DEEP LEARNING FOR NOISE REDUCTION IN NOISY AUDIO SIGNALS

Thor Højhus Avenstrup (s224233)
Yue Chang (s222429)
Aarabhi Datta (s232873)
Sergio Monzón (s232515)

DTU

ABSTRACT

(Aarabhi and Sergio) This project aims at reducing the amount of noise present in an audio signal by using a VQ-VAE architecture. Initially, the reconstructed audio was produced using the DAC neural audio codec (Descript Audio Codec). Once the results using DAC were obtained, we then upgraded the model by integrating a HiFi++ discriminator along with additional tailored loss functions. The project involves preprocessing the noisy audio, finetuning the pretrained model and evaluating the model's performance using a metrics like signal-to-noise ratio. The upgraded model depicts greater noise reduction in audio, showcasing a clear reconstructed audio. Hence we see that implementing techniques such as HiFi++ along with additional loss functions, within the VQ-VAE framework offers audio processing and restoration in various applications.

1. INTRODUCTION (AARABHI)

The architecture of a novel model that is tailored to enhance speech clarity by noise reduction in audio signals is understood. The model consists of an initial state, a generator and a discriminator, integrated in a Generative Adversarial Network (GAN) framework. At the initial stage, clear speech is combined with noisy audio to create a combined audio. This forms the input for the encoder-decoder stage. The function of a generator is to create new data instances that are stored in a codebook. The vectors stored in a codebook are then processed and used to create a reconstructed audio signal in waveform and spectrogram formats. The reconstructed audio that is obtained is evaluated by the discriminator which is trained to distinguish between clear audio and the reconstructed audio. The discriminator contributes to the speech fidelity by iteratively improving it's ability to differentiate between the genuine audio and a reconstructed audio. In this report, we look into the specifics of the models components, understanding their roles, the various models we worked on and the significance of this approach in various applications. We will also see the usage of a codebook,encoder and decoder in the working of audio compression and decompressiion to generate a reconstructed audio.

2. RELATED WORK

2.1. AutoEncoders (Yue)

Variational Autoencoders (VAE)[1] is a deep learning technology used to generate models. It combines the structure of autoencoders and the concept of probabilistic graphical models. The core of VAE is to convert input data into the parameters of a probability distribution (usually the mean and variance), and sample from this distribution to generate new data.

More recently, some improved solutions based on VAE have come up. Vector Quantised-Variational AutoEncoder (VQ-VAE) [2] combines the concept of variational autoencoder (VAE) and vector quantization (VQ) technology to quantize the continuous representation of the latent space into discrete codes. It is particularly suitable for processing tasks, like speech, where discretized representations are more efficient.

2.2. Audio codec (Yue)

In the field of audio codecs related to deep learning, Google's SoundStream[3] represents an important progress. As an end-to-end audio codec, SoundStream learns compression and decompression functions directly from the raw audio waveforms and supports different bitrates simultaneously using a single model. This makes it one of the most versatile compression models capable of handling different audio types.

2.3. Audio Synthesis (Yue)

HiFi-GAN[4] is mainly used to generate audio waveforms from Mel-spectrogram with high-fidelity. A multi-scale discriminator structure (MSD) is used, which enables the model to evaluate the authenticity of the audio at different time scales simultaneously, thereby more effectively capturing and improving the details of the audio.

HIFI++[5] is a framework for bandwidth extension (BWE) and speech enhancement (SE), which is improved based on the HiFi-GAN architecture. In the generator, compared with HiFi-GAN it introduces new modules: SpectralUNet, WaveUNet, and SpectralMaskNet to improve performance in SE and BWE problems. For the discriminator part, HIFI++ introduces multi-scale and multi-period discriminators to capture the periodic and temporal details.

Moreover, Residual Vector Quantization Generative Adversarial Network (RVQ-GAN) [6] was proposed as an improved method compared with VQ-VAE. It introduces the concept of residual connections based on vector quantization. The directly quantified latent representation is improved to be progressively refined through a series of residual modules. Each residual module attempts to capture finer-grained features of the input data.

