T-GSA

Transformer with Gaussian-weighted self-attention for speech enhancement

Jaeyoung Kim, Mostafa El-Khamy, Jungwon Lee

ICASSP 2020

Introduction

- Transformer Neural Networks(TNN) usually has good performance on many tasks.

- Transformer Neural Networks(TNN) usually has good performance on many tasks.

  But this is not the case in speech enhancement.

	PESQ					
Methods\SNR(dB)	**-10**	**-5**	**0**	**5**	**10**	**15**
CNN-LSTM	1.43	1.65	1.89	2.16	2.35	2.54
O-T	1.29	1.45	1.63	1.87	2.07	2.29

PESQ

Methods\SNR(dB)	**-10**	**-5**	**0**	**5**	**10**	**15**
CNN-LSTM	1.43	1.65	1.89	2.16	2.35	2.54
O-T	1.29	1.45	1.63	1.87	2.07	2.29

PESQ

Methods\SNR(dB)	**-10**	**-5**	**0**	**5**	**10**	**15**
CNN-LSTM	1.43	1.65	1.89	2.16	2.35	2.54
O-T	1.29	1.45	1.63	1.87	2.07	2.29
T-AB	**1.49**	**1.67**	**1.85**	**2.01**	**2.28**	**2.50**

PESQ

Methods\SNR(dB)	**-10**	**-5**	**0**	**5**	**10**	**15**
CNN-LSTM	1.43	1.65	1.89	2.16	2.35	2.54
O-T	1.29	1.45	1.63	1.87	2.07	2.29
T-AB	**1.49**	**1.67**	**1.85**	**2.01**	**2.28**	**2.50**

- Appropriate position information can bring performance improvements.
- The author proposed Gaussian Weighted Self Attention and applied it to Real & Complex Transformer.

Proposed Architectures

Gaussian Weighted Self Attention

$$P_l = \begin{bmatrix} p^l_{1,1} & p^l_{1,2} & \cdots & p^l_{1,T} \\ p^l_{2,1} & p^l_{2,2} & \cdots & p^l_{1,T} \\ & \vdots & & \\ p^l_{T,1} & p^l_{T,2} & \cdots & p^l_{T,T} \end{bmatrix}$$

$p^l_{i,j} = -\frac{(i-j)^2}{\sigma_l^2}$, σ_l is a trainable parameter.

Gaussian Weighted Self Attention (cont.)

- Usage in Self-Attentional Acoustic Models (as bias)

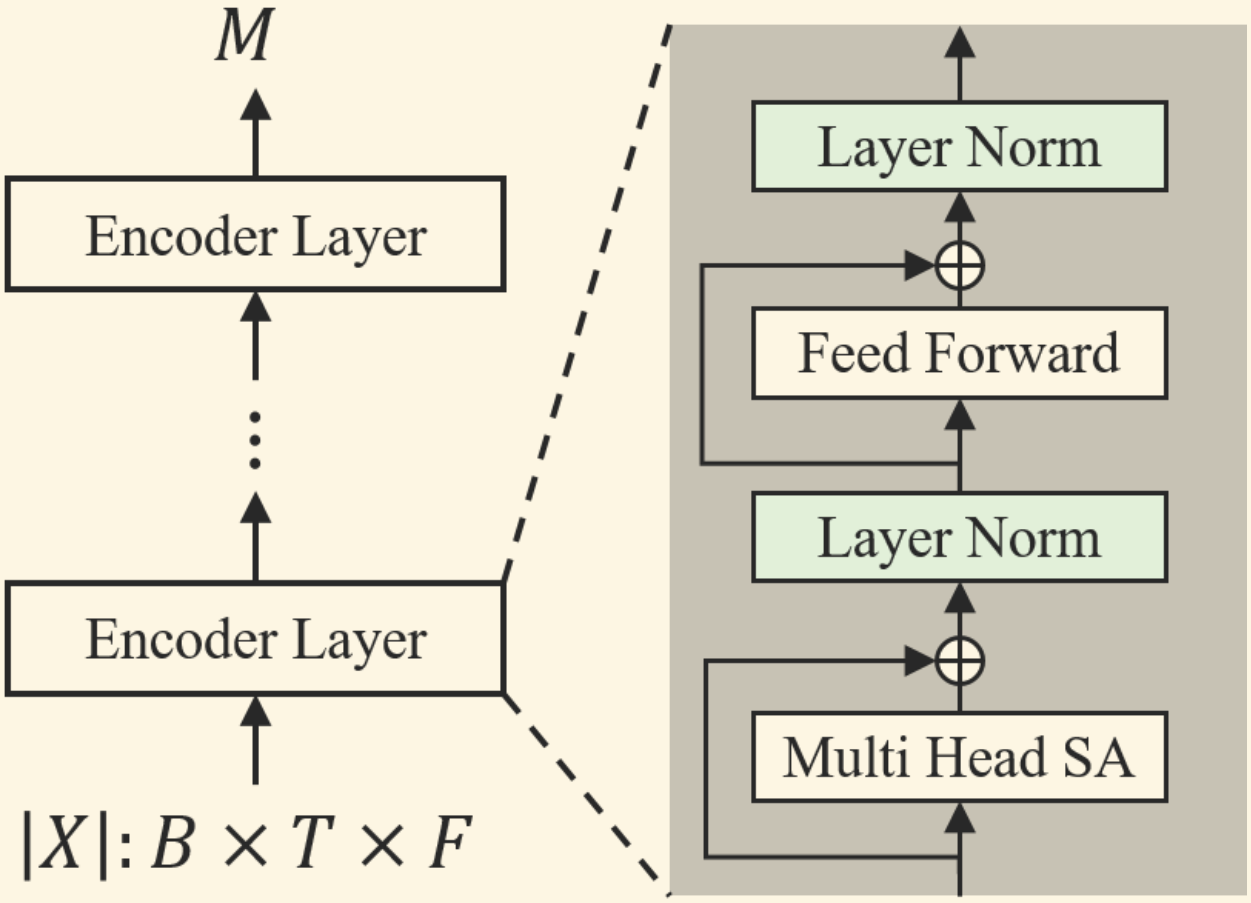
$$Softmax(\frac{Q_l(K_l)^T}{\sqrt{d}} + P_l)V_l$$

