Zurich University of Applied Sciences

Audio-denoising with Neural Network Architectures

Author:
Flurin Gishamer

Supervisor:
Prof. Dr. Mark Cieliebak

Institute of Applied Information Technology
Switzerland

January 31, 2021

Zurich University
of Applied Sciences

School of
Engineering

DECLARATION OF ORIGINALITY

Master's Thesis at the School of Engineering

By submitting this Master's thesis, the undersigned student confirms that this thesis is his/her own work and was written without the help of a third party.

The student declares that all sources in the text (including Internet pages) and appendices have been correctly disclosed. This means that there has been no plagiarism, i.e. no sections of the Master's thesis have been partially or wholly taken from other texts and represented as the student's own work or included without being correctly referenced.

Any misconduct will be dealt with according to paragraphs 39 and 40 of the General Academic Regulations for Bachelor's and Master's Degree courses at the Zurich University of Applied Sciences (Rahmenprüfungsordnung ZHAW (RPO)) and subject to the provisions for disciplinary action stipulated in the University regulations.

City, Date: Zürich 31.12.2021

Signature:

The original signed and dated document (no copies) must be included after the title sheet in all ZHAW versions of the Master's thesis submitted.

Abstract

In this work, the influence of background noise, reverberation and frequency filtering on the quality of a subsequent speech-to-text system is investigated (using Mozilla DeepSpeech). Also, three speech enhancement models were used to investigate the extent to which these confounding factors could be neutralized. The results indicate that reverberation has the strongest negative influence, followed by superimposed noise and frequency filtering. The system used for de-noising was able to neutralize the negative influence of superimposed noise best, compared to the other two categories.

Contents

1 Introduction

The aim of the present work is to investigate the influence of different confounding factors on the quality of the results of automatic speech recoginition (ASR) systems. As described in section 3.6, neutralizing the influence of interfering factors with the goal of improving the quality of transcriptions has already been extensively studied in the context of optimizing end-to-end speech-to-text (STT) systems, and such systems are also openly available (e.g. Facebook's fairseq framework [1]). A drawback of this approach is the requirement to have access to the actual STT system in order to adapt its architecture. Furthermore, these approaches are not parametric, meaning that specific parameters such as the intensity of individual speech enhancement components cannot be selectively controlled. The present work aims at providing a reasonable categorization of relevant confounding factors for a subsequent STT system, as explained in section 4.1, and the subsequent application of such confounding factors to a baseline data set in order to measure the concrete impact on its performance. Ultimately, exemplary speech enhancement models will be used to investigate potential improvements with respect to the subsequent STT system. The evaluation of the obtained data is performed by the CEASR framework proposed by Ulasik et al. in [22].

2 Theory

2.1 Acoustic properties of speech

According to Mapp in [2] speech mainly covers the frequency range from 100 Hz to 8 kHz, where higher frequencies up to 12 kHz are only affecting the overall sound and timbre, which suggests that intelligibility is only influenced little by frequencies in this range. This is also reflected in figure 1, that shows the importance of different frequency bands w.r.t. intelligibility.

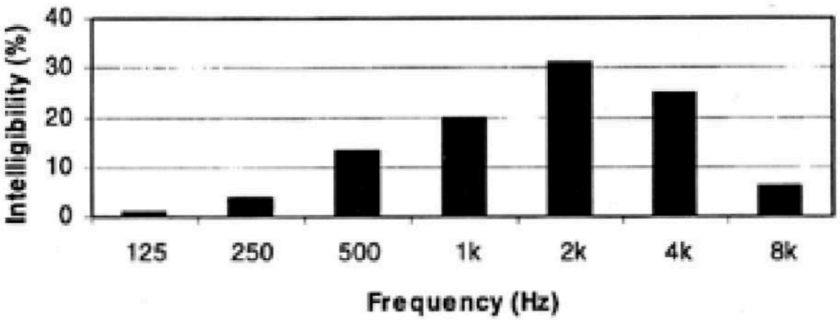


Figure 1: Octave-band contributions to speech intelligibility. Source: [2]

2.2 Measuring the error rate of a system

As stated in [3], in order to adequately assess relative error reduction in STT systems, on the one hand, a test data set containing more than 500 sentences from at least 5-10 speakers and, on the other hand, a metric to adequately measure this reduction is needed.

2.2.1 Word Error Rate WER

A commonly used metric for assessing the relative error reduction is the word error rate (WER). To determine the WER, the first step is to match a recognized utterance of an STT system with a correct reference utterance. In a second step, the number of insertions, deletions, and substitutions are summed and divided by the total number of words in the reference sentence, where Huang et al. define them in [3] as follows:

- *Substitution* **S**: an incorrect word was substituted for the correct word
- *Deletion* **D** a: correct word was omitted in the recognized sentence
- *Insertion* **I**: an extra word was added in the recognized sentence

The equation for calculating the word error rate, using substitutions, insertions and deletions is then:

$$WER = 100\% \times \frac{S + D + I}{\text{no. of words in correct sentence}} \tag{1}$$

2.3 Convolutions in the context of audio Processing

2.3.1 LTI - linear time invariant systems

Linearity: If we consider a digital systems $T\{.\}$, as explained in [3] it can be said that it is linear iff it has the property described in equation 2, for any values a_1 and a_2. In other words for two signals x_1 and x_2 the resulting signal $y[n]$ is identical, regardless of whether the two signals were first processed by the system $T\{.\}$ and subsequently summed, or whether the sum of the signals x_1 and x_2 has been processed by the system.

$$y[n] = T\{a_1x_1[n] + a_2x_2[n]\} = a_1T\{x_1[n]\} + a_2T\{x_2[n]\} \tag{2}$$

Time invariance: As further stated in [3], a system is time-invariant if it satisfies the property given in equation 3. This means that the system $T\{.\}$ always behaves in the same way, independent of time, i.e. if the signal $x[n]$ is delayed by n_0, the output $y[n]$ is delayed in the same way.

$$y[n - n_0] = T\{x[n - n_0]\} \tag{3}$$

Impulse Response: If a system satisfies the properties linearity and time invariance, it can be described by equation 4, where $*$ is the convolution operator and $h[n]$ is the impulse response of the system $T\{.\}$. The impulse response $h[n]$ in turn describes the output of the system when it receives an impulse as input. Huang et al. [3] explain, that to characterize an LTI system, we only need to know $h[n]$.

$$y[n] = \sum_{k=-\infty}^{\infty} (x[k]h[n-k]) = x[n] * h[n] \tag{4}$$

Applying convolutions: Huang et al. explain that the convolution property states that the Fourier transform of the convolution of two signals is equivalent to the product of the Fourier transform of these two signals. Thus, by transforming the signals from the time domain to the frequency domain, the process of convolution as shown in equation 4 can be achieved by simple multiplication.

