2022 IDL Midterm Report : Microsoft Teams to Cloud Speech Enhancement

Hyun-Kuk Lim*, **Chanwoo Kim***, **Taeyoung Chang***, **Urvish Takker***

Carnegie Mellon University
Pittsburgh, PA 15213
{hyunkukl, chanwoo2, tchang3, udt}@andrew.cmu.edu

Abstract

With an increasing trend in remote work, conferencing platforms are significantly becoming important. 'Microsoft Teams', a popular platform for real-time conference, is one of them. 'Microsoft Teams' uses its speech enhancement model because noises can provide unsatisfactory quality of speech and eventually disrupt conversations. Suppressing them is crucial for a stable meeting. However, it is not easy due to some challenges. Since it has to be real-time, enhancement should proceed fast enough in time-limited context. And even if it suppresses noises well in technical manner, perceptual quality and intelligibility, which are essential since the enhanced output is specifically for human hearing, are not guaranteed. Hence, we improve (not yet done...) real-time speech enhancement model FullSubNet and Demucs. To ensure perceptual quality and intelligibility, auxiliary loss, which is defined using acoustic metrics, is also used. (Details about changes we make and the evaluations, and our beating MS teams in certain way)

1 Introduction

Speech enhancement has become one of the most important technologies in a remote working environment. Meetings done through video conferencing platforms such as Zoom or Microsoft Teams are more prevalent than ever, and extracting clean speech from a noisy environment in real time is becoming more important to avoid interruptions during a conference. However, solely extracting clean speech in real-time is a difficult task, especially with imperfect information on the level of noise in the future. Furthermore, suppressing noise can also lead to the suppression of clean speech as well, removing essential parts of a conversation. Losing essential signals from speech may further degrade its quality, and to prevent this, noise must be filtered out the moment it is detected in audio. Thus, we want to build a model which would perform better in real time.

Our goal is that when we get noisy speech data derived from synthesizing clean speech and noise, we want to build a model (where the basic structure is FullSubnet and some properties of Demucs are added in a clever way? Not determined yet…) that yields a speech close to the clean speech. In other words, we want to build a model that fulfills a noise suppression while perceptual quality and intelligibility of the output is at least competitive with the noise suppression output of Microsoft Teams.

Here, the input is a spectrogram transformed from the synthesized speech file. The spectrogram can be viewed as a 2D tensor of a shape with (T, F). For each time $t \in \{1, \cdots, T\}$, the wave form of short time segment near t is Fourier-transformed (STFT) and the absolute value of this short term fourier transform at each frequency f is stored as F-dimensional vector. Our model is defined as mapping a spectrogram of noisy speech data to the training target cIRM which stands for complex

[*] Equal Contribution (random order)

Ideal Ratio Mask([10]). Since the desired output of cIRM can be viewed as the complex ratio of clean speech spectrum to noisy speech spectrum, training via the loss of measuring distance of actual output and the desired output can be interpreted as knowing well where the noise exists in time-frequency domain so that we can suppress such noise.

2 Related Works

2.1 Demucs

The authors of this paper([2]) utilize convolutional and recurrent neural networks to separate noise from the clean speech in waveform audio. By processing corrupted audio through convolutional encoder-decoder architecture with skip-connections they propose that clean speech can be extracted from latent representations of input audio. Furthermore, to account for sequential information in the audio a recurrent neural network is injected between encoders and decoders, which the decoder uses to reconstruct clean speech. To account for real-time noise suppression the authors accumulate standard deviations of audios only up to the current position for normalization, with paddings to account for a 3ms lookahead.

2.2 FullSubNet

This paper([4]) introduces a full-band and sub-band fusion model, FullSubNet, which enhances single-channel speech. Full-band model has its strength in finding global spectral patterns and cross-band dependencies. On the other hand, sub-band model has its strength in finding local spectral patterns and stationarity, which is an important factor that distinguishes noises from clean speech. Since both models have their own advantages, FullSubNet stacks both models to take both advantages. The whole speech spectrogram goes through a full-band model, and the output is concatenated with sub-band unit. Then the concatenated ones become frequency-wise inputs of the sub-band model. Hence, the sub-band model can find local patterns with complementary information captured from the full-band model. Besides, this model exceeds the high-ranked models in the DNS Challenge (INTERSPEECH 2020).