- **Usage in this paper** (as **weighted**)

$$Softmax(\frac{Q_l(K_l)^T}{\sqrt{d}} \odot e^{P_l})V_l$$

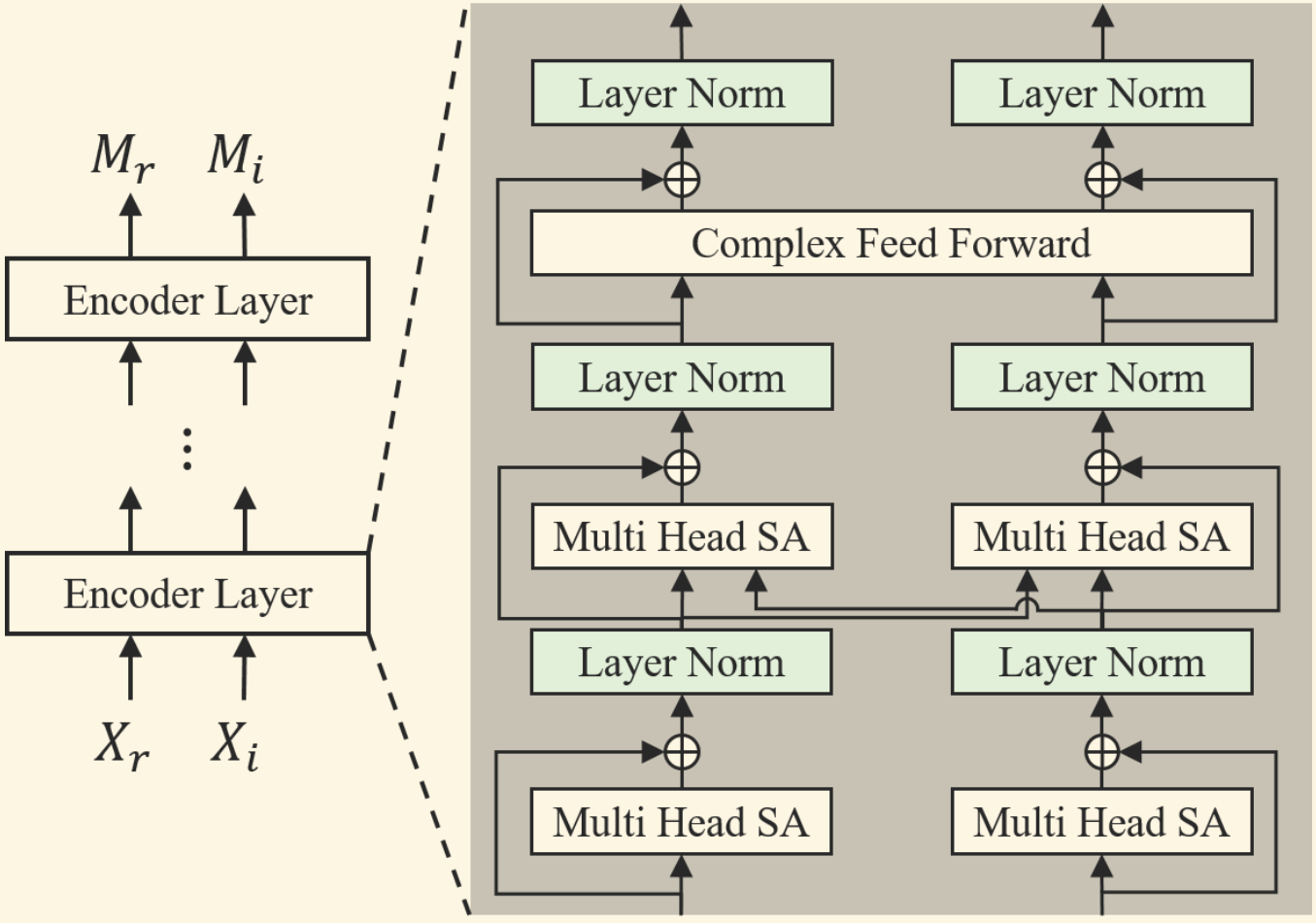
Model

Real Transformer



Model (cont.)

