem into a classification problem as described in the 3.2 section.

The calculations take into account a safety margin of 40 ms as discussed previously.

Table 3.11: Reverb-En model: Confusion and classification accuracy for T60

	$\langle 0, 160)$	$\langle 160, 288)$	$\langle 288, 416)$	$\langle 416, 544)$	$\langle 544, 672)$	$\langle 672, 800)$
$\langle 0, 160)$	1613	222	27	1	2	1
$\langle 160, 288)$	15	4372	68	1	8	9
$\langle 288, 416)$	9	78	4701	637	45	17
$\langle 416, 544)$	6	11	42	1798	1373	38
$\langle 544, 672)$	0	4	10	42	3290	834
$\langle 672, 800)$	0	2	5	4	90	2425
Accuracy (%)	98.17	93.24	96.87	72.41	68.43	72.95
RMSE (ms)	49.2	40.7	43.5	56.6	72.3	94.2

High accuracy in intervals with lower reverberations could be seen. Interval with smallest reverberations $\langle 0, 160)$ ms accuracy achieved 98.17%. High accuracy persists in low reverberated intervals but gradually decreases through the mid-range intervals. Correspondingly, the RMSE increases as the reverberation level. The RMSE is lowest at 40.7 ms for the low reverberated interval $\langle 160, 288)$ and escalates to 94.2 ms for the high reverberated interval ($\langle 672, 800)$ ms). The non-diagonal elements of the table show that there is considerable confusion between adjacent intervals. This confusion increases with higher reverberated intervals. The intervals from $\langle 416, 544)$ ms onward exhibit not only lower accuracy but also the highest rates

of misclassification. The model tends to underestimate reverberation time by lowering predictions to lower neighbouring intervals. Despite all challenges, the model performs effectively, as even an RMSE of 90 ms is considered good. The confusion matrix remains primarily block diagonal, indicating minimal misclassifications between very high and very low T60 intervals.

3.6 Combination scenario

This chapter focuses on the final model within the scope of the thesis. The model trained on the Combi-En dataset (3.1.3) predicts all three metrics: SNR, STOI, and reverberation.

The loss function on the training data has a decreasing trend, similar to the loss functions on Noisy models from Sections (3.4) and Reverberation models (3.5). This demonstrates the ability of the model to learn despite the increasing complexity of the task.

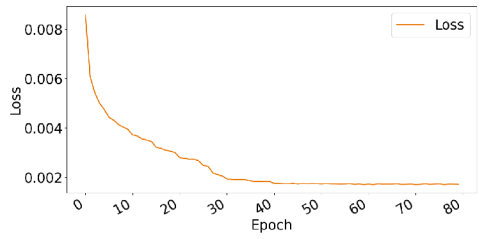

Figure 3.12: Loss function while training on combinated dataset

When analysing the validation loss function from Figure 3.13, a general decreasing trend with a slight oscillation is observed.

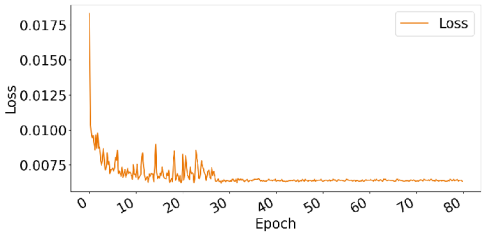

Figure 3.13: Loss function while validation on combined dataset

The quality of the model was assesed using the test part of the Combi-En dataset. For each predicted metric, RMSE was calculated both on the full dataset and on useful intervals.

Table 3.12 provides the average RMSEs for various parameters measured both in the overall test and in useful intervals.

The SNR and STOI metrics prediction did not show significant changes between the Noisy-En and Reverb-En models in the comparative analysis. Specifically, RMSE for SNR was 3.44 dB in the Noisy-En model and slightly improved to 2.97 dB in the Combi-En model. This could be explained by adjustments of scheduler parameters while training the Combi-En model, which are: lower factor 0.1 and more extended patience 12. Similarly, STOI remains consistent at approximately 0.11 across all trained models.

However, a notable observation was the high decrease in the quality of T60 predictions in the Combi-En model 86.15 ms compared to the Reverb-En model 61.92 ms. This suggests that accurately predicting reverberation levels, represented by the T60 metric, could be a more significant challenge. This increased difficulty primarily stems from reverberation masked by noise, which skews the reverberation estimates.

This difficulty in T60 estimation might be attributed to more than just the complexity of the problem. Still, it could also be due to the smaller dataset size available for T60 measurements at each noise level.