2.3 eGeMAPS

This paper([3]) suggests a minimalistic set of standard acoustic parameters such as Pitch, Jitter, Shimer, Loudness, Harmonics-to-Noise Ratio, and Harmonic difference. It is selected based on three criteria: the potential to discern physiological changes in voice production, the success in the past literature, and the theoretical importance. Through the experiments, the paper argues that, unlike the large brute-forced feature sets, the minimalistic parameter sets might reduce the danger of damaging generalization capabilities to unseen data. The authors proposed these parameters as the common baseline for evaluating future research.

2.4 Auxiliary Loss using eGeMAPS

Although noise suppression or removal is a significant part of the speech enhancement, we should also pay attention to retaining the perceptual quality and intelligibility of the speech signal. Traditionally, deep neural networks based approaches are trained exploiting pointwise differences in time-domain or time-frequency-domain, but this loss has been shown to be insufficient to bring out high perceptual quality of enhanced speech. To improve perceptual quality, two papers([11], [12]) suggest adding the loss derived by the difference between clean speech and enhanced speech with respect to eGeMAPS features as an auxiliary objective. For expressing the eGeMAPS features, the first one([11]) uses summary statistics of acoustic parameters, known as the functionals, per utterance, whereas the second one([12]) utilizes temporal acoustic parameters, or the low-level descriptors (LLD), of each utterance, allowing the optimization at each time step. Also, both papers propose to train eGeMAPS prediction network that can be plugged into other learning pipelines, since existing calculations for eGeMAPS are non-differentiable.

3 Model Description

3.1 FullSubNet

The model is a combination of a full-band model G_{full}, and a sub-band model G_{sub}. An input for G_{full} is a sequence of noisy full-band magnitude spectral features

$$\widetilde{\mathbf{X}} = (\mathbf{X}(1), \cdots, \mathbf{X}(t), \cdots, \mathbf{X}(T)) \in \mathbb{R}^{F \times T}. \tag{1}$$

And each feature consists of complex-valued time-frequency (T-F) bins of a certain time frame t

$$\mathbf{X}(t) = [|X(t, 0)|, \cdots, |X(t, f)|, \cdots, |X(t, F-1)|]^T \in \mathbb{R}^F. \tag{2}$$

The full-band model G_{full} outputs a spectral embedding with the size same as $\widetilde{\mathbf{X}}$, which captures the global context. Then the part of the embedding corresponding to each frequency is concatenated with the noisy sub-band signal of that frequency. Note that a noisy sub-band signal of a frequency is composed of the frequency and $2 \times N$ adjacent frequencies. Hence, there are total F sequences as the inputs for G_{sub}. An input sequence for the frequency f is

$$\widetilde{\mathbf{x}}(f) = (\mathbf{x}(1, f), \cdots, \mathbf{x}(t, f), \cdots, \mathbf{x}(T, f)) \in \mathbb{R}^{(2N+2) \times T} \tag{3}$$

where

$$\mathbf{x}(t, f) = [|X(t, f-N)|, \cdots, |X(t, f)|, \cdots, |X(t, f+N)|, G_{full}(|(X(t, f)|)]^T \in \mathbb{R}^{2N+2}. \tag{4}$$

Finally for each frequency, the sub-band model G_{sub} takes each input sequence and predicts a complex Ideal Ratio Mask (cIRM) sequence which distinguishes clean speech with respect to each frequency.

The full-band and sub-band models have basically the same structure with two unidirectional LSTM layers and one fully connected layer. The difference is that full-band model has 512 hidden units in each LSTM layers and uses ReLU activation after the fully connected layer, whereas sub-band model has 384 hidden units in each LSTM layers and uses no activation. Even though there are F different frequencies, there is only one unique sub-band model G_{sub}.

