Removing the noise from audio/Mp3 using wiener filter and autoencoders

Narendra Chaudhary
Signal Processing and Machine Learning
Department of Electronics & Electrical Engineering
Indian Institute of Technology Guwahati, Guwahati, India
Email: nchaudhary@iitg.ac.in,

***Abstract*—The main objective of the project is to remove the added noise from the signal and compare the performance of wiener filter and deep learning techniques such as auto encoders for noise cancellation.**

I. INTRODUCTION

Speech signals are usually affected by noises during the communication process. For suppressing the noise signal that is combined with the speech signal, Several techniques were developed for among them optimal wiener filter can be the one of the most fundamental approach for noise cancellation, Later on machine learning and deep learning techniques was introduced to attain the better performance.This project shows the capacity of wiener filter and deep learning techniques based algorithm for removal of noise by estimating the signal by means of removing the noise signal form the corrupted signal.

This project consist of two parts, in first part the noise from the corrupted signal is removed using the wiener filtering technique and in second part the auto encoders are used for prediction of the noise free signal.

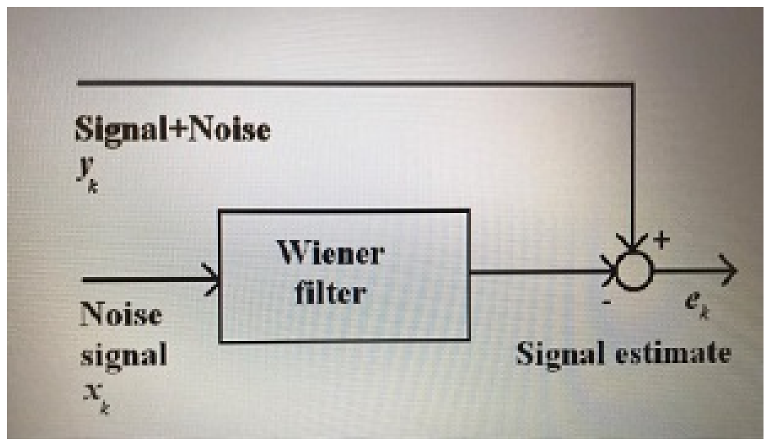
II. REMOVAL OF NOISE USING WIENER FILTERING

A. Filters

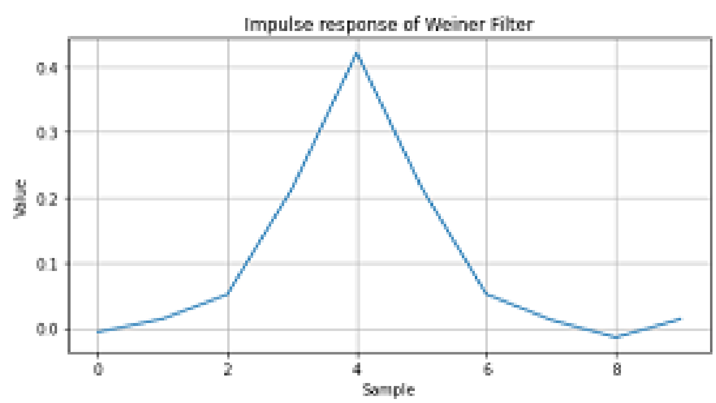
Filters are the basic elements in the signal processing system used to suppress the unwanted signal i.e. noise from the desired signal several techniques are used for filtering. Usual method of estimating the signal corrupted by noise is to pass to it through the filter that tend to suppress the noise and leaving the desired signal this is so called direct filtering. Explain your system model or the dataset. If required, include the figure.

B. Wiener Filters

Wiener filter is a filtering system that is an optimal solution to the statistical filtering problem. The statistical approach to the solution of the linear filtering problem we have to assume the characteristics of the desired signal to be estimated and noise to be removed. So the filter has to be designed in such a way that the output of filter minimizes the effect of the noise from the noisy data which is given as input to the filter. The meaningful approach to the problem of linear filtering is by minimizing the mean square value of error signal which is described as the difference between the desired signal and filter output.