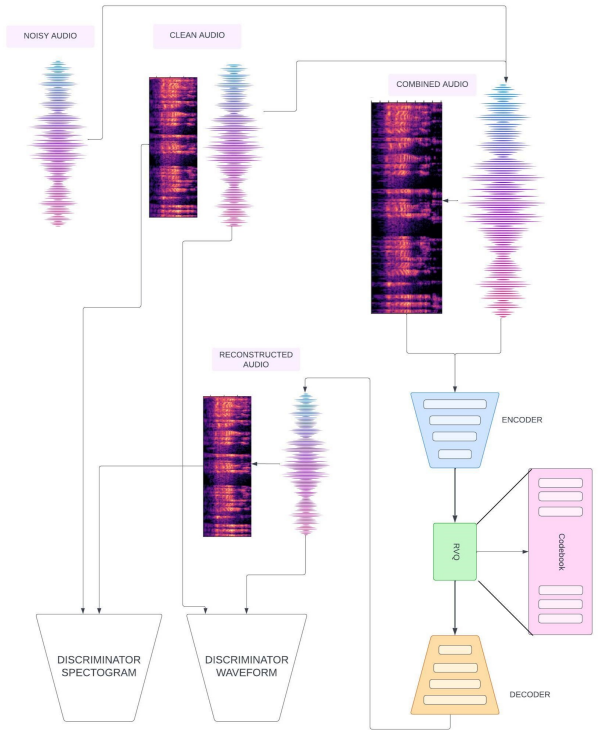


Fig. 1. Neural network model **(Sergio)**

3. MODEL

Fig.1 presents our Generative Adversarial Network(GAN)-based RVQ-model for speech denoising and enhancement based on the pre-trained neural audio codec architecture Descript Audio Codec (DAC). Our approach stems from the synergy of neural audio codecs and discriminative learning to improve the clarity of speech in noisy audio samples. As we can see in the initial step, we take a clear speech from a dataset and combine it with an audio file of noise. Then the combined audio consists of the waveform and the spectrogram. This audio goes through the generator. The function of the generator is to create new data instances. After the encoder-decoder has processed the combined audio, the output is the reconstruction audio consisting of a waveform and a spectrogram. Those files go through a discriminator with the original waveform and spectrogram respectively. The function of that discriminator is to determine whether an input is real (the original clear speech) or if it is fake (the reconstructed audio). Overall, Our goal is to enable generators to generate audio that is indistinguishable from the clear speech data set, and discriminators must be constantly improved to keep their judgment from being "deceived" by the progress of the generator. This adversarial process is constantly looped in training, with the generator constantly trying to generate more clear audio without noise, and the discriminator constantly trying to improve its recognition ability. Ideally, this process results in a generator generating high-quality, authentic data, and a discriminator being able to accurately identify the original audio and generated audio.

In addition, as we know the speech signal is represented as the periodic signal, but traditional activation functions are not periodic, such as Leakage ReLU and SeLU, which means they may not effectively capture the periodic characteristics and pattern of the signal, especially for speech, since these activation functions provide additional periodicity and phase information. So in our experiment, we use both periodic (Snake) and non-periodic activation functions (SeLU) to evaluate if the inductive bias of a periodic activation function is there and leads to increased quality. The experimental results also support this hypothesis.

3.1. Generator architecture (Yue and Aarabhi)

Fig.2 shows the detail of our generator.

3.1.1. Encoder

First, for the encoder, we have the primary convolutional layer which uses a 1D convolution operation to extract features from raw audio. Then we have several encoder blocks. These are the core parts of the generator, and each block includes