3.2 DEMUCS

DEMUCS utilizes convolutional encoder-decoder architecture with U-net skip connections [8] with a recurrent neural network in between the last encoder layer and the first decoder layer to capture sequential information of audio. Given noisy audio $\mathbf{x} \in \mathbb{R}^T$ where x is corrupted from $\mathbf{y} \in \mathbb{R}^T$ using an additive noise $\mathbf{n} \in \mathbb{R}^T$, or $\mathbf{x} = \mathbf{y} + \mathbf{n}$, the goal of the model is to find f such that $f(\mathbf{x}) \approx \mathbf{y}$.

Both the encoder and the decoder architectures consist of two convolution layers. The encoder's first convolution extracts representation of the input or the output of the previous layer and the second doubles the number of channels using a 1x1 convolution, or channel-wise pooling, and the output is used as input into the next encoder/LSTM layer and the corresponding decoder layer following U-net architecture. ReLU activation and GLU activation are used respectively. The decoder's architecture is an inverse operation of the encoder, where the first is the 1x1 convolution followed by GLU and the second is a transposed convolution. The decoder takes as input output from the previous decoder or LSTM layer and the output from the connected encoder to create a waveform of enhanced audio, or $f(\mathbf{x})$. The LSTM network R captures sequential information, or a non-linear transformation, of the latent representation $\mathbf{z}$ from the last encoder layer, denoted $\hat{\mathbf{z}} = R(\mathbf{z}) = LSTM(\mathbf{z}) + \mathbf{z}$, which goes into the first decoder layer.

4 Dataset

We will use the dataset from the github repository of DNS Challenge 2020. There is an explanation for the dataset in [6].

For training data, we will get the noisy speech dataset by synthesizing clean speech and noise. The clean speech dataset comes from the public audiobooks dataset called Librivox. It has recordings over 10,000 public domain audiobooks by 11,350 speakers where majority of them are in English. Many of these recordings have excellent speech quality, but still some are of poor speech quality with speech distortion, background noise and reverberation. Thus, DNS Challenge 2020 has filtered the clean speech data based on the speech quality, which is measured by the Mean Opinnion Score (MOS), and chose only the upper quartile with respect to MOS as the clean speech dataset. The resulting dataset has 500 hours of speech from 2150 speakers and all the filtered clips are split into segments of 10 seconds.

The noise clips were selected from Audioset and Freesound. Audioset is a collection of about 2 million human-labeld 10 second sound clips. They are drawn from YouTube videos and belong to about 600 audio events. Since certain audio event classes such as music and speech are overrepresented in the Audioset, DNS Challenge 2020 has tried to balance the dataset by sampling so that each class audio event class has at least 500 clips. Also, DNS Challenge 2020 has removed the clips with any kind of speech activity since speech-like noise can bring out the suppression of speech while the trained model tries to suppress speech-like noise. The resulting noise dataset has about 150 audio classes and 60,000 clips. An additional 10,000 noise clips from Freesound are also augmented to the dataset.

The noisy speech dataset is created by adding clean speech and noise at various SNR levels. (Synthesizing the clean speech and noise : same method as the paper? Or new method?)

Although what we observe in the real world is not perfectly expressed by the synthetic dataset, we take advantage of the synthetic dataset since most of the speech enhancement (SE) models require a clean reference for utilizing objective metrics such as PESQ or STOI.

For the test set, we can use synthetic test dataset which adds the clean speech from the Graz University's clean speech dataset and noise clips from the Audioset and Freesound, which are not present in the training set. Since these synthetic clips come with ground truth references which are the clean speech data, we can evaluate the method using objective metrics such as PESQ and STOI. Whereas for the test set, or 'blind' test set, there is no ground truch references provided. We can use this set for the final evaluation using subjective metrics.

Here, for our problem, we can write $x = y + n$ where y is the clean speech, n is the noise, and the x is the synthesized noisy speech. Simply put, x is the input for our model and the y is the desired output of our model. If we denote $\hat{y}$ as the actual output of our model and $y^{\star}$ as the output from the Microsoft Teams platform, then we want for the divergence of $\hat{y}$ and y to be smaller than that of $y^{\star}$ and y. In this way, we can measure how well our noise suppression model works compared to Microsoft Teams.