2.3.2 Relevance in the context of audio

Berners et al. mention in [4] that equalizers, filters and reverbs are such LTI systems. Taking into account the remarks in 2.3.1, it becomes apparent that artifacts resulting from amplification and attenuation of individual frequency bands, or artifacts caused by (time-delayed) reflections of the original signal, can be modeled accurately by recording an impulse response of the system in question (also called sampling) and a subsequent convolution with the target signal.

The situation is different for artifacts resulting from the addition of harmonics, where we speak of (non-) harmonic distortion (depending on whether the partials added by the system are integer multiples of the existing partials, which cannot be modelled accurately with impulse responses.

2.4 Reverb - a brief overview

Suppose the setting of a sound source in a closed room; in such a scenario, we speak of a diffuse field, which, according to MacDonald, can be described as a space where multiple reflections of the same sound source reach the listener simultaneously [5].

Huang et al. explain that in such a scenario, a microphone records, in addition to the direct signal, reflections of walls and other objects in the respective room. They further point out that current Speech Recognition systems are not able to cope with reverberated speech to the extent humans are able to, which leads to poor performance in the presence of reflections, such as those found in normal office spaces [3].

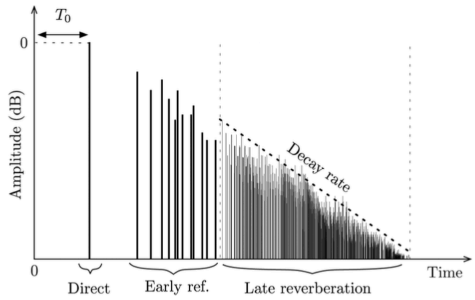


Figure 2: Room Impulse Response. Source: [6]

In figure 2, we see the typical development of a signal and its reflections in a diffuse field. Välimäki et al. explain in [6] that the sound waves emitted by a sound source reach a listener usually in multiple stages. The direct signal reaches a listener after time T_0. In the next phase, referred to in the graphic as early reflections, reflections from nearby surfaces reach the listener. The late reverberation is the stage where the reflections themselves hit the various surfaces in the room and are thus reflected again, with different amplitude i.e. frequency spectra, depending on the condition of the surface material.

The reverberation time RT60 is defined as the period that elapses until the amplitude of an acoustic signal, i.e. its reflections, has decreased by 60 dB. An approximation can be calculated with Sabine's equation (based on Everest in [7]):

$$RT_{60} = \frac{0.161V}{S\bar{\alpha}} \tag{5}$$

Where:

RT_{60}: the time in seconds required for a sound to decay 60 dB.
V: the volume of the room.
S: the boundary surface area,
$\bar{\alpha}$: the average absorption coefficient.

Example: We assume a room with the following dimensions: Length = 10m, width = 8m, height = 4m. For the sake of simplicity, we assume that all walls, ceiling and floor are made of concrete, and we do not consider windows or doors. We now want to determine the reverberation time of this room at a frequency of 1 kHz.

1. Find the absorption coefficient for concrete at 1 kHz , which we

identify as $\alpha = 0.05$

2. Calculate the volume of the room: $V = 320m^3$
3. Calculate the surface area: $S = 304m^2$
4. Apply equation 5 to obtain: $RT_{60} = \frac{0.161 \times 320}{304 \times 0.02} = 8.473s$

To summarize: Reverb can be considered as the sum of reflections of a sound event. The reverberation time RT60, in turn, is the time interval that elapses until the volume of a signals reflections has decreased by 60 dB.

2.4.1 Convolution reverb

Convolutional reverb is a technique where either a synthetically generated or a recorded room impulse response is applied i.e. convolved onto a recording, which results in a signal that resembles the characteristic reverberation pattern associated with the room that has been sampled [8]. In other words one can, by recording an impulse response, apply the reverb properties of a given room to any recorded audio signal. To obtain an impulse response of a room, first a measurement has to be conducted, which can be done by recording the shooting of a pistol, popping an air balloon or even clapping in ones hand. It has to be noted however, that the position of the sound-source relative to the listener is set by the position of the microphone recording above mentioned sound-sources, and can't be altered after the recording process.

3 Related Work

3.1 Noise reduction

Among different architectures of artificial neural networks, previous research has shown that RNNs in particular are a suitable tool for speech enhancement i.e. speech de-noising, and morevoer, as Valentini-Botinhao et al. note in [9], an effective means for improving the results of STT systems. Furthermore, Valin was able to show in [10] that an RNN model (where he uses a GRU-network) with only 4 layers is capable of outperforming traditional MSE spectral estimators for noise reduction, although it should be noted that he combines the neural model with a pitch filter controlled by an algorithmic heuristic.

3.2 Speech source separation

A method that belongs to the category of speech enhancement is speech source separation, an issue that is closely related to the cocktail party effect. Isik et al. could show in [11] that deep clustering is a promising method for single-channel speech separation, and that it also has a positive effect on the results of subsequent STT systems. Their architecture also uses an RNN, but in the form of an LSTM network. In [12], Luo et al. argue that for speech separation on the WSJ0-2mix data set, current state-of-the-art results can be achieved through an architecture based on bidirectional LSTM networks.