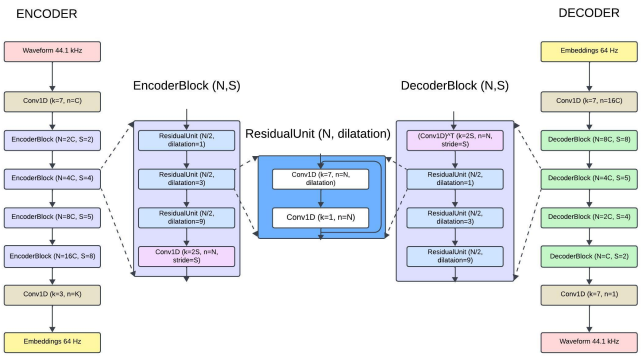


Fig. 2. Structure of generator **(Sergio)**

one or more residual units, which allow the network to learn the extra details of the input data rather than learning the entire data structure. This is because the residual units contain dilated convolution which increases the receptive field of the neural network layer, a wider area of input data can be covered without increasing the number of parameters. It allows the network to focus on learning the difference between the input data and the current representation, i.e., the ”residual”. These differences may be details that were not sufficiently represented in the original latent representation. After multiple encoder blocks, the last 1D convolutional layer further transforms the features to output a final embedding representation, typically with a lower sampling rate. These processes of the encoder convert high-dimensional raw audio data into lower-dimensional representations, which can reduce computational complexity and storage requirements. On the other hand, the low-dimensional representations capture the most important features of the raw audio, with fewer parameters that need to be updated, which can speed up the training process.

3.1.2. Decoder

The purpose of the decoder is to reconstruct the latent representation generated by the encoder back to the original dimension of the audio waveform. Here we convert the latent representation to a higher-dimensional feature representation by a 1D convolution with a convolution kernel size of 7 and a channel count of 16 times the original number of channels. Then we have several decoder blocks which include deconvolution (also known as transpose convolution) operations that step by step upsample the feature map, returning to a dimension close to the original audio waveform. Similar to the encoder blocks, each decoder block also includes one or more residual units with dilated convolution to help recover the temporal detail. The final step of the decoder is a 1D convolutional layer that converts the upsampled features into a final audio waveform output with spectrum, typically with the same sampling rate as the original input waveform, here is 44.1 kHz.

3.2. Discriminator Architecture (Thor)

We use two different discriminator setups in our experiments. The first is the one originally used with DAC (42M parameters) and the second one is from HiFi++ (70M parameters). The DAC discriminator design combines different discriminators: multi-period (MPD) and multi-resolution discriminators (MRD) that allow for analysis of different audio features while the HiFi++ discriminators are based on MPD and multi-scale discriminator (MSD).

3.2.1. Multi-Period Discriminator

The MPD divides the waveform into segments of different lengths (2,3,5,7,11), reshapes them to 2D, pads them and uses 5-layers of 2D convolutional layers increasing in channels to 1024. This allows for a representation similar to images and captures temporal features/artifacts in a specific segment or across segments. In the HiFi++ implementation of MPD the difference is weight normalization.

3.2.2. Multi-Scale Discriminator

MSD focuses on different frequency bands of the audio signal by using 3 discriminators using 1D convolutions of different kernel sizes on progressively downsampled audio. The increased down-sampling results in the discriminators learning different features specific to the sample level. High-frequency features are not present in the downsampled signal as an example.[7]

3.2.3. Multi-Resolution Discriminator

The MRD uses the spectrogram of the audio signal seperating it into multiple frequency bands which allows for analysis of audio across frequencies and captures the spectral features of the signals. To do this STFT with different window lengths (2048, 1024, 512) and hop-length 1/4 of the window length is used to compute the spectrograms at multiple resolutions on which a series of 2D convolutional layers are used to extract features and details.

3.3. Training objective (Thor)

To ensure stable training we utilize a combination of weigthed losses. This includes the L1-loss, Mel-loss, SFTF-loss, SDR-loss, codebook-loss, commitment-loss and GAN-losses.

We utilize two different setups for GAN-losses. The first one is the default discriminator loss used in the DAC architecture which is a combined multi-period (MPD) and complex multi-resolution spectrogram discriminator (MRSD). The second one is the discriminator configuration described in the HiFi++ architecture which consists of both a MPD and a multi-scale discriminator (MSD).

For each of the discriminator setups we use both the feature loss $\mathcal{L}_{FM}(\varphi)$ calculated at each convolutional layer and GAN-loss $\mathcal{L}_{GAN}(\varphi)$ with discriminator parameters φ as losses to the generator.