5 Evaluation Metric

Evaluation Metric List	
Metric	Mathematical Expression
STOI	$STOI = \frac{1}{JM}\sum_{j,m} d_{j,m}$ • J : the number of one-third octave bands (frequency) • M : the total number of frames (time) • $d_{j,m}$: sample correlation between $\mathbf{x}_{j,m}$ and $\overline{\mathbf{y}}_{j,m}$ • $\mathbf{x}_{j,m}$: the short-time temporal envelope of the clean speech • $\overline{\mathbf{y}}_{j,m}$: the short-time temporal envelop of the (normalized and clipped) degraded speech
PESQ	$PESQ = a_0 + a_1 D_{ind} + a_2 A_{ind}$ • D_{ind} : average disturbance • A_{ind} : asymmetric average disturbance • a_0, a_1, a_2 : a_0 is set as 4.5 and $a_1, a_2 < 0$ can be modified for the given task
fwSegSNR	$fwSegSNR = \frac{10}{M}\sum_{m=1}^{M-1}\frac{\sum_{j=1}^{K} W(j,m)\log_{10}\frac{\lvert X(j,m)\rvert^2}{(\lvert X(j,m)\rvert-\lvert\hat{X}(j,m)\rvert)^2}}{\sum_{j=1}^{K} W(j,m)}$ • K : the number of bands • M : total number of frames • $\lvert X\rvert$: weighted (by a Gaussian shaped window) clean signal spectrum • W : weight on j-th frequency and m-th frame, which is exponential of $\lvert X\rvert$

STOI([9]) stands for a short-time objective intelligence measure. Essentially it measures the average of every correlation coefficient between each of short-time segments of DFT(discrete Fourier transform)-based band for clean speech and the output of noise reduction from the noisy speech input.

Perceptual Evaluation of Speech Quality (PESQ)([7]) is a metric used for obtaining Mean Opinion Scores (MOS) of speech in an audio. This objective metric, ranging from -0.5 to 4.5, measures the quality of an audio based on a sum of disturbances in the audio, and while it is not intended for speech enhanced by noise suppression, we follow [1], our baseline model, to compare our model's performance.

fwSegSNR([5]), Frequency weighted Segmental Signal-to-Noise Ratio, is a generalized short time performance measure. It gives each frequency's SNR two kinds of weights. First, static frequency weighting that is derived from known psycho-acoustic properties of hearing. Second, dynamic frequency weighting which is related to the speech production mechanism.

6 Loss Function

For this section, we shall use the notations below.

- Time domain

  - $\mathbf{y} \in \mathbb{R}^T$: a clean speech waveform
  - $\mathbf{n} \in \mathbb{R}^T$: a noise waveform
  - $\mathbf{x} = \mathbf{y} + \mathbf{n}$: a noisy speech waveform
- Time-Frequency domain
  - $\mathbf{Y} \in \mathbb{C}^{T \times F}$: a complex spectrogram for clean speech
  - $\mathbf{N} \in \mathbb{C}^{T \times F}$: a complex spectrogram for noise
  - $\mathbf{X} = \mathbf{Y} + \mathbf{N}$: a complex spectrogram for noisy speech
- Speech-Enhancement model
  - A speech enhancement model G maps $\mathbf{x} \in \mathbb{R}^T$ to $\hat{\mathbf{y}} \in \mathbb{R}^T$ where $\hat{\mathbf{y}}$ is a waveform of enhanced speech.

6.1 Loss function for Demucs

Given $\mathbf{y}$ and $\hat{\mathbf{y}}$, the loss $\mathcal{L}_{Demucs}$ can be expressed as

$$\mathcal{L}_{Demucs} = \mathcal{L}_{Wave} + \lambda \mathcal{L}_{Spectrogram}$$