Table 3.12: Models: average RMSE Values in dB for test datasets

	Value
Combi-En	
SNR (dB)	2.97
STOI	0.11
T60 (ms)	86.15
SNR in useful interval (dB)	1.97
STOI in useful interval	0.10
T60 in speech intervals (Ideal) (ms)	75.50
T60 in speech intervals (Practical) (ms)	79.46
Noisy-En	
SNR (dB)	3.44
STOI	0.11
SNR in useful interval (dB)	1.85
STOI in useful interval	0.09
Reverb-En	
STOI	0.10
T60 (ms)	61.92
T60 in speech intervals (Ideal) (ms)	56.80
T60 in speech intervals (Practical) (ms)	58.13

From the comparison of the tables with the prediction results of the SNR metric on the Noisy-En test dataset (3.3) and on the Combi-En test dataset (3.13), there is no significant change in the quality of the predictions. The highest accuracy on Combi-En model is observed in the highest SNR interval $(20, \infty)$ with 96.64%. The accuracy in the lowest interval is also high, 81.13%, indicating robust performance in extremely noisy conditions. Accuracy tends to increase as SNR increases. RMSE values vary across SNR intervals, with the lowest error at 1.7 dB for the $\langle 5, 10\rangle$

interval and the highest at 6.6 dB for the $(-\infty, -5\rangle$ interval, reflecting greater prediction difficulty at lower SNRs. Despite the increased complexity of the problem, the table still has a diagonal character. The accuracy of the selected SNR intervals has slightly decreased, manifested in the increase of substitutions to neighbouring intervals.

Table 3.13: Combi-En model: Confusion and classification accuracy for SNR

	$(-\infty, -5\rangle$	$(-5, 0\rangle$	$(0, 5\rangle$	$(5, 10\rangle$	$(10, 20\rangle$	$(20, \infty)$
$(-\infty, -5\rangle$	13058	474	34	4	1	0
$(-5, 0\rangle$	2691	17899	448	15	0	0
$(0, 5\rangle$	333	1649	19309	697	21	0
$(5, 10\rangle$	13	110	1121	17415	918	2
$(10, 20\rangle$	0	2	55	810	20696	1080
$(20, \infty)$	0	0	0	1	830	31114
Accuracy (%)	83.13	88.90	92.09	91.94	92.12	96.64
RMSE (dB)	6.6	2.1	1.8	1.7	2.2	2.0

Table 3.14 presents the estimation accuracy for T60 values obtained from the Combi-En test set. Accuracy is generally highest along the diagonal of the confusion matrix, highlighting where predictions are most precise. Accuracy varies notably across different intervals, with the highest accuracy observed at 93.40% for the $\langle 0, 160)$ interval, demonstrating excellent performance in predicting very low reverberation times. Conversely, the lowest accuracy is 62.81% for the $\langle 672, 800)$ interval, suggesting challenges in predicting the highest reverberation times.

RMSE values further illuminate the average magnitude of errors across predictions. The lowest RMSE is 60.9 ms for the $\langle 288, 416)$ interval, indicating more accurate predictions in this range. In contrast, the highest RMSE, observed in the $\langle 672, 800)$ interval, is 127.6 ms, signifying less reliability in those predictions. Notably, moderate to high reverberation intervals, specifically from $\langle 416, 544)$ to $\langle 672, 800)$, where a substantial number of estimates fall into adjacent lower categories, indicating that the network tends to underestimate the level of reverberation. This is particularly evident for intervals $\langle 544, 672)$ and $\langle 672, 800)$, where many predictions incorrectly fall into adjacent lower categories. The model accuracy decrease as the T60 increase, but RMSE demonstrates, that the real predictions are still usable.

The lower prediction performance of the T60 metric may be due to the smaller training dataset. The training dataset was replicated six times due to noise but the reverberation dataset size was not (because of necessity of saving original T60 labels). For more details on the selection of impulse responses, see the Section 3.1.2.

Table 3.14: Combi-En model: Confusion and classification accuracy for T60

	⟨0, 160)	⟨160, 288)	⟨288, 416)	⟨416, 544)	⟨544, 672)	⟨672, 800)
⟨0, 160)	9207	3739	107	12	58	51
⟨160, 288)	315	22555	837	119	212	153
⟨288, 416)	180	1331	25880	3500	722	316
⟨416, 544)	107	335	1583	9570	7656	842
⟨544, 672)	46	150	619	1468	18630	6056
⟨672, 800)	3	24	92	229	1570	12526
Accuracy (%)	93.40	80.17	88.88	64.24	64.58	62.81
RMSE (ms)	94.8	69.8	60.9	74.7	89.1	127.6

Interesting nuance is that the interval from $\langle 0, 160)$ ms shows a high accuracy but also a high RMSE. This could be explained by the nature of the T60 metric itself. T60 measurements in this interval are tightly bounded by 0. As a result, even small prediction errors can result in relatively high RMSE values. The range is limited and it makes each error proportionally significant.

4 Conclusion

To perform non-intrusive speech quality assessment, a data-driven model based on convolutional neural network was trained. T60 reverberation time, SNR signal-to-noise ratio and STOI speech intelligibility were predicted without the need for reference signals by models.