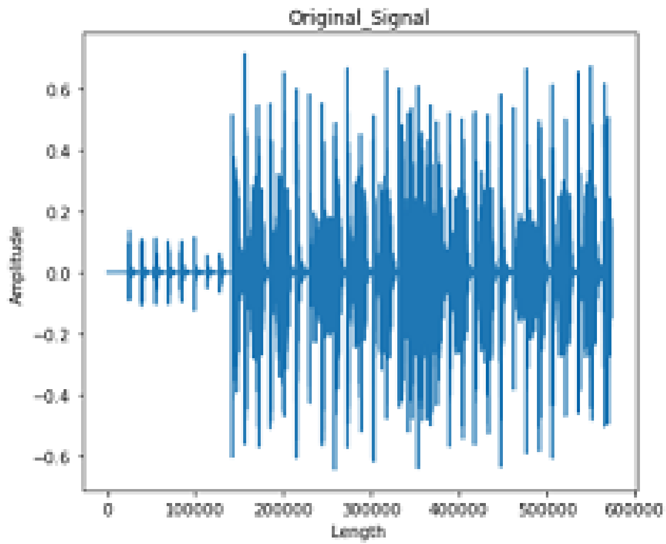


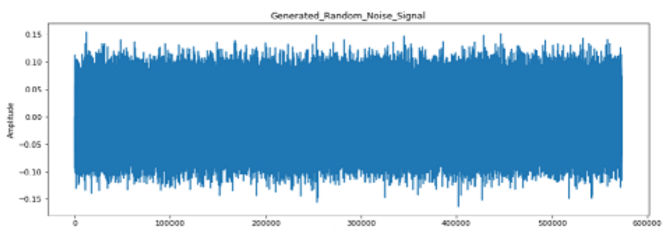
$$Error(ek) = yk - \hat{x} \ , \tag{1}$$

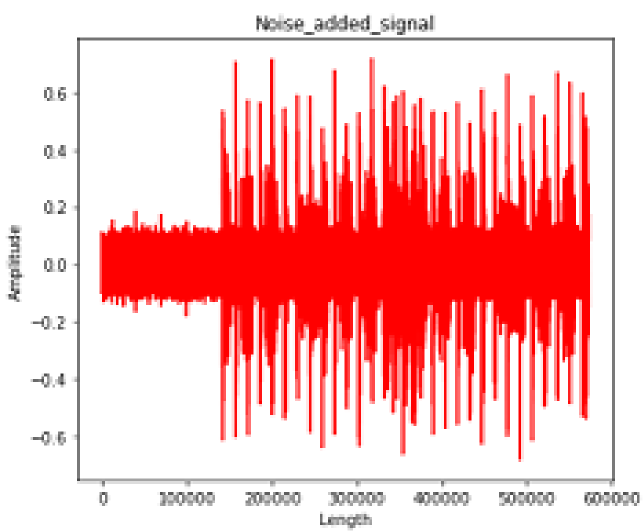


III. METHODOLOGY OR ALGORITHM TO REMOVE THE NOISE USING WIENER

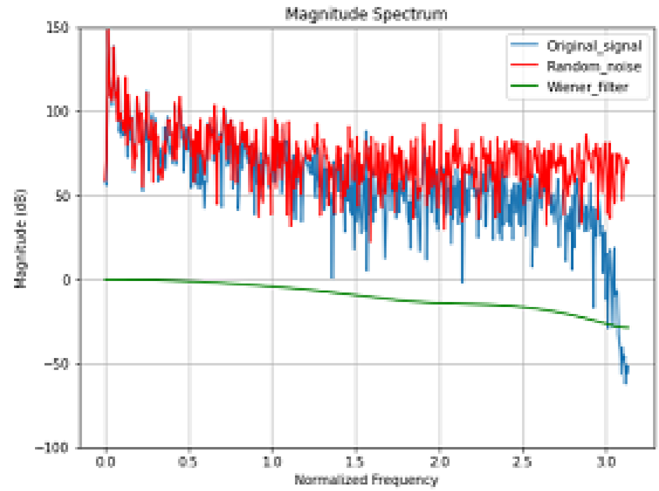
Here I use the audio signal of sample length 573300 and added random AWGN noise to it of the same length keeping SNR = 10 and form the noise added signal from there addition the wave-forms of each are shown below.



