SPEECH ENHANCEMENT USING DUAL SIGNAL TRANSFORMATION LSTM NETWORK AND OTHER TECHNIQUES

Novemeber 17, 2020

A Report submitted to the Saarthi.ai on the Speech Enhancement as part of assignment for the position of Deep Learning intern

By
Vidyasree
R151701(Roll-ID)
r151701@rguktrkv.ac.in

Contents

Abstract

Speech Enhancement is used in almost all modern communication systems. When the speech is being transmitted its quality will degrade due to the interference of various disturbances or systems it may pass through. Due to the development and significant rise in the modern Deep Learning or Machine Learning, there is a good improvement in the SOTA State of Art Techniques in the domain of Speech. Recently different techniques evolved in Speech Enhancement include Phase aware Speech enhancement with U-Net, Deep Convolutional Recurrent Network for Phase aware Speech Enhancement, Channel Attention Dense U-Net, Dual Signal LSTM Transformation for Real-Time Noise Suppression, Audio-Visual Speech enhancement, Speaker Agnostic Rapid SE, Self Attentive GANs for SE etc.,

1 Introduction

Speech Enhancement (SE) can be done by using various techniques. Some of the recent papers of SE I mentioned above. Here I focused on a paper named “Dual-Signal Transformation LSTM Network for Real-Time Noise Suppression”. The part of the code I used in the project is the official implementation. In this project I will discuss various techniques involved and I used in the SE.

DLTN- Dual -signal transformation LSTM network for real-time speech enhancement. This approach combines a short-time Fourier transform (STFT) and a learned analysis and synthesis basis in a stacked-network approach with less than one million parameters.

With the uprising of deep neural networks, several novel approaches for audio processing methods based on deep models were proposed. Such models process complete sequences and exploit past and future information of the signals to suppress undesired signal parts.

When designing frame-based algorithms with neural networks, recurrent neural networks (RNN) are a common choice. RNNs have produced convincing results in the field of speech enhancement and speech separation . Long short term memory networks (LSTM) represent the state-of-the-art in separation.

The approach is here to merge both analysis and synthesis approaches in one model by using a stacked dual signal transformation LSTM network (DTLN). The proposed model presented here cascades two separation cores, the first features an STFT signal transformation while the second used a learned signal representation . This order was chosen to create a robust magnitude estimation with the first core and enable the second core to further enhance the signal with phase information.

This combination is explored for the first time in the context of noise reduction and could provide beneficial effects while maintaining a relatively small computational footprint. The stacked network here used is considerably smaller as most previously proposed LSTM networks.

2 Dataset Preparation

The dataset I used here is “**Noisy speech database for training speech enhancement algorithms and TTS models”.** The link for the dataset is “https://datashare.is.ed.ac.uk/handle/10283/1942”. The dataset comprises Clean speech, Noisy speech 48kHz waveforms containing 28 native English Speakers with around 400 sentences each(total 11,572 sentences each) in both training sets whereas the test set comprises of Clean speech, Noisy speech 48kHz waveforms containing 2 native English Speakers with around 400 sentences (total 824 sentences each)in both test sets.

My system configuration is Intel® Core™ i5-3230M CPU @ 2.60GHz × 4 with Ubuntu 18.04.5 LTS. So I used small dataset . The dataset I used in this project is downsampled Clean Speech, Noisy Speech 8Khz waveforms. Training set I used 2328 sentences both in Clean speech, Noisy speech. In Test set I used 300 sentences both in Clean Speech, Noisy Speech.

All samples I used from the Dataset are Downsampled to 8KHZ.

3. Methods

Noise suppression is related source separation problem. It only returns the speech signal and discards the noise.
The first signal transformation of DLTN:

In the time frequency domain, the separation problem can be formulated as follows:
The microphone signal y is described by

$$y[n] = x_s + x_n \quad \text{.....(1)}$$

where x_s and x_n are the speech and noise components of time signals, respectively.
Here the desired signal is the speech signal.
The signal y is transformed with an STFT in a complex time-frequency representation (TF), the TF representation of the estimated speech signal $\hat{X}_s$ can be predicted as follows:

$$\hat{X}_s(t, f) = M(t, f) \cdot |Y(t, f)| \cdot e^{j\varphi_y} \quad \text{.....(2)}$$

where $|Y|$ is the magnitude of the STFT of y
M is a mask (with masking values ranging from 0 to 1) that is applied to Y
$e^{j\varphi_y}$ is the phase of the noisy signal.

$\hat{X}_s$ can now be transformed back with an inverse STFT to $\hat{x}_s$. In this formulation, the phase of the noisy signal ($e^{j\varphi_y}$) is used to predict the clean speech signal.

The second signal transformation of DLTN:

To create the feature representation w_k, the mixture is split into overlapping frames y_k of length L with frame index k. The frames are multiplied by U , which has N × L learned basis functions

$$w_k = y_k U \quad \text{.....(3)}$$

W_k dimension is Nx1

frames y_k dimension is Lx1 and U dimension is NxL

To recover the speech representation d_k from w_k , a mask m_k can be estimated given by

$$\hat{d}_k = m_k \cdot w_k \quad \text{.....(4)}$$

where $\hat{d}_k$ is the feature representation at index k of the estimated speech signal.

$\hat{d}_k$ can be transformed back to the time domain by

$$\hat{x}_k = \hat{d}_k V \quad \text{.....(5)}$$

where: V contains N learned basis functions of length L.
$\hat{x}_k$ is the estimated frame at index k.
The estimated time signal $\hat{x}_s$ is reconstructed by using an overlap-add procedure.

