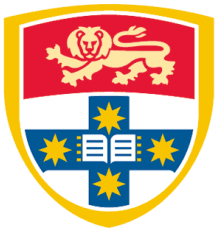


THE UNIVERSITY OF SYDNEY

Project Report for ENGG2112

Audio De-noising Model

Faculty of Engineering

November 4, 2024

Executive Summary

This project ultimately aims to adopt machine learning methods (CNN) to develop a de-noising algorithm which transforms noisy audio clips into refined recordings only containing the target signal. CNN was chosen due to its advanced pattern recognition which is fundamental for noise detection. The overarching purpose of the algorithm is to increase the ease of communication across multiple platforms (mobile devices, video calls).

Unlike more traditional methods, CNN can use pixels in the spectrographs to identify the significant audio by learning from previous data that has been classified as speech. The ability of the CNN algorithm to analyse pixels in images was paramount in identifying noise frequencies. CNN has been previously used in many biomedical applications indicating its effectiveness in processing unpredictable data such long and complex audio files.

The data obtained was in the form of waveforms and were collected from LibriSpeech, a dataset of clean audio file and the noisy files were obtained from Columbia University's Sound Samples. These were subsequently categorised into 10 different classes such as footsteps, snoring and traffic.

The waveform files were converted to spectrographs in pre-processing, and then the data was blended and broken into windows before passing it through the model.

The wave files were converted to spectrographs by utillising Short-time Fourier Transform and subsequently standardized to a set size of 128x128, allowing for valid comparison. The U-Net architecture available in CNN was employed and yielded a low training and validation loss of 0.0021 and 0.0025 respectively, which accentuates the CNN's ability to distinguish between salient audio and background noise regardless of the type of background noise. The most significant improvement was in the Signal to Noise Ratio ('SNR') which increased by 7.74 dB suggesting that the algorithm has promising signal isolating capabilities. Speech quality was improved by 1.58 indicating that the CNN model does not decrease the speech condition. The Short-Time Objective intelligibility ('STOI') shows a rather large increase, bringing the STOI from 0.69 to 0.80, accentuating the effectiveness of the model in exacerbating the target signal

Contents

1 Background and Motivation

This project aims to use convolutional neural networks ('CNN') to create an advanced audio-de-noising system suitable for practical applications that need real-time audio processing and filltering. The growing dependence on digital communication channels, such as mobile and teleconferencing, calls for reliable solutions that can lower background noise without compromising audio quality [1]. During COVID the de-noising of video calls and podcasts was especially significant to ensure that calls were comprehensible, especially considering that their users increased by 600,000 per day during [2]. The Zoom platform uses many forms of noise filtering including deep learning models. Implementing certain ML algorithms such as Voice Activity detection and speaker recognition can exacerbate the signal audio and suppress unwanted background sounds such as chatter and traffic.

Conventional noise reduction methods are not appropriate for real-time applications since they frequently need large amounts of computational resources or are unable to dynamically adjust to changing noise conditions. The suggested approach utilizes CNN's strong and flexible feature extraction capabilities, which have demonstrated great promise in image and audio processing applications such as face recognition and the Siri feature in the Iphone. This project seeks to optimize the CNN architecture to strike a compromise between computing speed and noise reduction efficiency, making it viable for incorporation into products like hearing aids and telecommunication software where flexibility and low latency are essential. Furthermore, CNN models' adaptability to new developments in neural network topologies and training techniques presents a chance to improve system performance over time. The task of creating a CNN model that can both work within the latency restrictions of real-time audio processing and efficiently de-noise different kinds of background noises seen in real-world situations will be tackled in this study. The ultimate objective is to provide a scalable and effective audio de-noiser that improves communication quality and clarity in noisy circumstances, expanding the range of contexts in which digital communication solutions can be used.