Generator losses produced by the discriminator for the generator with the parameters θ thus become:

$$\mathcal{L}_{GAN}(\theta) = \lambda_{FM} \sum \mathcal{L}_{FM}(\theta) + \lambda_{Gen} \sum \mathcal{L}_{Gen}(\theta)$$

Where the combined GAN-loss for the generator is given as the sum of the weighed mean of the feature map losses and the generator losses. The feature map and generator losses are computed at different kernel sizes given the clean audio and the generated audio as inputs. The combined loss for the generator becomes the sum of all the weighed losses:

$$\mathcal{L}(\theta) = \sum_{i}^{k} \lambda_i \mathcal{L}_i(\theta)$$

Where λ_i is weight corresponding to the specific loss $\mathcal{L}_i(\theta)$ given the parameters θ of the generator.

4. EXPERIMENT AND RESULTS

4.1. Dataset (Thor)

For speech, we use the VCTK audio dataset from the which includes 109 English speakers, each with approximately 400 unique sentences with silence trimmed, and for noises we use the noise fullband both from DNS-Challenge-4 [8]. We mix samples from these two datasets to create a clean and a noisy speech pair. Noise is sampled randomly and mixed in at different specific SNR-levels [5dB and 10dB] and the samples are randomly cut to a length of 0.5 seconds for the training run. The samples are at 44.1kHz and mono-channel.

4.2. Model training (Thor)

The model is trained for one epoch corresponding to one pass over the VCTK audio dataset with a batch-size of 16 and a learning rate of 3e-4. To optimize the parameters of both the generator (DAC) and the discriminator we use AdamW[9].

We use three different experimental configurations to investigate the effects on the quality of the denoising outputs. The first configuration is a finetune based on the training setup of DAC with the original discriminator, the second one is investigation of the effect of a different discriminator configuration using HiFi++ discriminators, and the third is replacing all the snake activations with SeLU activations in the generator while still using the HiFi++ discriminator.

4.3. Metrics (Thor)

We use the pretrained model provided by torchaudio-squim[10] to estimate the objective measures Scale-Invariant Signal-Distortion-Ration (SI-SDR), Perceptual Evaluation of Speech Quality (PESQ), and Short-Time Objective Intelligibility (STOI) and the subjective measure Mean Opinion Score (MOS). These are used to evaluate the quality of the produced speech after denoising with the network.

4.4. Results (Thor)

Models and measures				
Model	MOS	PESQ	SI-SDR	STOI
HiFi++@SNR5	3.664	3.254	9.863	0.963
HiFi++@SNR10	**3.968**	**3.719**	**11.246**	**0.995**
DAC@SNR5	3.503	2.862	4.246	0.979
DAC@SNR10	3.429	2.625	3.468	0.937
SeLU@SNR10	3.965	3.444	9.092	0.977

The table above summarizes the experiments that've been run. Replacing the discriminator design with HiFi++ improved the quality in all metrics and replacing snake activation function with SeLU while using the HiFi++ discriminator provided comparable MOS-scores but lower scores in PESQ, SI-SDR and STOI. The DAC paper[11] found that replacing ReLU with snake improved performance across all metrics and suggested it was due to the inductive bias inherent in snake as it is a periodic activation function. In our experiment, we found that SeLU performed worse than snake but was still competitive which might be due to the loss of information in negative activations inherent in ReLU but not present in SeLU which might be crucial in the processing of audio data. Using SeLU over snake might be suitable in tasks where lower computational requirements of the network are of higher importance than quality of the generated output.

5. CONCLUSION

This project aims to reduce the noise and enhance the speech components in audio signals using the RVQ-GAN architecture. Initially, reconstructed audio was generated using the DAC decoder, and then the model was upgraded by integrating a HIFI++ discriminator with customized loss functions. This approach not only efficiently processes the original noisy audio but also shows clearer reconstructed audio. Experimental results show that the use of the HIFI++ discriminator and periodic loss functions significantly enhances audio processing and restoration within the RVQ-GAN framework. Especially at higher signal-to-noise ratios, the model using HIFI++ demonstrates superior noise reduction across all metrics. Moreover, we found that replacing the Snake activation function with SeLU led to a decrease in performance, suggesting that periodic activation functions may be crucial in

improving the performance of audio data processing.