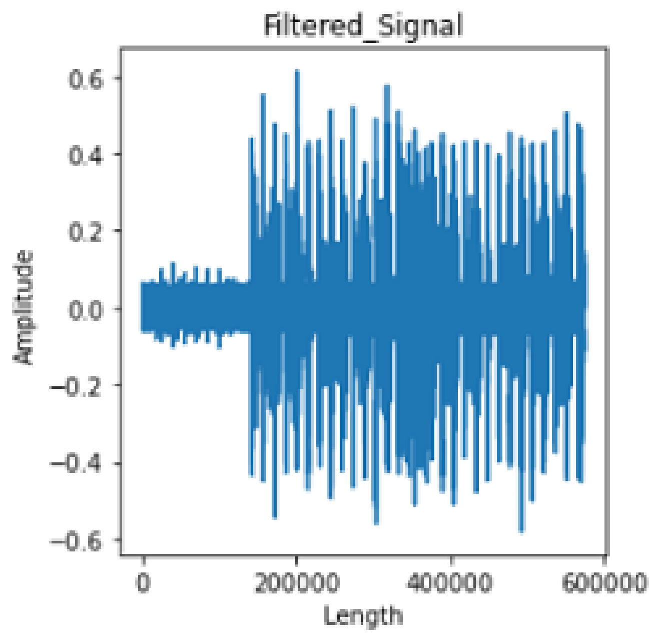




To observe the behaviour of the signals and filter, I draw the joint magnitude spectrum of Original signal, Random noise wiener filter.



After passing our noise added signal to the filter we get the noise removed signal which is as follows



At last the efficiency of the model is calculated and that comes out to be 42.23 ie. the error in the form of noise can be reduced up to 42 percent.

IV. Methodology or Algorithm to remove the noise using Auto encoders

A. Autoencoders

Autoencoder is a neural network with unsupervised training, which means that we don't need to supply a target function. There is only a training set, which is also the target set. A convolutional autoencoder uses convolutional neural networks.

An autoencoder maps the input signal to a lower dimensional representation using its encoder part. In this way it is similar for instance to an audio encoder, which compresses an audio signal into a representation with fewer bits than the original audio signal.

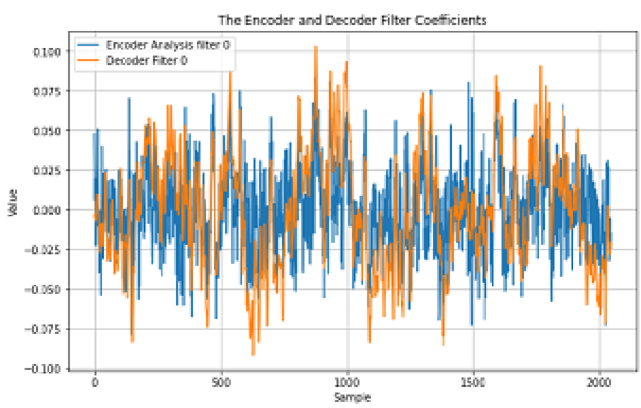
The decoder part of an autoencoder maps the lower dimensional representation back into the higher dimensional representation. This is similar to an audio decoder, which decodes the compressed version back to an audio signal.

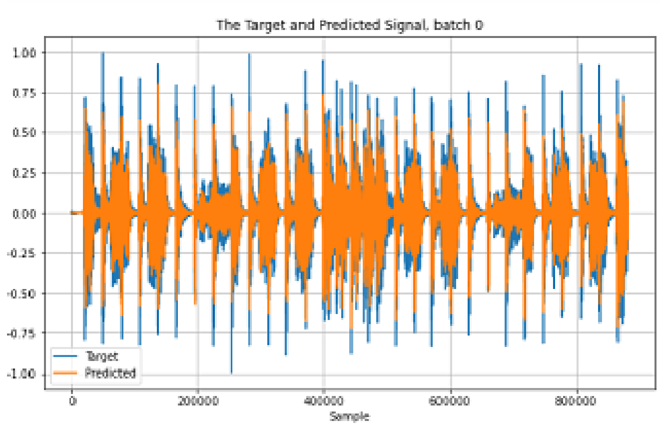
The reconstruction after the decoder part should be as close as possible to the original. Hence the original (the training set) is also the target.