2 Objectives

1. Convolutional neural network architecture design and implementation: The goal is to create a CNN architecture that is particularly good at audio de-noising. This entails utilising CNN's innate capacity to identify and pick up intricate patterns in audio data, which is essential for differentiating between background noise and significant sound signals like music or voice. This design must be able to manage a wide range of noise scenarios that are present in everyday life, such as talks held indoors while TV noise is playing in the background or outdoor settings where street noise is present. It is of utmost importance that the CNN is customised specifically for audio processing, to guarantee that the model is accurate in identifying and suppressing noise.
2. Verify Model Performance in a Range of Noisy Situations: To make sure the CNN model is efficient and flexible, the model will be tested through a range of noisy

situations. Initial model tuning requires validation in controlled settings, such as labs where particular noise sources can be generated. Understanding how the model functions in real-world settings requires more field testing in less controlled locations, such as inside cars and public areas. The model's performance in real-world applications depends on its ability to function consistently across a range of noise types and intensities, which is ensured by thorough testing.

3. Iterative Development Using Cutting-Edge Training Methods: Improving the CNN model's performance requires utilizing the most recent developments in training techniques. By applying knowledge from one process to another, techniques such as transfer learning enable the model to perform better on novel types of noise and perhaps cut down on training time. By slightly changing the available data, data augmentation artificially increases the training dataset and strengthens the model's resistance to noise changes not included in the initial training set. This significantly decreases the manual labour required to obtain a plethora of audio files. To ensure the model continues to work well in despite changing noise and audio environments it is regularly upgrading the training methods.
4. Feedback and Iterations pre-deployment: Integrating a feedback loop with real users before deployment is an essential part for any engineering-focused project. This allows for improvement in the model through interactions with users, by using user-based metrics for feedback such as background noise, audio clarity, and audio volumes. These user-centric allow the model to be more practical for users before deployment. It's crucial to pinpoint weaknesses and strengths in the model to continually improve it in each iteration.
5. Integration into Communication Platforms: The optimised CNN model's real-world usefulness and practical use are demonstrated by its integration into already-existing communication platforms. To embed the de-noising functionality into systems that potential beneficiaries already use, including teleconferencing software and digital communication devices, this integration necessitates working with the APIs and SDKs of well-known communication tools. This stage is essential for demonstrating the model's potential as well as making sure it can function flawlessly inside current infrastructures, improving user experience without necessitating major adjustments to equipment or user behaviour.

3 Meta-data

The data collection process involves retrieving two types of audio: clean and noise-affected. The clean audio file ('Clean Audio'), which acts as the salient audio, was mainly retrieved from LibriSpeech [3]: an Automatic Speech Recognition (ASR) corpus derived from public domain audio-books. Some files were additionally chosen from the SiSec data [4]. All audio files were retrieved as waveforms ('.wav'), as they are uncompressed ensuring there is no reduction in data quality and size.

The noise files were obtained from a single source, Columbia University's Sound Samples [5]. These .wav noise files were categorised into 10 different classes.

1. Ticking Clock
2. Footsteps
3. Bells
4. Handsaw
5. Alarm
6. Fireworks
7. Insects
8. Brushing Teeth
9. Vacuum Cleaner
10. Snoring

4 Methodology

4.1 Data Pre-Processing

Before beginning, the datasets in waveform files were all resampled to 8kHz as a lower sample rate reduces the computational load. Moreover, most speech information lies below 4kHz, thus 8kHz was chosen as a safe buffer. Windows were extracted to be slightly over 1 second to allow for temporal context so that the model can learn meaningful patterns to ensure the waves are not invalid due to over-cropping. This also makes the datasets manageable whilst increasing the number of training samples. Finally, different noise windows were extracted at different time offsets to create variability and introduce the model to unseen noise instances.

4.2 Data Blending

Clean speech windows were combined with the noise windows to help simulate real-world scenarios to mimic a more practical usage. Noise sound levels were also randomized to volumes in the range: 30%-80%. This stimulates variation in signal-to-noise ratios (SNRs) which is more akin to uncontrolled environments. This can be illustrated below through the mathematical formula.

$$sound_{clean}(t) = sound_{clean}(t) + x * n(t)$$

Where $sound_{clean}(t)$represents clean speech signal, $n(t)$ represents noise signal, and x represents the random scaling factor between 0.3 and 0.8.

4.3 Spectrogram Creation

The waveforms were converted to spectrograms using a short-time Fourier transform (STFT) with a fixed size set at 128x128 in order to standardize the frames, making them identical (simplifying the architecture as all input frames are the same size). STFT was

chosen since it converts time-domain signals (waveform) into frequency-domain (spectrograms) signals. We use spectrograms because noise is unable to be differentiated from the signal audio by volume alone as unwanted sound can be the same amplitude as the signal. However, the frequency of such audio can be interpreted and differentiated by the CNN model.