Here, $\mathcal{L}_{Wave}$ is a waveform loss

$$\mathcal{L}_{Wave}(\mathbf{y}, \hat{\mathbf{y}}) = \|\mathbf{y} - \hat{\mathbf{y}}\|_1$$

and $\mathcal{L}_{Spectrogram}$ is a multi-resolution STFT loss

$$\mathcal{L}_{Spectrogram}(\mathbf{y}, \hat{\mathbf{y}}) = \mathcal{L}_{sc}(\mathbf{y}, \hat{\mathbf{y}}) + \mathcal{L}_{mag}(\mathbf{y}, \hat{\mathbf{y}})$$

where the spectral convergence loss and magnitude loss are given as

$$\mathcal{L}_{sc}(\mathbf{y}, \hat{\mathbf{y}}) = \frac{\left\| |\mathbf{Y}| - |\hat{\mathbf{Y}}| \right\|_F}{\| |\mathbf{Y}| \|_F} \quad \text{and} \quad \mathcal{L}_{mag}(\mathbf{y}, \hat{\mathbf{y}}) = \frac{1}{T} \left\| \log |\mathbf{Y}| - \log |\hat{\mathbf{Y}}| \right\|_1$$

6.2 Loss function for FullSubNet

Given noisy waveform $\mathbf{x}$ and clean waveform $\mathbf{y}$, we can get complex spectrogram $\mathbf{X}$ and $\mathbf{Y}$ respectively. FullSubNet takes noisy spectrogram $\mathbf{X}$ as an input and the output is the estimated complex Ideal Ratio Mask(cIRM) $\hat{\mathbf{M}} \in \mathbb{C}^{T \times F}$ while the target output is complex Ideal Ratio Mask $\mathbf{M} \in \mathbb{C}^{T \times F}$ such that $\mathbf{Y} = \mathbf{M} \odot \mathbf{X}$ where $\odot$ is an element-wise multiplication. FullSubNet uses the mean squared error between $\mathbf{M}$ and $\hat{\mathbf{M}}$ as the loss $\mathcal{L}_{cIRM}$.

6.3 Adding auxiliary loss

For the objective function of both of enhancement models, Demucs and FullSubNet, an auxiliary loss regarding acoustic parameters can be added to improve their perceptual quality and intelligibility.

$$\mathcal{L}_{new} = \mathcal{L}_{original} + \gamma \mathcal{L}_{acoustic}$$

For $\mathcal{L}_{acoustic}$, [11] suggests $\mathcal{L}_{eGeMAPS}$ while [12] proposes $\mathcal{L}_{TAP}$.

$\mathcal{L}_{eGeMAPS}$ is designed to narrow the difference between the eGeMAPS functionals outputs (which are $D = 88$ dimensional vectors) of the clean speech and enhanced speech. Given clean waveform $\mathbf{y}$ and enhanced waveform $\hat{\mathbf{y}}$, the eGeMAPS loss is represented as the following.

$$\mathcal{L}_{eGeMAPS} = \|\mathbf{e} - \hat{\mathbf{e}}\|_2^2 = \left\| g(\mathbf{y}) - h_\phi(\hat{\mathbf{Y}}) \right\|_2^2$$

Here, g is original non-differentiable function to calculate eGeMAPS and h_ϕ is a neural network parameterized by weights ϕ, which is an estimator for eGeMAPS functionals.

On the other hand, $\mathcal{L}_{TAP}$ is designed to narrow the difference between the outputs of 25 temporal acoustic parameters with T discrete time frames of the clean speech and enhanced speech. Given clean waveform $\mathbf{y}$ and enhanced waveform $\mathbf{y}$, the TAP loss is expressed as the following.

$$\mathcal{L}_{TAP} = \text{MAE}(A_\psi(\mathbf{y}) \odot \sigma(\boldsymbol{\omega}), A_\psi(\hat{\mathbf{y}}) \odot \sigma(\hat{\boldsymbol{\omega}}))$$

Here, A_ψ takes a signal input and outputs the temporal acoustic parameter estimates using a recurrent neural network parameterized by ψ. Also, $\boldsymbol{\omega}$ expresses the frame energy weights, which is derived from the Parseval's identity ; $\mathbf{w} = \frac{1}{F}\sum_f |\mathbf{Y}(f)|^2$ where $\mathbf{Y}(f) \in \mathbb{C}^T$ for each $f = 1, \cdots, F$ and $\mathbf{Y} = (\mathbf{Y}(1), \cdots, \mathbf{Y}(F))$. Note that σ denotes the sigmoid function.