In conclusion, this study confirms that in a deep learning framework, the appropriate selection and adjustment of model components, particularly discriminators and activation functions, can significantly enhance the performance of audio denoising. This opens up new possibilities for audio processing and restoration in various applications, including but not limited to speech enhancement, noise reduction, and communication.

Future research can continue to explore the impact of different activation functions and architectural choices on improving the quality of audio signal processing, as well as the feasibility and efficiency of these techniques in practical applications.

6. REFERENCES

[1] Diederik P. Kingma and Max Welling, "An introduction to variational autoencoders," *Foundations and Trends® in Machine Learning*, 2019.

[2] Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu, "Neural discrete representation learning," 2018.

[3] Neil Zeghidour, Alejandro Luebs, Ahmed Omran, Jan Skoglund, and Marco Tagliasacchi, "Soundstream: An end-to-end neural audio codec," 2021.

[4] Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae, "Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis," 2020.

[5] Pavel Andreev, Aibek Alanov, Oleg Ivanov, and Dmitry Vetrov, "Hifi++: A unified framework for bandwidth extension and speech enhancement," in *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. June 2023, IEEE.

[6] Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu, "Neural discrete representation learning," 2018.

[7] Kundan Kumar, Rithesh Kumar, Thibault de Boissiere, Lucas Gestin, Wei Zhen Teoh, Jose Sotelo, Alexandre de Brebisson, Yoshua Bengio, and Aaron Courville, "Melgan: Generative adversarial networks for conditional waveform synthesis," 2019.

[8] Harishchandra Dubey, Ashkan Aazami, Vishak Gopal, Babak Naderi, Sebastian Braun, Ross Cutler, Hannes Gamper, Mehrsa Golestaneh, and Robert Aichner, "Icassp 2023 deep noise suppression challenge," in *ICASSP*, 2023.

[9] Ilya Loshchilov and Frank Hutter, "Decoupled weight decay regularization," 2019.

[10] Anurag Kumar, Ke Tan, Zhaoheng Ni, Pranay Manocha, Xiaohui Zhang, Ethan Henderson, and Buye Xu, "Torchaudio-squim: Reference-less speech quality and intelligibility measures in torchaudio," 2023.

[11] Rithesh Kumar, Prem Seetharaman, Alejandro Luebs, Ishaan Kumar, and Kundan Kumar, "High-fidelity audio compression with improved rvqgan," 2023.

A. APPENDIX

A.1. Github repository

Project Url:`https://github.com/thorhojhus/deep_learning_audio_project`

Deep Learning for Noise Reduction in Clean Speech

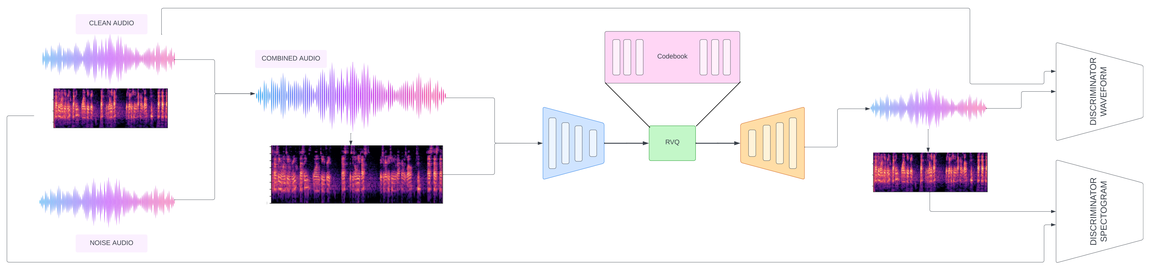
Thor Højhus Avenstrup - s224233, Yue Chang - s222429, Aarabhi Datta - s232873 and Sergio Monzón - s232515

DTU Compute · Technical University of Denmark
Kgs. Lyngby, Denmark

Introduction

- In this poster, our aim is to present the neural network model we have implemented, with the objective of reconstructing audio with sound to obtain clearly audible output.
- First, we will delve into the architecture of the neural network, and subsequently, we will explain various types of losses and other measures that we have employed to ensure high sound quality. Finally, some utilities and reference will be explained.