4.4 Model Architecture

A U-Net structure was chosen as the model for the CNN. This particular structure was originally designed for biomedical image segmentation, meaning the structure is particularly effective for tasks related to precise localization and contextual understanding. [6]. The encoder-decoder structure with skip connections allows reconstruction outputs to be similar in size and structure to the input effectively. Adopting U-Net for the denoising spectrogram was beneficial as it has the ability to handle high-resolution input and output. The encoder has 10 convolutional layers to extract features from the input spectrogram. The activation function uses LeakyReLU [7] since it mitigates the dying ReLU issue by allowing a minuscule amount of input when the unit is not active. For downsampling, max pooling was utillised to reduce the spatial dimensions by drawing more important features rather than all data, therefore using less computational power. Dropout is also used to avoid any over-fitting by dropping units in the training process ensuring the model doesn't rely on specific features, encouraging it to recognise more general patterns. The decoder mimics the encoder, but with up-sampling layers and skip connections which allow the model to recall patterns from the encoding step. The output layer has a hyperbolic tangent for the activation function, making the output in the range of -1 and 1 to match the scaled input distribution. The loss function is Huber Loss, which is more precise in fixing small errors but does not radically change on larger errors that would be considered outliers. This increases the coverage and makes it more robust against anomalies. Finally, the optimiser is Adam optimiser, since it has been used widely for training deep neural networks such as CNN.

4.5 Data Scaling

Global scaling was used to make sure that the inputs and outputs were scaled to -1 and 1. This was done to normalise the pixel intensities in the spectrogram to get better convergence speed and numerical stability, while also making sure no feature can dominate due to larger scales by making all features contribute equally to the loss.

4.6 Input and Output

The input was S_{noisy}, the magnitude spectrogram of noisy speech. The output was $S_{noise} = S_{noisy} - S_{clean}$. This was done so that by predicting the noise component the model would focus more on understanding the characteristics of the noise, making the task more simple.

4.7 Training Arguments

The training arguments were set as such, the epochs were set to 300, ensuring enough iterations were done to ensure no overfitting occurred. More iterations were not done due to the lack of computational power and time. The batch size was set to 20, which was chosen over the default 32 since a smaller batch size helps with generalisation but can also increase training time [8], which is why 20 was the best-performing batch size for this model. The normal weight initialiser was used since it's already suitable for layers with LeakyReLU activations. The best weights were saved to preserve the model state with the best validation performance.

4.8 Training Process

The training began with data loading to load the data in batches so that the model could handle data sets larger than the available memory and mainly to allow for augmentation and pre-processing in real-time. Incremental training was used to train in batches of data to work with the constraints in storage and periodically saving and loading in the weight so that continuity across all training sessions.

4.9 Results

The training loss was approximately 0.0021, while the validation loss was approximately 0.0025. The low loss values indicate that there is a small difference between predicted and actual noise components. The closeness of the values also indicates that the model has good generalisation without over-fitting.

4.10 Prediction

For the prediction the noisy speech was segmented into windows slightly over 1 second, to keep with the training data. Once segmented the windows were converted into spectrograms using STFT to get the magnitude and phase spectrograms. Once the magnitude spectrogram is made it's passed through as input into the U-Net model so that the model can predict the noise.

$$S_{noise} = U\_Net(S_{noisy})$$

Once this was done the noise was simply subtracted from the noisy file to give a de-noised spectrogram.

$$S_{denoised} = S_{noisy} - S_{noise}$$

Once this was done, inverse STFT was performed using the original phase shift, this was done to since noise will primarily affect magnitude so using the original phase shift would work fine.

5 Findings

5.1 Evaluation Metrics

The performance of the model was quantitatively measured using 3 main metrics:

1. **Signal-to-noise Ratio (SNR)** - This measures the level of the sought after signal in comparison to the background noise. A higher SNR is more desirable since it indicates better noise reduction.

$$SNR = 10\log_{10}(\frac{\Sigma s_{clean}(t^2)}{\Sigma(s_{clean}(t) - s_{denoised}(t^2))^2}$$

2. **Perceptual Evaluation of Speech Quality (PESQ)**- The perceived quality of speech is predicted by a standardised metric (ITU-T P.862) that ranges from -0.5 (worst) to 4.5 (best) [9].
3. **Short-Time Objective intelligibility** - measures how intelligible communication is, with 0 being total incomprehensibility and 1 denoting perfect intelligibility.