We aim to fine-tune two pre-trained models (Demucs and FullSubNet) under the supervision of loss that we have just mentioned in order to get enhanced speech with high quality in terms of perception and intelligibility. During backward propagation of this fine-tuning process, the weight parameters ϕ and ψ of estimator networks for eGeMAPS functionals and TAP are trained prior to the fine-tuning and stays frozen while the enhancement model weight parameters are optimized.

7 Baseline

Microsoft Teams is a business communication platform developed by Microsoft, of which usage share keeps increasing so that its number of users reaches 270million in 2022 (Source). It mainly provides lots of features that support live speech in meetings, calls, and so on. Hence, the need for speech enhancement is inevitable. So we aim to improve it more and adopt Microsoft Teams as the baseline of our model. Although papers from Microsoft including [1] suggest Microsoft's research into an extension of the Demucs architecture, the exact model used in Microsoft Teams is not publicly available due to the confidentiality of their research. However, we can still set our model's baseline as the model used in Microsoft Teams by obtaining the suppressed audio by inputting the audio through their platform. To minimize the noise that could come from recording audio through a microphone, we inject the corrupted audio through a virtual microphone to simulate the inference stage of inputting audio files into Microsoft's model.

8 Analysis of Current Findings

Analysis is critical to understanding the state-of-art of the present models and system. We employ the dataset from DNS Challenge 2020 [6], specifically the test set that consists of 150 clean and synthetically noisy audio pairs. The noisy data is as input to the FullSubNet and Demucs models for our analysis of enhancement.

Table 1: Statistics of Evaluation Metrics of Noisy and Enhanced Audio

Metric	Statistics	Noisy	DEMUCS	FullSubNet
PESQ	mean	1.5822	2.6447	2.8885
PESQ	std	0.4575	0.6332	0.6719
STOI	mean	0.9152	0.9652	0.9641
STOI	std	0.0646	0.0328	0.0349
fwSNRSeg	mean	12.6233	17.1348	16.9617
fwSNRSeg	std	4.9981	3.7549	4.3833

Table 1 shows the mean and standard deviation of PESQ, STOI and fwSNRSeg over 150 audio signals of each audio set. We calculate these metrics between the clean audio set and each of noisy set, enhanced set using DEMUCS, and enhanced set using FullSubNet. We find according to the statistics that both DEMUCS and FullSubNet improves the noisy audio as all three metrics increase for the outputs of DEMUCS and FullSubNet. PESQ value represents the perceptual evaluation of speech quality and optimal value for the same is 4.5 and the PESQ value nearer to 4.5 implies better audio quality. It is observed that the mean PESQ values of both Demucs and FullSubNet are significantly higher than the mean PESQ value of noisy data.

Similarly, we observe that statistics for STOI and fwSNRSeg give similar results. STOI denotes correlation with the clean speech and higher STOI implies output is more correlated to the clean speech suggesting improvement over noisy set. In the results we observe that FullSubNet and Demucs have higher mean values for STOI over noisy. fwSegSNR denotes ratio of audio signal over noise, and higher fgSegSNR shows that the level of noise is lower. As anticipated, Demucs and FullSubNet have higher mean fwSegSNR values compared to the mean fwSegSNR value for the noisy data.