Neural Network Model

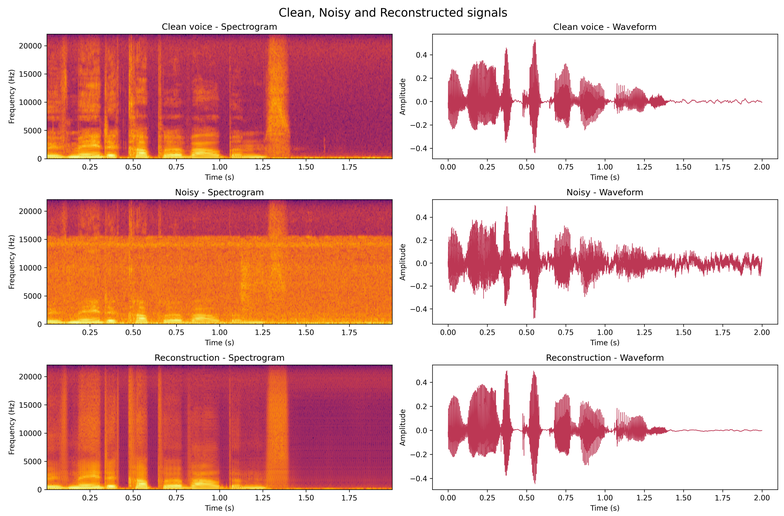


Our model consist on a **initial step**, then a **generator** and then a **discriminator**.

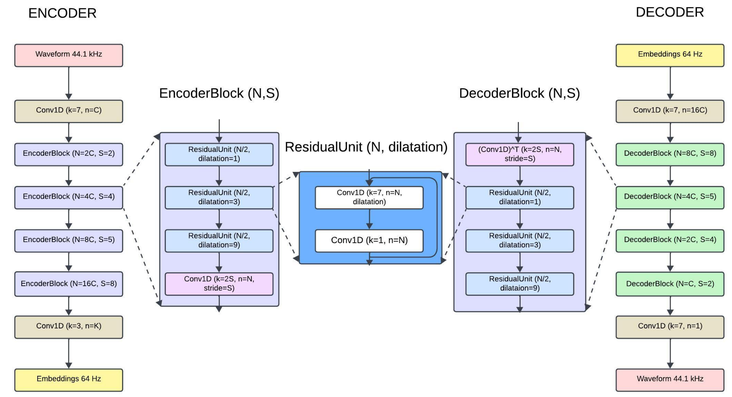
In the **initial step**, we take a clear speech from a dataset and combine it with a noise audio file. Then the combined audio consists of the waveform and the spectogram. This audio goes though the **generator**/encoder-decoder. The function of the generator is to create new data instances. After the encoder-decoder has processed the files, the output is an audio (called reconstruction audio) consisting on a waveform and a spectogram. Those files goes through a discriminator with the original waveform and spectogram respectively. The function of that discriminator is determine whether an input is real (the original clear speech) or if it is fake (the reconstructed audio). The **discriminator** is trained so that it increasingly differentiates both audios better.

To do this we have finetuned a pretrained SOTA Neural Audio Codec (Descript-Audio-Codec (DAC) for speech enhancement and noise removal. We experimented with different discriminators utilizing the DAC discriminator and the HiFi++ discriminator that is superior in vocoding and speech enhancement tasks. In our training loop we combine different audio recordings of noise and speech at different SNR-levels into one file, which we then pass into the generator (DAC). The losses are then calculated based on the reconstructed speech and the corresponding clean speech.
For the discriminators in DAC we use "multi-resolution spectrogram discriminator (MRSD)" that uses spectrograms calculated with SFFT at different resolutions, and for HiFi++ we use both "multi-scale discriminators (MSD)" and "multi-period discriminators (MPD)".

- MRSD computes the spectrograms of the input audio and uses stacks of 2D convolutions on separate bands focusing on different aspects of the audio signal and frequency.
- MSD uses the waveform of the audio-signal at different resolutions using 1D convolutions. Captures rhytmic patterns.
- MPD uses the spectrogram of the audio-signal at different scales using 2D convolutions of different kernel sizes. Captures tonality.