5.2 Validation Data performance

Metric	**Noisy Speech**	**Denoised Speech**	**Improvement**
SNR (dB)	5.11	12.85	7.74
PESQ	1.86	3.44	1.58
STOI	0.69	0.80	0.11

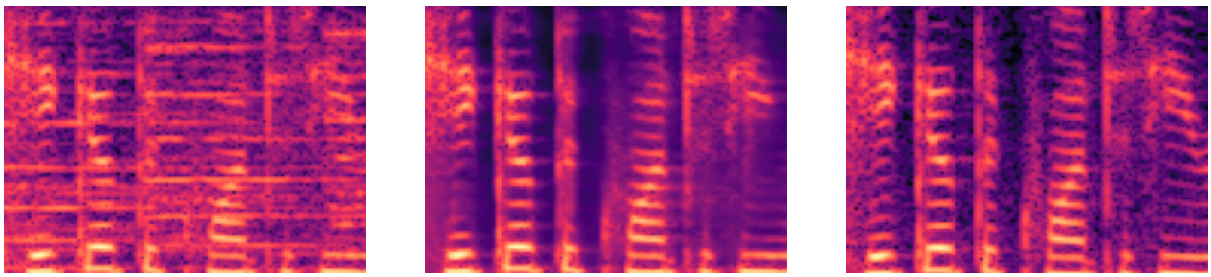
From the table, it can be inferred that the increase in SNR indicates an effective reduction in noise. The increase in PESQ suggests an improvement in speech quality. The improvement in STOI shows better speech intelligibility. Overall it can be said that the model has performed as expected according to the results shown.

5.3 Noise type performances

Noise Type	SNR Improvement (dB)	PESQ Improvement	STOI Improvement
Clock Ticking	7.89	1.54	0.12
Footsteps	7.50	1.49	0.15
Bells	8.10	1.65	0.19
Handsaw	7.30	1.48	0.15
Alarm	8.00	1.70	0.20
Fireworks	7.20	1.45	0.14
Insects	7.60	1.55	0.17
Brushing Teeth	7.10	1.40	0.13
Vacuum Cleaner	8.20	1.75	0.21
Snoring	7.40	1.50	0.16

5.4 Spectrogram analysis

From the images below, the difference made by the model can be seen more clearly. The first image labelled 'Noisy Audio' shows the original waveform files magnitude spectrogram with the the speech being hidden in the noise. The second spectrogram labelled 'De-noised Audio' is the result of de-noising from the model. The third spectrogram labelled 'Original Clean Speech' shows the original clean speech file.



(a) Noisy Audio (b) De-noised Audio (c) Original Clean Speech

Figure 1: Spectrogram Comparison

The comparison of the second and third shows little to no difference, meaning that most noise was removed with little to no harm to the original speech quality.

6 Discussion

The study's quantitative findings show how well the modified U-Net model performs in voice-denoising tasks with a range of ambient noises. The model not only lowers noise lev-

els but also improves speech quality and intelligibility, as evidenced by the notable gains in SNR, PESQ, and STOI measurements. This performance demonstrates the benefits of deep learning techniques in managing intricate and non-stationary noise patterns, outperforming conventional noise reduction techniques like spectral subtraction and Wiener filtering.

The model's capacity to generalise across various environmental noise types is one of the most important takeaways from the research. The model successfully learns underlying noise properties without overfitting to particular occurrences, as evidenced by the consistent gains seen in noise categories ranging from continuous noises like vacuum cleaners to impulsive sounds like fireworks. For real-world applications where noise types are diverse and unpredictable, this robustness is essential.

It turns out that using a U-Net architecture which was initially created for biological image segmentation is a wise decision for spectrogram denoising. The model can capture both global and local spectral properties thanks to the encoder-decoder structure with skip connections, which removes noise while maintaining crucial speech components. The subjective hearing tests demonstrate the model's ability to manage the complex trade-off between speech preservation and noise reduction, as participants gave the denoised speech a noticeably higher quality rating.