3.3 Bandwidth extension

A negative factor to be counteracted by speech enhancement is the loss of parts of the frequency spectrum of a given audio signal. One approach which is applied in this context is the so-called bandwidth extension. In [13], Kuleshov et al. present a procedure called Audio Super Resolution, which can interpolate missing parts of an audio signal by using an achitecture based on a convolutional neural network and thus restore missing parts of the frequency spectrum. The bottleneck architecture [13] described by them seems to be a variant of the autoencoder architecture.

3.4 Speech de-reverberation

Xiao et al. note in [14] that ASR systems have satisfactory results when the microphone is in close proximity to a speaker, but that the performance of such systems is still poor when the microphone is further away from the speaker. To counteract this fact, they use a DNN trained on pairs of clean and reverberant speech signals. They could report an improvement in the results of subsequent ASR systems, but they also mention that their approach leads to distortion in the enhanced speech signals, when a high reverb component is present [14]. Ernst et al. investigated in [15] approaches to de-reverberation, comparing on the one hand a U-Net and on the other hand a GAN with a U-Net as generator, reporting that their approach would outperform other methods, which they support by the comparison through the results of the REVERB Challenge [16].

3.5 Data augmentation

Another approach to counteract the negative influence of confounding factors on the quality of STT results is the application of data augmentation methods. This approach is particularly suitable for use in end-to-end systems. In [17] Park et al. introduce SpecAugment, a method that operates directly on the log mel spectrogram of the input speech signal. They apply three types of deformations to the signals: firstly time-warping, which stretches and compresses a spectrogram in the time domain, secondly frequency masking, which masks a frequency band over the entire duration of the signal, and thirdly time masking, which masks all of the frequencies of a single time step.

3.6 Intermediate representations of speech

An approach that is also suitable for use in end-to-end systems is to calculate an intermediate representation of the speech signals in advance, which is then used as a basis for the subsequent STT task. A practical realisation of this approach is Wav2Vec which is described by Schneider et al. in [18]. In detail they describe their approach as "unsupervised pre-training for speech recognition by learning representations of raw audio" [18]. They justify the use of their approach by the fact that pre-training can be performed using unlabeled data, and the resulting representations are able to improve the quality of STT systems while reducing the amount of labeled training data needed.

4 Methodology

In this section the various confounding factors that have a negative influence on the quality of ASR system are to be categorized and then the general procedure, common to all evaluation data sets shall be described, which includes the application of the CEASR evaluation framework. In a last step the procedure for the individual categories is outlined, comprising the degradation of the baseline data set as well as the subsequent enhancement with the exemplary speech enhancement models chosen for each category.

4.1 Categorization of confounding factors

As a starting point for the categorization, known factors influencing speech intelligibility should be considered, whereby the consideration shall be limited to the acoustic transmission quality of speech signals, which is only a subset of the total number of factors influencing speech intelligibility, which again is a subset of the "context of communication" [19]. These should then be adapted with regard to the possibility of creating synthetic data sets, which in turn will serve as the basis for further investigation.

According to Dong and Lee "speech intelligibility is often degraded due to near-end reverberation and background noise" [20]. Mapp explains that the frequency range of speech lies between 100 Hz and 8 kHz [2], he furthermore notes that "upper frequencies contribute most to intelligibility, with the octave band centered on 2 kHz contributing approximately 30%, and the 4 and 1 kHz bands 25% and 20% respectively" [2]. The inclusion of distorted speech signals was also considered, but was rejected due to the findings of Young et al. in [21], which indicate that the effects of distortion on speech w.r.t. intelligibility can be neglected, this was shown for amplitude distortion in communication circuits, which also would have resembled the setup in the paper at hand. Based on these statements, the following categories are proposed:

- Reverberated speech signals
- Speech signals superimposed by noise
- Speech signals with reduced frequency range

4.2 General procedure

Care was taken to ensure that the procedure for the 3 categories was as similar as possible, as far as this made sense. In the following the steps and configurations are described.

4.2.1 Processing pipeline

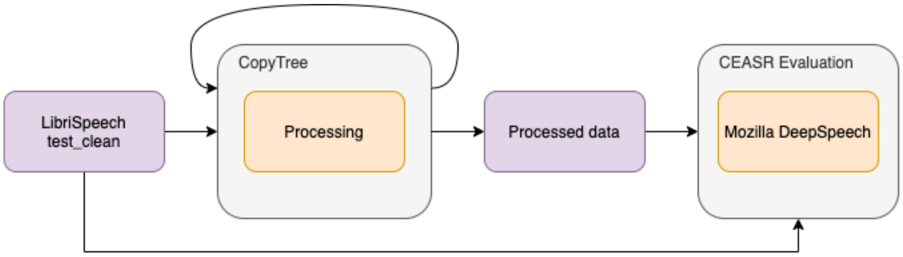


Figure 3: Processing-pipeline

Figure 3 depicts the general processing pipeline. The purple boxes represent speech datasets, the gray boxes represent framework code, and the orange boxes represent concrete processing steps.