The models are able to return high accurate SNR predictions on all considered noisy datasets either with or without reverberation. Moreover, combined model even predicts little better than the noisy one, despite the fact, that it resolves more complex task. STOI predictions are stable on all trained models and vary from 0.10 up to 0.12. One exception is the noisy Korean test set (which contains unseen language) where STOI metric reaches up 0.17. So, some language dependency in Noisy models was assumed. With regard to reverberation time detection, the model gives reliable results on both noiseless and noisy datasets with reverberation. Both models exhibit high RMSE on signals with high reverberation. The T60 metric predictions on the noisy reverberated dataset are worse than on the noiseless dataset due to the complexity of the task. As the T60 metric suffers the most from adding noise in the Combined dataset, a decision to extend the training set was made to enhance estimation in the future work.

In the next phase of development, i.e., for diploma thesis, we will concentrate on development of recurrent neural network architecture, with a focus on integrating attention mechanisms such as self-attention. These advancements are expected to enhance the model temporal analysis capabilities, thereby providing a more dynamic and context-aware assessment of speech quality. This approach is not only projected to improve accuracy across all levels of reverberation but also to enhance the model utility in real-world, variable acoustic settings.

Appendix

The source code used to train the models and the sample dataset is located at https://github.com/mothspaws/BP-2024.

References

[1] TAAL, Cees H. et al. An Algorithm for Intelligibility Prediction of Time–Frequency Weighted Noisy Speech. *IEEE Transactions on Audio, Speech, and Language Processing*. 2011, vol. 19, no. 7, pp. 2125–2136. Available from DOI: 10.1109/TASL.2011.2114881.

[2] SMYTH, Tamara. *Music 175: Time and Space*. San Diego (UCSD), Department of Music, University of California, 2016.

[3] GAMPER, Hannes et al. Intrusive and Non-Intrusive Perceptual Speech Quality Assessment Using a Convolutional Neural Network. In: *2019 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. 2019, pp. 85–89. Available from DOI: 10.1109/WASPAA.2019.8937202.

[4] *Recommendation P.563: Single-ended method for objective speech quality assessment in narrowband telephony applications: ITU-T*. ITU-T Recommendation P.563. 2004.

[5] CHANWOO, Kim and Richard M. STERN. *Robust Signal-to-Noise Ratio Estimation Based on Waveform Amplitude Distribution Analysis*. Pittsburgh, 2008. Department of Electrical, Computer Engineering, and Language Technologies Institute Carnegie Mellon University.

[6] EATON, James, Mike BROOKES, and Patrick A. NAYLOR. *A comparison of non-intrusive SNR estimation algorithms and the use of mapping functions*. London, 2013. Department of Electrical and Electronic Engineering, Imperial College.

[7] LÖLLMANN, Heinrich W., Andreas BRENDEL, and Walter KELLERMANN. *Comparative study of single-channel algorithms for blind reverberation time estimation*. Aachen, 2019. Friedrich-Alexander University Erlangen-Nürnberg.

[8] ZEZARIO, Ryandhimas E., Szu-Wei FU, and Chiou-Shann FUH. *STOI-Net: A Deep Learning based Non-Intrusive Speech Intelligibility Assessment Model*. Auckland, 2020. Research Center for Information Technology Innovation, Academia Sinica.

[9] *Python implementation of STOI* [online]. 2007. [visited on 2024-04-04]. Available from: https://github.com/mpariente/pystoi.

[10] *Calculation of the reverberation time* [online]. [visited on 2024-05-06]. Available from: https://sengpielaudio.com/calculator-RT60.htm.

[11] *Room Impulse Response Generator* [online]. 2007. [visited on 2024-04-04]. Available from: https://github.com/ehabets/RIR-Generator/blob/master/rir_generator.pdf.

[12] SERGEEVA, Viktoriia and Jiří MÁLEK. *Určení kvality řečové nahrávky bez reference*. Liberec, 2023. Projekt. Technická univerzita v Liberci.

[13] *How to Easily Draw Neural Network Architecture Diagrams* [online]. 2021. [visited on 2023-01-31]. Available from: https://miro.medium.com/v2/resize:fit:1400/format:webp/1*kQtbGWZgi3n35Qojkg8cFw.png.

[14] *Acoustic scene classification* [online]. 2019. [visited on 2023-02-07]. Available from: https://dcase.community/challenge2019/task-acoustic-scene-classification.

[15] *Libre speech* [online]. [visited on 2022-05-13]. Available from: https://www.openslr.org/12/.

[16] *HUI Audio Corpus German* [online]. 2023. [visited on 2024-01-15]. Available from: https://opendata.iisys.de/dataset/hui-audio-corpus-german/.

[17] *Common Voice Delta Segment Korean* [online]. 2022-2024. [visited on 2024-01-15]. Available from: https://commonvoice.mozilla.org/en/datasets.