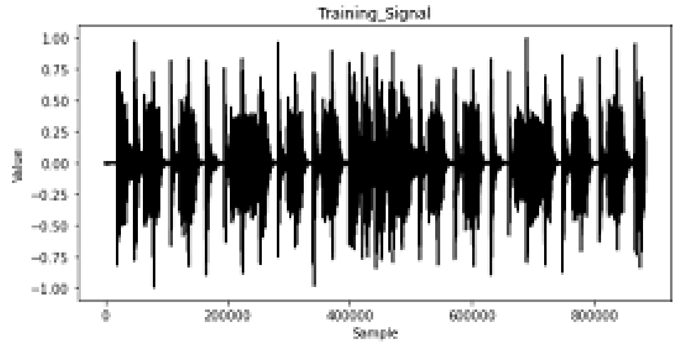
Here we used an activation function ”tanh”, With a ”stride” or downsampling factor of N=1024, and filter kernels of size 2N.

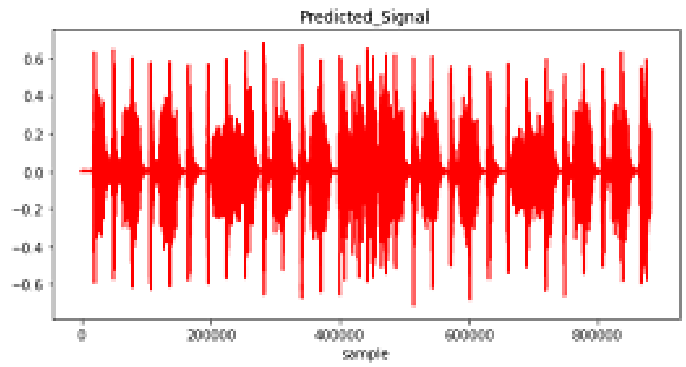
To obtain the same dimension as the input at the encoder output, which is a critical sampled filter bank, we need N filters, or N ”out channels”.

But to reduce the dimensionality after encoding, we have to choose a lower number, which makes it an over-critically sampled filter bank, for instance ”outchannels=32”. This is similar to an audio coder where we simply drop the filter subbands with the highest frequencies. We also have to choose ’padding’ ie. the number of zeros which we have to pad before and after our audio signal before the convolution. To align the output of the decoder with the input signal for the case of symmetric filter kernels (impulse responses), we used a padding of filter-length/2-1, or kernel size/2-1), because then the filtering starts with the first sample at the center of the filter, which corresponds to the the position where the filter outputs the correspondingly filtered sample, for both encoder and decoder.









The stride we used in our autoencoder corresponds to the down and up-sampling rate of the corresponding filter bank. This sampling together with the convolution has the effect of processing the signal in blocks of size of the stride.

It is observed that if we shift our input signal by prepending the number of stride zeros at the beginning (hence delaying it by ”stride” samples), we will get the same blocks, just one block later. But if we prepend a number of zeros which is not an integer multiple of the stride, the blocks will look different,and hence the result will look different, not just shifted.

In this project it is observed by testing the trained autoencoder twice, first with 100 zeros added at the beginning at the beginning of the training signal and secondly by adding 1024 zeros at the beginning of the signal

At the end we find that the output appears more noisy in case of 100 zeros and is bit more clear in case of 1024 zero added at the beginning of the signal.

B. *Limitations*

Since the autoencoder is a neural network so it is complex to design for the large data length. Model take more time to get trained.

C. *Advantages*

More accurate then other non deep learning algorithms

V. Conclusions

In this project we find that we get more accurate filtered signal by using convolutional autoencoders as compare to the wiener filtering.

References

1) A Noise Reduction Method Based on LMS Adaptive Filter of Audio Signals by Yang Liu1 , Mingli Xiao and Yong Tie,3rd International Conference on Multimedia TechnologyICMT 2013)

2) Single Channel Audio Source Separation using Convolutional Denoising Autoencoders by Emad M. Grais and Mark D. Plumbley