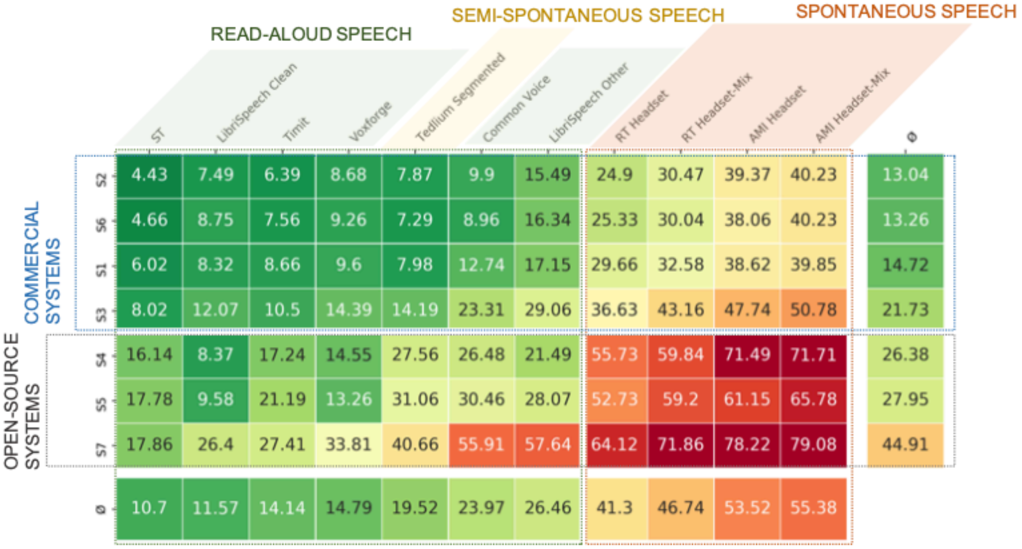
As can be seen in the figure 3, the first step is to pass the test_clean subset of the LibriSpeech corpus to the software framework in order to transform the speech samples. The software was designed in such a way that all effects, both degradation and enhancement steps (the orange box labeled ”Processing” in the figure), are applied in the same way over a file hierarchy, the code of which can be found on: `github.com/flurin-g/DegradedSpeech`. The result is a copy of the file hierarchy containing the modified speech samples i.e. the synthesized evaluation data sets. The same procedure was applied to each of the three defined categories noise, reverb and frequency filtering: First, a degraded version was created, which was then run through the software a second time to apply the enhancement effect to what is shown in the figure as a the round reverse arrow. The concrete implementation details of the degradation and optimization steps are given in sections 4.3, 4.4 and 4.5 for each of the three categories.

All data sets, i.e. both the baseline data set and the synthetic evaluation data sets are in mono. For all evaluations data sets, regardless of their respective categories, as a last step the altered speech samples were normalized, by applying the operation shown in equation 6, in order to prevent clipping.

$$f(x) = \frac{x - \overline{x}}{max(|x|)} \tag{6}$$

In a third step, the unprocessed test_clean as well as the modified data sets were passed to the CEASR framework, which is responsible for transcribing the samples from all passed data sets. In a fourth step, the CEASR framework from Ulasik et al. [22] calculated aggregated values for the WER. By following this procedure it is possible to investigate the influence of the confounding factors defined in section 4.1 on the subsequent STT system. These results are the basis for the subsequent evaluation.

4.2.2 Considerations on the evaluation framework



		ST	LibriSpeech Clean	Timit	Voxforge	Tedlium Segmented	Common Voice	LibriSpeech Other	RT Headset	RT Headset-Mix	AMI Headset	AMI Headset-Mix	ø
COMMERCIAL SYSTEMS	S2	4.43	7.49	6.39	8.68	7.87	9.9	15.49	24.9	30.47	39.37	40.23	13.04
	S6	4.66	8.75	7.56	9.26	7.29	8.96	16.34	25.33	30.04	38.06	40.23	13.26
	S1	6.02	8.32	8.66	9.6	7.98	12.74	17.15	29.66	32.58	38.62	39.85	14.72
	S3	8.02	12.07	10.5	14.39	14.19	23.31	29.06	36.63	43.16	47.74	50.78	21.73
OPEN-SOURCE SYSTEMS	S4	16.14	8.37	17.24	14.55	27.56	26.48	21.49	55.73	59.84	71.49	71.71	26.38
	S5	17.78	9.58	21.19	13.26	31.06	30.46	28.07	52.73	59.2	61.15	65.78	27.95
	S7	17.86	26.4	27.41	33.81	40.66	55.91	57.64	64.12	71.86	78.22	79.08	44.91
	ø	10.7	11.57	14.14	14.79	19.52	23.97	26.46	41.3	46.74	53.52	55.38	

Figure 4: Comparison of STT Systems (DeepSpeech = S5). Source: [22]

As shown by Ulasik et al. in [22] the CEASR framework provides the means to utilize multiple models for speech transcription, and to create a detailed analysis of the performance of the considered STT models as well as the corpora used. In this work, the STT system

Mozilla DeepSpeech was used. It was chosen as an exemplary STT system because it is openly available, and a well documented system, making it a good example for the category of contemporary end-to-end STT models. As can be seen in the heatmap shown in figure 4, Mozilla DeepSpeech ranks in the mid-range of the open source systems, and, as pointed out by Ulasik et al., the open source systems in general rank considerably lower than their commercial counterparts.

4.2.3 Baseline data set

The LibriSpeech corpus was chosen as the source for the speech signals, it contains segments of the recordings of audio-books found in the LibriVox Project [23]. In order to perform the evaluation of speech de-noising systems on the evaluation datasets created, and at the same time to allow the training of such systems on the LibriSpeech corpus, it was decided to limit the samples used in the creation of evaluation datasets to the test dataset of the LibriSpeech corpus, which consists of 5.4 hours of speech recordings [23].

4.3 Speech signals superimposed by noise

4.3.1 Data set used

For the creation of the data set that superimposes noise onto speech signals the UrbanSound8K data set was used, which contains 8732 recordings of urban sounds [24]. Furthermore the samples were limited to recordings with the attribute salience=2, which is described by the authors as occurrences that were subjectively perceived as being in the background of the recording [24]. This reduced the number of recordings used to superimpose the speech signals to just over 3'000.

4.3.2 Processing

To allow the addition of noise from the UrbanSound8K data set with speech signals from LibriSpeech, the sampling rate of the noise samples had to be reduced from 48 kHz to 16 kHz, which is the sampling rate used by LibriSpeech. To account for the different lengths of the samples, the noise samples were truncated, in the case when they were longer than the respective speech sample or repeated to match the length of the speech sample. Similar to the reverberated samples, the noise sample was scaled by a factor of 0.8 and then mixed with the speech sample.

4.3.3 De-noising

the Dual Signal Transformation LSTM (DTLN) described by Westhausen in [25] was used as an exemplary de-noising system, to determine to which extent de-noising is able to improve the quality of subsequent ASR results, when applied to the created evaluation data set superimposed by noise.