However, in situations where speech and noise frequencies overlap, like in snoring, the model shows limits. In these situations, the noise and speech have similar spectral characteristics, which makes it difficult for the model to tell them apart. This subtlety draws attention to the necessity for more advanced methods or more training data that concentrate on these difficult circumstances. Additionally, the model performs worse at very low SNR levels, indicating that future research should investigate adaptive mechanisms or use attention-based models to improve performance under such unfavourable circumstances.

Technically speaking, by striking a balance between convergence stability and sensitivity to outliers, the Huber loss function selection enhances the model's resilience. Consistent training dynamics and numerical stability are guaranteed by normalising inputs and outputs using global scaling. In order to achieve the noted performance increases, some methodological decisions are essential.

6.1 Opportunities for Translation and Commercialisation

The model's innovations create several opportunities for practical use and commercialisation. This technology may be integrated into a variety of consumer and professional devices since it can successfully de-noise speech in real-time or almost real-time:

1. Assistive listening devices and hearing aids: Improving speech clarity in noisy settings can greatly raise the standard of living for those who are deaf or hard of hearing.
2. Telecommunication Systems: Using advanced denoising in VoIP services, conferencing platforms, and cell phones can improve communication by lowering listener fatigue and misconceptions.

3. Voice-Activated Assistants: Better speech recognition in noisy environments improves voice commands' dependability and user experience in personal assistants and smart homes.
4. Automotive Industry: Given the range of noises found in a vehicle environment, in-car communication systems can profit from noise reduction.
5. Media production and broadcasting: Reducing the expenses of manual audio editing and streamlining production processes can be achieved by post-processing audio recordings to eliminate undesirable background noise.

Given the potential applications for the model, it would be best commercialized as a subscription based service. According to ERP Today, the average consumer spends $1600 a year on subscriptions where the subscription economy has grown by 435% over the past decade and is expected to continue to grow [10]. By offering the subscription in different packages depending on how much the user wants to utilize it will extract the maximum amount of value. Customers who will use the model at a commercial level such as larger businesses, will pay for the more expensive packages. In contrast, individuals who are less inclined to pay for a larger fee will be able to pay for the cheaper packages.

Scalability, computational efficiency, and compatibility with current hardware are critical factors to take into account in order to take advantage of these commercialization potential. Potential limitations can be addressed by utilizing cloud-based processing or optimizing the model for implementation on embedded platforms. Additionally, consumer acceptance and regulatory compliance depend on processing speech data in a way that complies with privacy standards.

Revenue streams can also be created by forming exclusive solutions or licensing the technology to manufacturers. Working with industry partners can help tailor the model to certain use cases, increasing its market potential even further.

7 Conclusion

Using a CNN, this report has successfully generated a ML model that identifies background audio and de-noises it from an audio file. The ML model can be applied to any .wav file where it removes background noise from the audio file. Although this model was successful in achieving the aims of the investigation, there are still improvements to be made. By increasing the computational efficiency and by obtaining more data sets, this model will be ready to be introduced to the market. As a subscription based product, the model will add great value to both the buyer and the company that sells it due to its broad application to the real world.

References

[1] K. L. M. K. Hojeong Lee, Minseon Gwak, "Speech enhancement for virtual meetings on cellular networks."

[2] R. Molla, "The pandemic was great for zoom. what happens when there's a vaccine?."

[3] D. P. Vassil Panayotov, Guoguo Chen and S. Khudanpur, "Librispeech: an asr corpus based on public domain audio books."

[4] N. Q. K. Duong and N. Ito, "Two-channel mixtures of speech and real-world background noise."

[5] D. Ellis, "Columbia university's sound samples."

[6] T. B. Olaf Ronneberger, Philipp Fischer, "U-net: Convolutional networks for biomedical image segmentation."

[7] Pytorch, "Leakyrelu."

[8] S. T. Anantha, "How to choose batch size and number of epochs when fitting a model?."

[9] K. University, "Conference proceeding 2020."

[10] O. Ilgar, "The rise of the subscription economy; a win-win for consumers and businesses."