Inside the Generator



Evaluation and Losses

Following subjective measures have been predicted using SQUIM from torchaudio. DAC means default discrimator using in Descript-Audio-Codec and SNR indicating how the level of noise added.

Models and measures				
Model	MOS	PESQ	SI-SDR	STOI
HiFi++@SNR5	3.664	3.254	9.863	0.963
HiFi++@SNR10	**3.968**	**3.719**	**11.246**	**0.995**
DAC@SNR5	3.503	2.862	4.246	0.979
DAC@SNR10	3.429	2.625	3.468	0.937
SeLU(HiFi++)@SNR10	3.965	3.444	9.092	0.9771

- **MOS** (Mean Opinion Score): Subjective method of evaluating voice quality through human listeners. This is estimated using SQUIM [6]
- **PESQ** (Perceptual Evaluation of Speech Quality): Objective method for assessing voice quality by simulating the human auditory system. [7]
- **SI-SDR** (Scale-Invariant Signal-to-Distortion Ratio): Evaluating the fidelity of a voice signal. It measures the similarity between the original voice signal and the reconstructed voice signal. [8]
- **STOI** (Short-Time Objective Intelligibility): Assessing the intelligibility of a voice signal. It estimates the intelligibility of speech by comparing the original voice with the reconstructed voice. [9]

Loss functions used during training:

- **STFT Spectral Loss**: Computes the loss between the Short-Time Fourier Transform (STFT) representations of the original and generated waveforms. More details can be found in the [5]
- **MEL Loss**: Calculates the loss between the Mel Spectrogram representations of the original and generated audio signals.
- **Waveform Loss**: This is the Mean Absolute Error (MAE) between the original and generated waveforms.
- **SDR Loss**: Signal-to-Distortion Ratio (SDR) is a metric used to quantify the ratio of the signal's power to the power of distortion components, offering a measure of the audio quality or the effectiveness of signal separation and enhancement techniques.
- **SI-SDR Loss**: Scale-Invariant Signal-to-Distortion Ratio. It is a modification of the SDR loss that doesn't account for the scaling of the input.
- **Adversarial Feature Loss**: Created from the discriminator and applied to the generator. It encourages the generator to produce features that are indistinguishable from real audio features by the discriminator.
- **Adversarial Generator Loss**: This loss is part of the GAN framework, where the generator is trained to fool the discriminator into classifying the generated audio as real. It quantifies how well the generator performs in this adversarial setup.
- **Commitment Loss**: The purpose of this loss is to encourage the encoder to commit to a specific code in the discrete latent space for each input.
- **Codebook Loss**: The codebook loss ensures that the discrete latent representation effectively captures the structure of the input data and is representative of the training data distribution.

Utilities

- Use of hearing aids. The model is useful because, when used in noisy public spaces (traffic, crowded places, etc.), it can filter out unwanted noise and leave only clean audio, such as a person's voice
- Voice calling and meetings - improved voice quality and compression.
- Voice recordings

References

[1] Neural Discrete Representation Learning, Aaron van den Oord, Oriol Vinyals, Koray Kavukcuoglu

[2] SoundStream: An End-to-End Neural Audio Codec, Neil Zeghidour, Alejandro Luebs, Ahmed Omran, Jan Skoglund, Marco Tagliasacchi, https://arxiv.org/pdf/2107.03312.pdf

[3] Hifi++: A unified framework for bandwidth extension and speech enhancement, Pavel Andreev, Aibek Alanov, Oleg Ivanov, Dmitry Vetrov, https://arxiv.org/pdf/2203.13086.pdf

[4] https://github.com/descriptinc/descript-audio-codec

[5] https://arxiv.org/abs/1810.11945

[6] Vocabulary for performance, quality of service and quality of experience, 2017

[7] P.862 : Perceptual evaluation of speech quality (PESQ): An objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs, 2001

[8] SDR – half-baked or well done?, 2010

[9] A Short-Time Objective Intelligibility Measure for Time-Frequency Weighted Noisy Speech, 2010