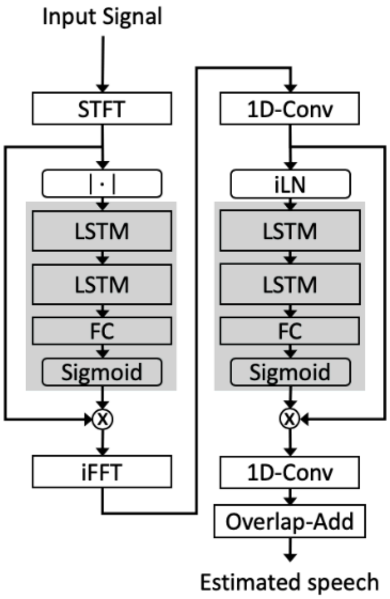


Figure 5: DTLN model architecture. Source: [25]

The DTLN model consists of 2 separate LSTM modules, where each of those consists of 2 LSTM layers followed by a fully connected layer followed by a sigmoid activation function. In the paper of Westhausen those are called separation cores. The input to the first separation core is transformed into the frequency domain by applying a Fast Fourier Transform, its output then "... is multiplied by the magnitude of the mixture and transformed back to the time domain using the phase of the input mixture, but without reconstructing the waveform" [25]. Equation 7 describes the transformation of the input by the first core:

$$\hat{X}_s(t,f) = M(t,f) \cdot |Y(t,f)| \cdot e^{j\phi y} \tag{7}$$

Where:

$\hat{X}_s$: Time-frequency representation of the estimated speech signal
M: Mask with values in the range [0,1] which is applied to Y
$|Y|$: magnitude of the STFT of y
y: The noisy microphone signal
$e^{j\phi y}$: Phase of the noisy signal

As one can see when relating figure 5 to equation 7, the first core is not conditioned to predict the clean speech signal, but rather to predict a mask, which is then applied to the STFT of the noisy microphone signal. The product of this operation is transformed back with an inverse STFT, forming the output of the first core. Before the output of the first core is passed on to the second core, it is transformed by a 1D-convolution, which is used to obtain a feature representation of the processed input and a subsequent normalization layer. After the second separation core, a 1D-convolution layer is used again, to transform the signal back into the time-domain [25].

4.4 Reverberated speech signals

4.4.1 Data set used

As source for the impulse responses, which are used to apply reverb to the voices, a data set called Echothief was used. It includes, in addition to regular rooms, impulse responses of several locations in North America, such as caves, stairwells, underpasses, glaciers and fortresses. [26].

Echothief contains 78 different impulse responses, which represent the characteristic reverb properties of real rooms. To be able to evaluate models on a subset of these impulse responses in the future, the dataset was segmented into a test set and a dev set. To create the evaluation dataset, only the part of the impulse responses were used, which were annotated with "test". Then one of these impulse responses was randomly applied to a speech sample.

4.4.2 Processing

In a first step, the respective speech sample was converted into a pseudo-stereo format in which the mono signal was duplicated identically for the left and right sides, since the impulse responses were recorded in stereo. In a second step the impulse responses were convolved onto the speech sample, using scipy's fftconvolve function. Afterwards the result of the convolution operation was added to the original speech sample, scaled by a factor of 0.8. As a last step, the resulting sample was converted back to mono. In order to keep the results on the subsequent evaluation on ASR systems comparable, the reverb tail was cut off on the processed samples, to match the length of the original speech-samples.

4.4.3 De-reverberation

As an exemplary system to neutralize the reverb component on speech signals, the approach described in "Speech Dereverberation Based on Variance- Normalized Delayed Linear Prediction" (which is also known as wpe method) by Naktatani et al. was used. Since they only provide an implementation in Matlab, for the evaluation Hung's Python implementation was used, which is available on Github [27]

As already mentioned in section 2.4.1, by convolving an impulse response of a room onto a target sample, its characteristics can be transferred. To reverse this process, i.e., to remove the reverberant parts of a recording, Nakatani et al. [28] focus on finding an inverse filter for the rooms impulse response, which can then be used to deconvolve the reverberant recording, thereby aleviating the effect of reverberation. They use a statistical model-based approach to compute said inverse filter, by which they are able to determine it without prior knowledge of the room impulse response. They outline their approach for signals recorded with two microphones, but mention that in practice it has been shown to be effective for signals recorded with only one microphone.

$$x_t^m = \sum_{k=0}^{L_h-1} h_k^m s_{t-k} + b_t^{(m)} \tag{8}$$

Equation 8 defines the model used. Here x stands for the reverberated recording, h for the room impulse response, s for the source signal and b for noise. A comparison with equation 4 shows that this essentially describes the convolution of a source signal with an impulse response.

$$y_t = \sum_{m=1}^{L_m} \sum_{k=1}^{L_w - 1} w_k^{(m)} x_{t-k}^{(m)} \tag{9}$$

Equation 9 can be interpreted in a similar way: It describes how the de-reverberated signal is obtained, namely by applying the inverse filter w to the reverberated recording x. The process is visualized in figure 6:

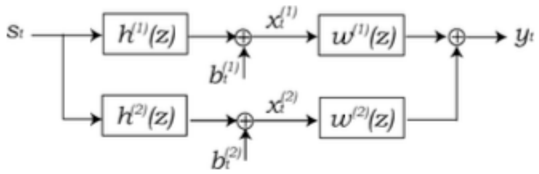


Figure 6: Observation and de-reverberation model: Source [28]

4.5 Speech signals with reduced frequency range

4.5.1 Data set used

No additional data set was needed to create the reduced frequency spectrum evaluation data set, as this was directly obtained from the application of the filters described in 4.5.

4.5.2 Processing

In order to give a practical reference to the reduction of the frequency band, it was decided to use values that are also frequently encountered in everyday life as a reference point. Therefore the frequency band of traditional landline phones and VoIP systems was considered. According to Hecht in [29] the frequency range of traditional landline telephones spans 300 Hz to 3.4 kHz, and the frequency range of VoIP systems extends this range to about 7 kHz.

Consequently, it was decided to remove the low frequencies with a low-cut filter whose cutoff frequency was varied for each sample by drawing the value from a uniform distribution from the interval [250, 350] Hz. The same procedure was applied to the high frequencies, but using the interval [1.5, 3.5] kHz with a high-cut filter. In both cases a 6th order (32db/Oct.) Butterworth filter was used.

