Noise Separation Problem Solution

Problem Description:

We need to build a model that improves the sound quality of the audio data and the main thing we need to remove the noises as best as possible from the audio data.

My Approach:

This problem having multiple solutions, First thing I try traditional way to separate noises from audio data but it did not work well, So I try Exploring Neural network-based solution firstly I try searching for similar kinds of problem and possible solution on the internet, I found some solutions that use CNN based model, but that CNN based approaches are not worked well and then I started reading Papers and found DTLN (Dual-Signal Transformation LSTM Network) it is quite good for our problem and then I started exploring this and I found its good to go and there are many pre-train models as well on this networks, So I decided to train and build a final model on this one.

The approach is Very simple we give the Noisy data to the input and target data as clean speech and train our model.

All the detailed steps performed by me are followings:

Database Selection:

The training dataset was created from the provided audio data of the DNS-Challenge. 500 h of data were created by using the provided scripts. The default SNR range (0 to 40 dB) was changed to -5 to 25 dB to include negative SNRs and limit the total range. All further parameters remained unchanged. The 500h dataset was divided into training (400 h) and cross-validation data (100 h), which corresponds to the common 80:20 % split. All training data was sampled at 8 kHz.

We have created two separate class for each, we give the model to the Noisy single and predict the clean single audio.

The clean audio data having a speech in different languages and some clean singing data.

And for Noisy audio we are having a mixer of the followings:

1. Air conditioner
2. Car horn
3. Children playing
4. Dog bark
5. drilling
6. engine idling
7. gunshot
8. jackhammer
9. siren
10. street music

Feature Extraction:

For the feature extraction firstly I try MFCC (Mel-frequency cepstrum) buts it does not work well-giving lots of false prediction and then I go for the Short-time Fourier transform (STFT) its best for extracting short term change in an audio wav file and its good for our noise separation task also.

Model Development:

For developing the model, we are trying 2 separate models CNN based approach and DTLN based approach we found result given by DTLN are way better for the noise separation problem.

Model Details:

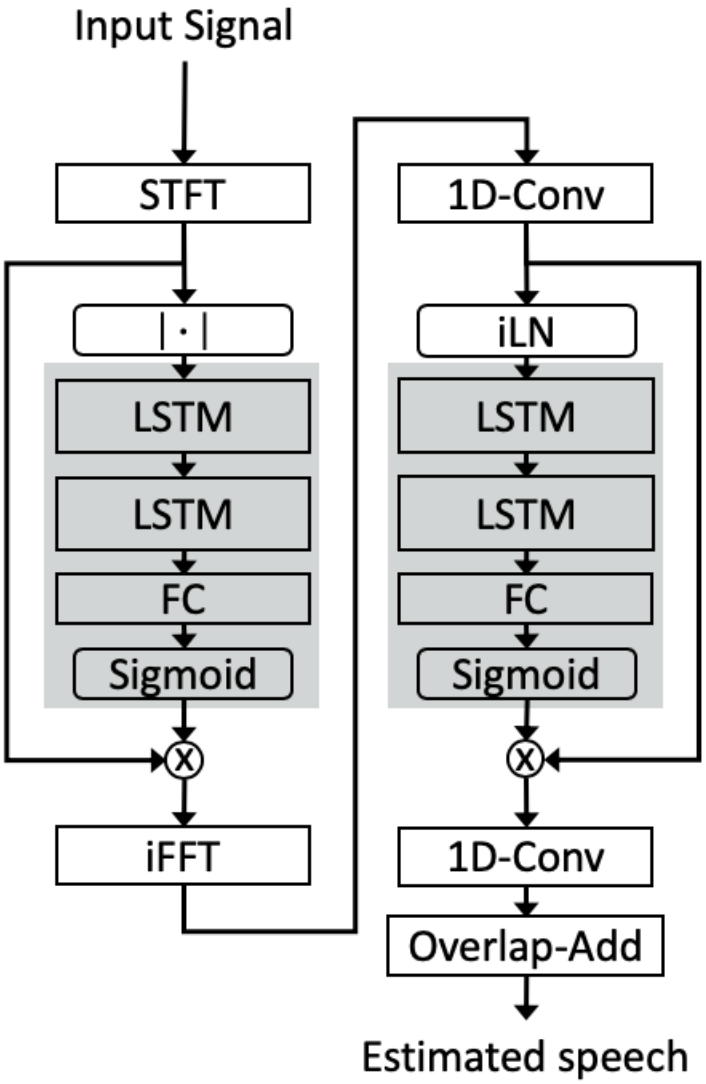


Figure 1: This is a model Diagram we used in this problem, details are given below

Short-time Fourier transform (STFT): For Features Extraction and Giving the input to the model.

LSTM: Long short-term memory network good for retaining long term information.

FC: fully connected layer is used to flatten the array.

Sigmoid: it is an activation function, good for binary classification problem.

iFFT: Inverse fast Fourier transform

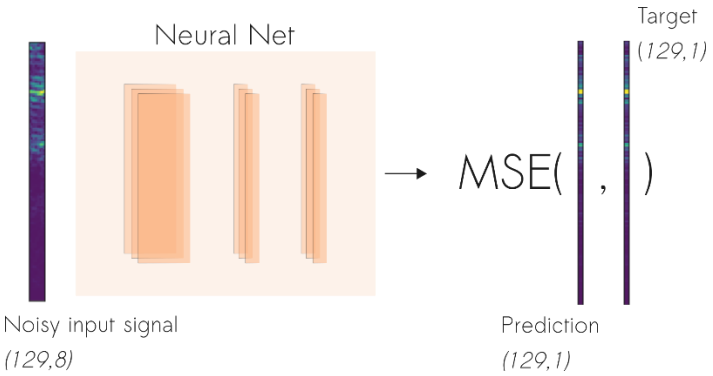
CNN: The frames coming from the first network are processed by a 1D-Conv layer to create the feature representation.

instant layer normalization (iLN): Instant layer normalization is similar to standard layer normalization.

Optimization: For optimization, we are using an ADAM optimizer.

Evolution:

Mean Squared Error between the estimated and clean magnitude STFT of the speech signal.



Final Word:

This is the full approach I have used for building this model, I didn't face much more challenges, I have also copied some codes from the GitHub repo. And come out with this solution.

All the testing steps are given in the GitHub repo I have provided.