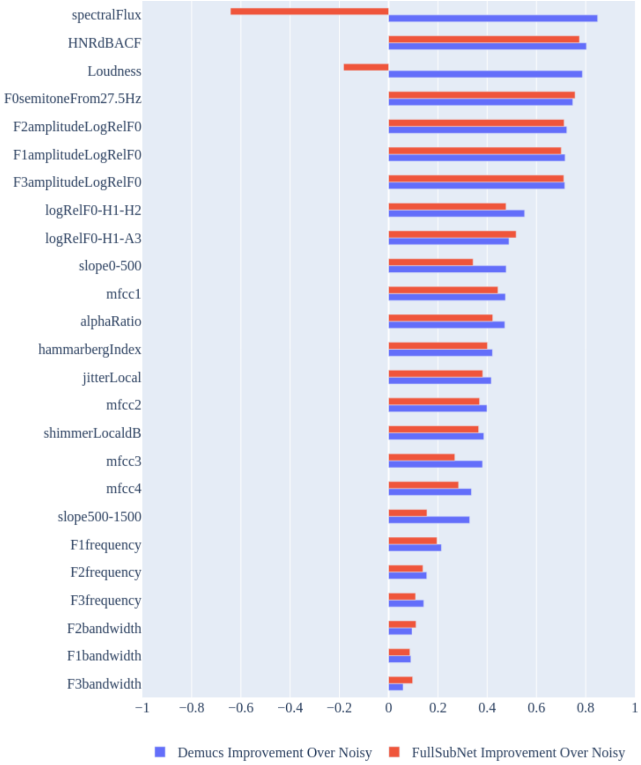

Figure 1: Graph of improvements by DEMUCS and FullSubNet over noisy audio on 25 temporal acoustic parameters. The improvement is a percentage difference between the MAE of parameters for the noisy set and the MAE of each model's enhanced audio set.

eGeMAPS suggest that traditional evaluation metrics might not be sufficient to gauge the overall perceptual quality of the speech. Thus, 25 temporal acoustic parameters, or low-level descriptors, are introduced to measure the overall perceptual quality of the speech. In the Figure 1 , we analyze all parameters over small windows of audio and calculate the Mean Absolute Error (MAE) of each eGeMAPS parameter between the models' output and the original clean speech. We perform the same calculation for noisy data as well and obtain the improvement of enhancement models as percentage difference between the MAE from noisy data and the MAE from each model output.

We expect this value to be positive as we anticipate MAE for noisy data to be higher than the MAE calculated by respective model output and from the figure we can see the results are mostly as expected. However, for features such as spectralFlux and Loudness, the error between output of FullSubNet and the clean speech was larger than the error between noisy and clean audio.

Figure 2 shows improvement by DEMUCS and FullSubNet in different statistics of temporal acoustic parameters, or the functionals. Although Figure 1 shows the percentage acoustic improvements in low-level descriptors, improvements in the functionals would convey that the overall statistics of clean speech and enhanced speech are similar, implying better performance. Although Figure 2 shows that most statistics have improved from the noisy set, the enhanced set from both DEMUCS and FullSubNet were significantly different in some some of the functionals such as F1 bandwidth or loudness.

While the evaluation metrics from Table 1 show that both DEMUCS and FullSubNet significantly improves the quality of synthetically noisy audio, the percentage audio improvement plots for the low-level descriptors and the functionals suggest that the model outputs fall behind in some of the parameters. We can infer from these results that improving these parameters to be closer to the original clean speech would provide cleaner audio signal as well as better perceptual quality of the audio.