4.5.3 Frequency bandwidth extension

As an exemplary system for reconstructing the removed parts of the frequency spectrum during the creation of the evaluation data set, the method presented by Kulsehov et al. in [13] was used. The upsampling of audio signals with the purpose of recovering missing parts of the frequency spectrum is generally referred to as bandwidth extension. Kuleshov et al. assume a low-resolution signal x, with a sampling frequency of R_1. From this signal, a higher resolution signal is to be reconstructed with sampling rate R_2:

$$x = \{x_{1/R_1}, ..., x_{R_1 T_1/R_1}\}$$
$$y = \{y_{1/R_2}, ..., y_{R_2 T_2/R_2}\}$$
$$\text{such that: } R_2 > R_1$$

Their model is inspired by convolutional autoencoders with the addition of residual connections, which they use to encourage the model to learn a hierarchy of features.

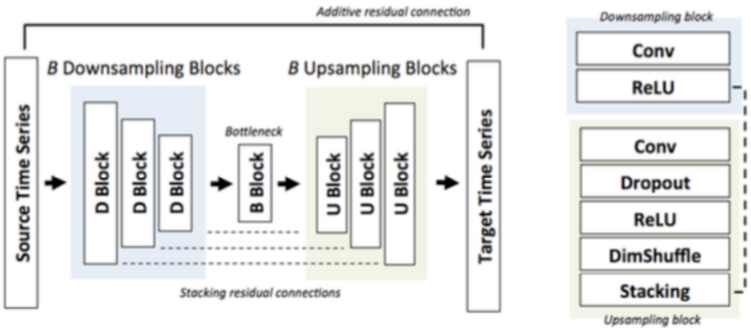


Figure 7: Architecture of the audio super-resolution model: Source [13]

As shown in figure 7 , the model is divided into a downsampling, a bottleneck and an up-sampling part. The down- and up-sampling parts consist of the same number of blocks, where each block consists of a convolutional layer and a batch normalization layer followed by a ReLU activation function. During the down-sampling the spatial dimension is halved, which is reversed during the up-sampling.

5 Results

The results of the evaluation are presented in the following. First, the results concerning the confounding factors i.e. degraded speech samples will be adressed, and then the speech samples processed by the speech enhancement models presented in section 4.3.3, 4.4.3 and 4.5.3 will be addressed.

Degraded speech samples				
Measure	Test clean	Added noise	Reverberated	Frequency filtered
Avg. WER	0.07007	0.48818	0.52963	0.15540
Q1 WER	0.00000	0.12500	0.17583	0.00000
Q2 WER	0.02703	0.42857	0.50000	0.10345
Q3 WER.	0.10000	0.89905	0.91667	0.22222
Min. WER	0.00000	0.00000	0.00000	0.00000
Max. WER	1.00000	1.50000	1.25000	2.00000

Table 1: Influence of the confounding factors on the WER of DeepSpeech. Q1-3 denote 1st-3rd quartile.

Enhanced speech samples				
Measure	Test clean	De-noised	De-reverberated	Bandwidth-extend
Avg. WER	0.07007	0.40984	0.50417	0.81454
Q1 WER	0.00000	0.10000	0.15385	0.72727
Q2 WER	0.02703	0.33333	0.44444	0.88799
Q3 WER.	0.10000	0.72270	0.90513	0.97163
Min. WER	0.00000	0.00000	0.00000	0.00000
Max. WER	1.00000	1.50000	1.66667	1.33333

Table 2: Influence of the confounding factors on the WER of DeepSpeech. Q1-3 denote 1st-3rd quartile.

In tables 1 and 2 the results of the evaluation of the synthetic evaluation datasets performed by the CEASR framework can be seen. As already mentioned under 4.2.1, the evaluation is divided into two stages: Table 1 shows the results of the first stage, in which the influence of the confounding factors on the WER of the following ASR system (in this case Mozilla DeepSpeech) was determined. Table 2 again shows to what extent or if the WER has improved by applying the speech enhancement models.

5.0.1 Degraded speech samples

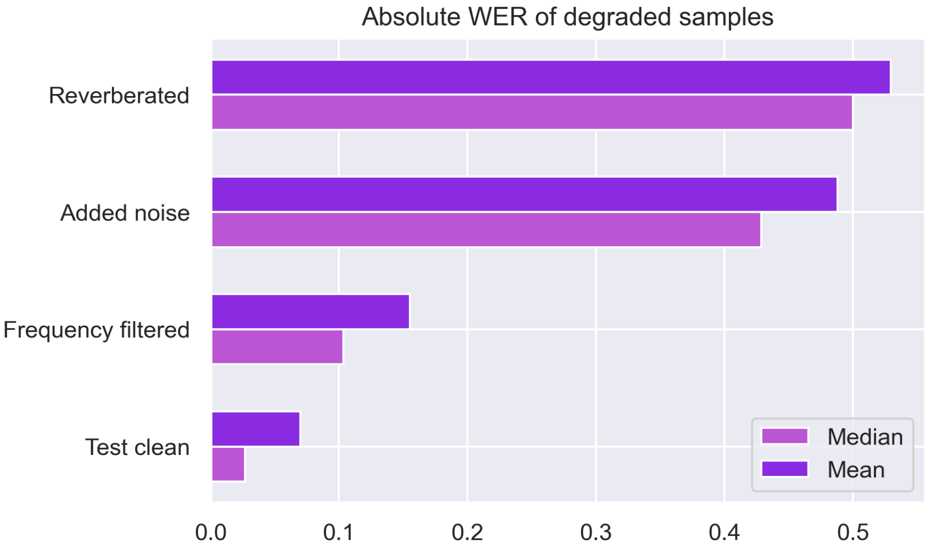


Figure 8: Mean and Median of the absolute WER for the degraded speech samples

In figure 8 The absolute WER of the different confounding factors is shown. For comparison, the WER of the baseline dataset is shown in the last line, denoted as Test clean. The first thing to notice is that frequency filtering has the least effect on WER. However, it must be noted that these can only be compared with the values of reverberated and added noise on a conditional basis, since here, in contrast to the above-mentioned, an addition with an confounding factor did not take place, but the corresponding frequency components on the original signal were removed.