4 Network Architecture and Model Designing

The stacked dual-signal transformation LSTM network architecture introduced has two separation cores.

They are: 1) First separation core containing two LSTM layers followed by a fully-connected (FC)layer and a sigmoid activation to create a mask output.
It uses an STFT analysis and synthesis base. The mask predicted by the FC layer and the sigmoid activation 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.

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

The feature representation is processed by a normalization layer before it is fed to the second separation core.

2) In the second one, the predicted mask of the second core is multiplied with the unnormalized version of the feature representation. The result is used as input to a 1D-Conv layer for transforming the estimated representation back to the time domain. In a last step, the signal is reconstructed with an overlap and add procedure.

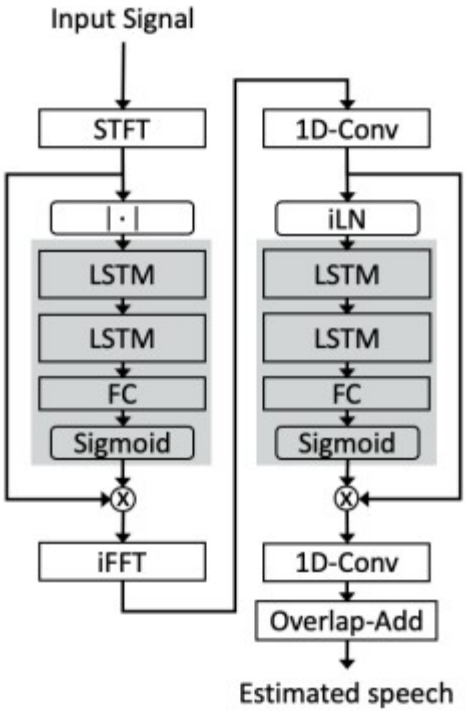

Figure 1: Illustration of the proposed network architecture. The processing chain on the left shows the first separation core using the STFT signal transformation while the building blocks on the right represent the second core with learned feature transformations based on 1D-Conv layers.

5.Techniques to improvise the model

The DTLN model used 4 LSTM layers each comprising of 128 units. The frame size is 32 ms and the shift 8 ms. The FFT size is 512. The 1D-Conv Layer to create the learned feature representation has 256 filters. During training, 25% of dropout is applied between the LSTM layers.
The Adam optimizer is used with a learning rate of 10e-3 and and a gradient norm clipping of 3. The learning rate is halved if the loss on the validation set does not improve for three consecutive epochs. Early stopping is applied if loss on the validation set does not decrease for ten epochs. The model is trained on a batch size of 32, and each sample has the length of 2 s. Original model is trained for 200 epochs on Nvidia RTX 2080 TI. The model I trained comprises of small dataset where I ran only for 10 epochs on Intel I5 processor where each epoch lasts for 180 sec.

We can still increase the LSTM layers but the model parameters with rise.
We can increase the units of LSTM layer like 256, 512,1024 units.
We can still try with different optimizers like NAG,SGD, RMSprop,ADA grad etc.,
Annealing learning rate can be applied.

Since transformers didnot perform well unlike different NlP tasks like Machine/Speech translation, we can't focus on them.
We can try for attention based Speech Enhancement mechansims where humans can focus more on the important speech components using high attention while perceving unimportant region in low attention and can adjust the focal point over time.

6 Other Approaches in deep learning for speech enhancement

In 2016, a model based on signal-to-noise ratio (SNR) aware Convolution Neural Network (CNN) was put forth for Speech Enhancement . This CNN model can efficiently handle the local temporal and spectral speech signals. Hence, the model can effectively separate the speech signals and noise from an input signal. Two SNR-aware algorithms were proposed using CNN with the intention of improving the generalization capability and accuracy of these models.

Waveform domain approaches for speech enhancements have been recently suggested: WaveNet , Generative Adversarial Networks (SEGAN). These waveform domain models operate on samples of speech by modeling the enhancement task in raw speech waveform. Therefore, they have the potential of using phase information if properly designed.

SEGAN:The SEGAN consist of two neural networks,namely, Generator and Discriminator. The Generator network is inspired by Autoencoder architecture. The Generator encoder consists of 11 layers of stride-2 convolution with growing depth, resulting in a feature map at the bottle-neck of 8-time steps with depth 1024. This feature map is concatenated with latent vector "z", sampled randomly from uniform noise distribution. The resultant concatenated vector is input to an 11-layer up-sampling decoder, with skip connections from corresponding input feature maps. The least square based loss function is used to train SEGAN with additional L1 norm to preserve the structure of the enhanced signal.

SE-FFT: SE-FFTNet model explores the underlying time domain structure of speech and noise which is important for enhancement. The wider dilation in the initial layers of the model enables it to learn clean speech structure effectively from the input noisy mixture. The results confirm that the model with a decreasing dilation pattern over depth (SE-FFTNet) performs better than the model with increasing dilation pattern (SE-InvFFTNet). This finding on the influence of dilation width will be useful while implementing the new architecture in future speech enhancement models.

7. Results

Since I trained only on few training samples, the outresults haven't performed as good as the orginal model trained on Large dataset in GPU. But the model I trained performed better in the removal of some part of noise.

8. Future Work

I can still try various networks for SE as mentioned above like using GAN's, attention(local,global,multihead), GRU's,dilation methods etc.,

9. Conclusion

This DTLN is really good network for Speech Enhancement by using less number of parameters. This model brings good results on small amount of datasets trained on less epochs also. On further research work we can implement this on real time processing applications also.