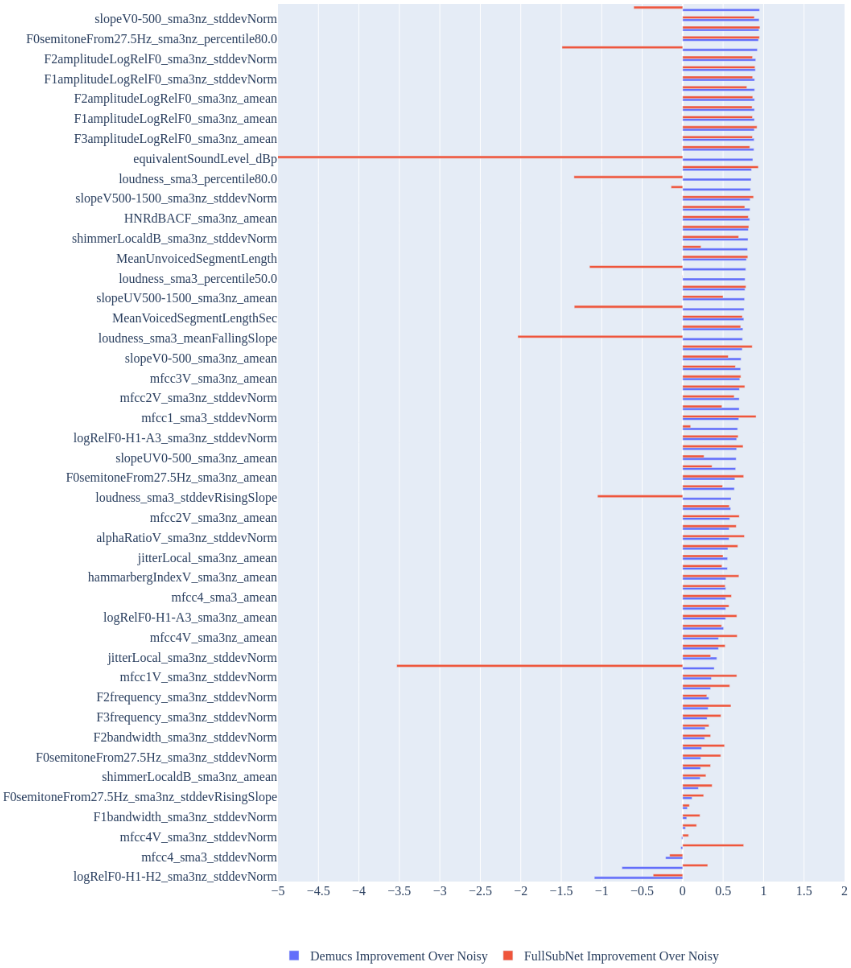

Figure 2: Graph of improvements by DEMUCS and FullSubNet over noisy audio on different statistics of temporal acoustic parameters. These statistics, referred to as the functionals, include mean, standard deviation, and percentiles of the eGeMAPS parameters. The improvement is derived using the same calculation as Figure 1.

References

[1] Sebastian Braun, Hannes Gamper, Chandan K. A. Reddy, and Ivan Tashev. Towards efficient models for real-time deep noise suppression. 2021.

[2] Alexandre Defossez, Gabriel Synnaeve, and Yossi Adi. Real time speech enhancement in the waveform domain. 2020.

[3] Florian Eyben, Klaus R. Scherer, Björn W. Schuller, Johan Sundberg, Elisabeth André, Carlos Busso, Laurence Y. Devillers, Julien Epps, Petri Laukka, Shrikanth S. Narayanan, and Khiet P. Truong. The geneva minimalistic acoustic parameter set (gemaps) for voice research and affective computing. *IEEE Transactions on Affective Computing*, 7(2):190–202, 2016.

[4] Xiang Hao, Xiangdong Su, Radu Horaud, and Xiaofei Li. Fullsubnet: A full-band and sub-band fusion model for real-time single-channel speech enhancement. 2020.

[5] Yi Hu and Philipos C. Loizou. Evaluation of objective quality measures for speech enhancement. *IEEE Transactions on Audio, Speech, and Language Processing*, 16(1):229–238, 2008.

[6] Chandan K. A. Reddy, Vishak Gopal, Ross Cutler, Ebrahim Beyrami, Roger Cheng, Harishchandra Dubey, Sergiy Matusevych, Robert Aichner, Ashkan Aazami, Sebastian Braun, Puneet

Rana, Sriram Srinivasan, and Johannes Gehrke. The interspeech 2020 deep noise suppression challenge: Datasets, subjective testing framework, and challenge results. 2020.

[7] A.W. Rix, J.G. Beerends, M.P. Hollier, and A.P. Hekstra. Perceptual evaluation of speech quality (pesq)-a new method for speech quality assessment of telephone networks and codecs. In *2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No.01CH37221)*, volume 2, pages 749–752 vol.2, 2001.

[8] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. 2015.

[9] Cees H. Taal, Richard C. Hendriks, Richard Heusdens, and Jesper Jensen. An algorithm for intelligibility prediction of time–frequency weighted noisy speech. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(7):2125–2136, 2011.

[10] Donald S. Williamson, Yuxuan Wang, and DeLiang Wang. Complex ratio masking for monaural speech separation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 24(3):483–492, 2016.

[11] Muqiao Yang, Joseph Konan, David Bick, Anurag Kumar, Shinji Watanabe, and Bhiksha Raj. Improving speech enhancement through fine-grained speech characteristics. 2022.

[12] Yunyang Zeng, Joseph Konan, Shuo Han, David Bick, Muqiao Yang, Anurag Kumar, Shinji Watanabe, and Bhiksha Rag. Taploss : A temporal acoustic parameter loss for speech enhancement (preprint).