The second thing to note is that adding reverb to the speech samples has a stronger effect on the WER than overlaying it with noise. It should be noted here that in both cases the speech samples were summed with the noise factors, with a scaling factor of 0.8, and furthermore the normalization operation counteracts the influence of any volume differences. Therefore, this result may indicate that reverb has a stronger influence on the WER of the subsequent ASR system, in this case Mozilla DeepSpeech, than the addition of noise.

5.0.2 Enhanced speech samples

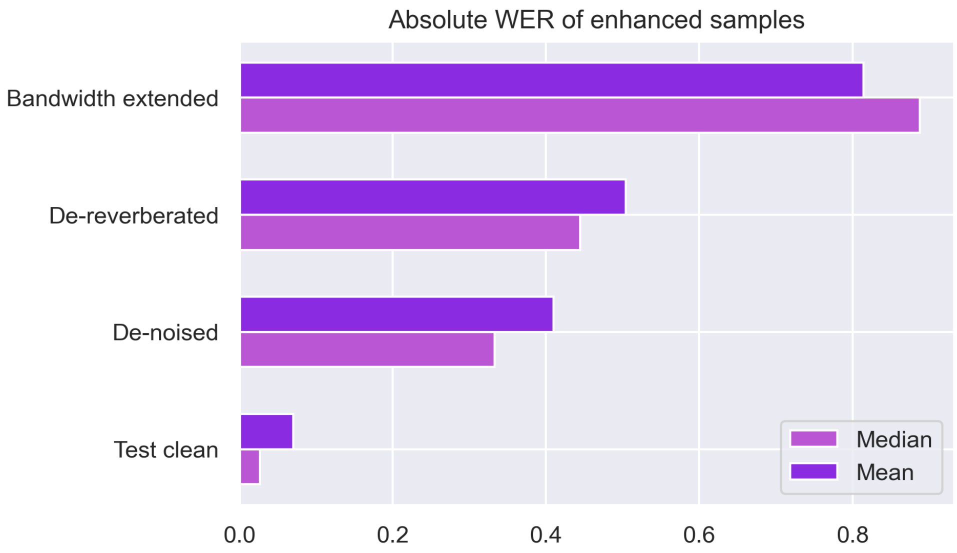


Figure 9: Mean and Median of the absolute WER for the enhanced speech samples

As shown in Figure 9, none of the speech-enhancement models could completely neutralize the respective confounding factors. However, what is striking when comparing these results with figure 8 is that the WER for bandwidth-extended samples is even higher than that of the frequency-filtered ones. It should be mentioned here that Kuleshov et al. make a note in this regard on their website, which justifies these results to some extent:

”the model is very sensitive to how low resolution samples are generated. Even using a different low-pass filter (Butterworth, Chebyshev) at test time will reduce performance.” [30]

Also, as already mentioned in section 4.5.3, Kuleshov et al. implemented the task as an extrapolation of a signal with a lower samplingrate to that of a higher one. While this may suffice for upsampling w.r.t. to the sampling frequency alone, it might not be for recovering lost information about the frequency spectrum, which is exactly what would have been needed to compensate for the loss of said information when using high- i.e. low-pass filters, as done in the degrading step of the frequency filtering.

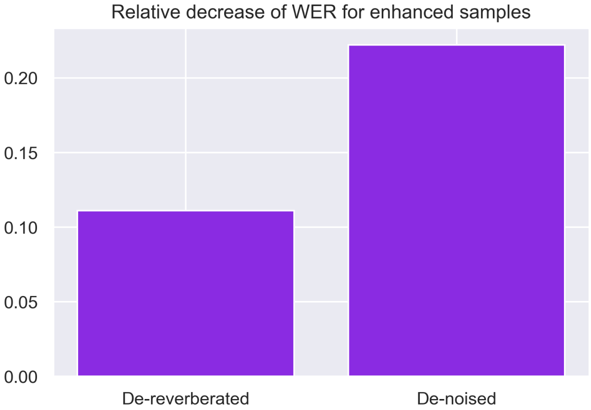


Figure 10: Relative decrease of the median WER for the speech enhancement models for de-reverbertion and de-noising.

Since the negative influence of the bandwidth extended model can already be seen in figure 9, this model will not be used in the consideration of the relative change in WER. Comparing the de-reverberated speech samples with the de-noised speech samples, it is obvious that the de-noising model is able to reduce the WER by a factor of 2 more effectively than the de-reverberation model.

6 Conclusion

As the results in section 5 showed, the effect of frequency filtering on WER was significantly smaller than for the other two confounding factors. This has to be considered under the assumption that the bandwidth limitation of typical telephony systems is a relevant starting point of the optimization. A more in-depth consideration of the influence of frequency filtering should therefore be preceded by the identification of the type of different frequency filtering influences, as an example the loss of high frequency components when Lavaleer microphones covered with clothing shall be mentioned. Most clearly recognizable is the negative influence of superimposed noise on the WER of the STT results, although it could be clarified here whether this is a specific property of Mozilla's DeepSpeech, or whether this can be generalized to other systems. The positive influence of the used de-noising model, in this case Westhausen's DTLN model [25], on the speech samples with superimposed noise is however clearly recognizable, so that the recommendation of de-noising as a preprocessing step for STT systems (in particular for Mozilla DeepSpeech) can be made on the basis of the obtained data. The findings for the negative influence on the WER in the case of superimposed noise can also be applied to the addition of reverb. However, it should be noted that the effect is even more pronounced, and that the model used to reduce the reverb could not neutralize it to the same extent as was the case for superimposed noise. This in turn motivates a deeper exploration of de-reverberation models, as based on the data collected it can be said that the negative influence was the largest for the examinated confounding factors, and the ability to neutralize it could have a high potential for improving STT results, i.e. to reducing the WER of a subsequent STT system.

References

[1] Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of NAACL-HLT 2019: Demonstrations*, 2019.

[2] Peter Mapp. Speech intelligibility – a jbl professional technical note. *JBL Professional Technical Note*, 1(26), 2000.

[3] Xuedong Huang, Alex Acero, Hsiao-Wuen Hon, and Raj Reddy. *Spoken language processing: A guide to theory, algorithm, and system development.* Prentice hall PTR, 2001.

[4] Dr. David Berners and Dr. Jonathan Abel. What is linear phase eq and why should i care?, November 2003. URL: `https://www.uaudio.com/webzine/2003/november/index2.html`.

[5] Scott MacDonald. Sound fields: Free versus diffuse field, near versus far field, Jul 2020. URL: `https://community.sw.siemens.com/s/article/sound-fields-free-versus-diffuse-field-near-versus-far-field`.

[6] V. Valimaki, J. D. Parker, L. Savioja, J. O. Smith, and J. S. Abel. Fifty years of artificial reverberation. *IEEE Transactions on Audio, Speech, and Language Processing*, 20(5):1421–1448, 2012.

[7] F Alton Everest. Master handbook of acoustics, 2001.

[8] William G Gardner. Efficient convolution without input/output delay. In *Audio Engineering Society Convention 97*. Audio Engineering Society, 1994.

[9] Cassia Valentini-Botinhao, Xin Wang, Shinji Takaki, and Junichi Yamagishi. Investigating rnn-based speech enhancement methods for noise-robust text-to-speech. In *SSW*, pages 146–152, 2016.

[10] Jean-Marc Valin. A hybrid dsp/deep learning approach to real-time full-band speech enhancement, 2018. `arXiv:1709.08243`.

[11] Yusuf Isik, Jonathan Le Roux, Zhuo Chen, Shinji Watanabe, and John R Hershey. Single-channel multi-speaker separation using deep clustering. *arXiv preprint arXiv:1607.02173*, 2016.

[12] Yi Luo and Nima Mesgarani. Tasnet: time-domain audio separation network for real-time, single-channel speech separation. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 696–700. IEEE, 2018.

[13] Volodymyr Kuleshov, S Zayd Enam, and Stefano Ermon. Audio super resolution using neural networks. *arXiv preprint arXiv:1708.00853*, 2017.

[14] Xiong Xiao, Shengkui Zhao, Duc Hoang Ha Nguyen, Xionghu Zhong, Douglas L Jones, Eng Siong Chng, and Haizhou Li. Speech dereverberation for enhancement and recognition using dynamic features constrained deep neural networks and feature adaptation. *EURASIP Journal on Advances in Signal Processing*, 2016(1):4, 2016.

[15] Ori Ernst, Shlomo E Chazan, Sharon Gannot, and Jacob Goldberger. Speech dereverberation using fully convolutional networks. In *2018 26th European Signal Processing Conference (EUSIPCO)*, pages 390–394. IEEE, 2018.

[16] Keisuke Kinoshita, Marc Delcroix, Sharon Gannot, Emanuël AP Habets, Reinhold Haeb-Umbach, Walter Kellermann, Volker Leutnant, Roland Maas, Tomohiro Nakatani, Bhiksha Raj, et al. A summary of the reverb challenge: state-of-the-art and remaining challenges in reverberant speech processing research. *EURASIP Journal on Advances in Signal Processing*, 2016(1):7, 2016.

[17] Daniel S. Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D. Cubuk, and Quoc V. Le. Specaugment: A simple data augmentation method for automatic speech recognition. *Interspeech 2019*, Sep 2019. URL: `http://dx.doi.org/10.21437/Interspeech.2019-2680`, `doi:10.21437/interspeech.2019-2680`.

[18] Steffen Schneider, Alexei Baevski, Ronan Collobert, and Michael Auli. wav2vec: Unsupervised pre-training for speech recognition, 2019. `arXiv:1904.05862`.

[19] Marjolein C Coppens-Hofman, Hayo Terband, Ad FM Snik, and Ben AM Maassen. Speech characteristics and intelligibility in adults with mild and moderate intellectual disabilities. *Folia Phoniatrica et Logopaedica*, 68(4):175–182, 2016.

[20] Huan-Yu Dong and Chang-Myung Lee. Speech intelligibility improvement in noisy reverberant environments based on speech enhancement and inverse filtering. *EURASIP Journal on Audio, Speech, and Music Processing*, 2018(1):3, 2018.

[21] Lamar L Young, Jeannette T Goodman, and Raymond Carhart. Effects of whitening and peak-clipping on speech intelligibility in the presence of a competing message. *Audiology*, 18(1):72–79, 1979.

[22] Malgorzata Anna Ulasik, Manuela Hürlimann, Fabian Germann, Esin Gedik, Fernando Benites, and Mark Cieliebak. Ceasr: a corpus for evaluating automatic speech recognition. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 6477–6485, 2020.

[23] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: an asr corpus based on public domain audio books. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5206–5210. IEEE, 2015.

[24] Justin Salamon, Christopher Jacoby, and Juan Pablo Bello. A dataset and taxonomy for urban sound research. In *Proceedings of the 22nd ACM international conference on Multimedia*, pages 1041–1044, 2014.

[25] Nils L. Westhausen and Bernd T. Meyer. Dual-signal transformation lstm network for real-time noise suppression, 2020. `arXiv:2005.07551`.

[26] Chris Warren. Echothief impulse response library, oct 2020. URL: `http://www.echothief.com/`.

[27] Alex Hung. A speech dereverberation algorithm, also called wpe, jan 2021. URL: `https://github.com/helianvine/fdndlp`.

[28] Tomohiro Nakatani, Takuya Yoshioka, Keisuke Kinoshita, Masato Miyoshi, and Biing-Hwang Juang. Speech dereverberation based on variance-normalized delayed linear prediction. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(7):1717–1731, 2010.

[29] Jeff Hecht. Full hd voice will soon give your phone an audio upgrade, Mar 2015. URL: `https://spectrum.ieee.org/telecom/standards/full-hd-voice-will-soon-give-your-phone-an-audio-upgrade`.

[30] Volodymyr Kuleshov, S Zayd Enam, and Stefano Ermon. Audio super resolution with neural networks, jan 2021. URL: `https://kuleshov.github.io/audio-super-res/`.

List of Figures

List of Tables