Complex Transformer



Output

- Real Mask

$$|\hat{S}| = M \odot |X|$$
$$\hat{S} = |\hat{S}|e^{\angle X}$$

- Complex Mask

$$\hat{S}_r = M_r \odot X_r - M_i \odot X_i$$
$$\hat{S}_i = M_r \odot X_i + M_i \odot X_r$$

$$\hat{s} = Griffin\text{-}Lim\ Algorithm(\hat{S})$$

Output (cont.)

Griffin-Lim Algorithm

$Input:\ complex\ spectrogram\ A_0 e^{j\Omega_0}$

for all $n\ from\ 1\ to\ N$ **do**

$$A_n e^{j\Omega_n} \leftarrow stft(stft^{-1}(A_0 e^{j\Omega_{n-1}}))$$

end for

$return\ stft^{-1}(A_0 e^{j\Omega_N})$

Experiments

Dataset

QUT-NOISE-TIMIT

- Number of Noise Types: 5
- Train
  - SNR: -5、5 dB
  - Total Length: 25 hours
- Test
  - SNR: -10~15 dB
  - Total Length: 12 hours

The detailed data selection is described here.

Dataset (cont.)

VoiceBank-DEMAND

- Train
  - SNR: 15、10、5 and 0 dB
  - Number of Speakers: 28
  - Number of Noise Types: 10
- Test
  - SNR: 17.5、12.5、7.5 and 2.5 dB
  - Number of Speakers: 2
  - Number of Noise Types: 5

Hyperparameter

- Number of Encoder Layer
  - Real: 10
  - Complex: 6
- Number of Input Dim: 1024
- Loss: $L_{SDR} + \alpha L_{PESQ},\ \alpha = 3.2$

QUT-NOISE-TIMIT

PESQ

Methods\SNR(dB)	**-10**	**-5**	**0**	**5**	**10**	**15**
Noisy Input	1.07	1.08	1.13	1.26	1.44	1.72
CNN-LSTM	1.43	1.65	1.89	2.16	2.35	2.54
O-T	1.29	1.45	1.63	1.87	2.07	2.29
T-AB	1.49	1.67	1.85	2.01	2.28	2.50
T-GSA	**1.54**	**1.76**	**2.00**	**2.28**	**2.51**	**2.74**
C-T-GSA	1.43	1.64	1.88	2.17	2.40	2.67

QUT-NOISE-TIMIT (cont.)

SDR

Methods\SNR(dB)	**-10**	**-5**	**0**	**5**	**10**	**15**
Noisy Input	-11.82	-7.33	-3.27	0.21	2.55	5.03
CNN-LSTM	-2.31	1.80	4.36	6.51	7.79	9.65
O-T	-3.25	0.92	3.39	5.35	6.39	8.10
T-AB	-2.80	1.18	3.67	5.67	6.78	8.18
T-GSA	-1.66	2.35	4.95	7.10	8.40	10.36
C-T-GSA	**-1.57**	**2.51**	**5.03**	**7.36**	**8.58**	**10.40**

Evaluation on VoiceBank-DEMAND

Method\Metrics	CSIG	CBAK	COVL	PESQ	SSNR	SDR
Noisy Input	3.37	2.49	2.66	1.99	2.17	8.68
SEGAN	3.48	2.94	2.80	2.16	7.73	-
WAVENET	3.62	3.23	2.98	-	-	-
TF-GAN	3.80	3.12	3.14	2.53	-	-
CNN-LSTM	4.09	3.54	3.55	3.01	10.44	19.14
T-GSA	**4.18**	**3.59**	**3.62**	**3.06**	**10.78**	**19.57**

Conclusion

- Positional Weighted is better than Positional Bias in Speech Enhancement task.
- Complex NN has more advantages in SDR-related evaluation metrics.

Possible Future Improvement

- Currently, each head on the same layer shares the positional weights.
- A set of position weights consists of only one Gaussian.

- The positional weights starts to spread from i==j as the center.

- Currently, each head on the same layer shares the positional weights.
- A set of position weights consists of only one Gaussian.
  Use multiple Gaussian functions

- The positional weights starts to spread from i==j as the center.

- Currently, each head on the same layer shares the positional weights.
- A set of position weights consists of only one Gaussian.
  Use multiple Gaussian functions

- The positional weights starts to spread from i==j as the center.
  Add center offset.

Appendices

Appendix. A

Griffin-Lim Algorithm

$Input:\ complex\ spectrogram\ A_0 e^{j\Omega_0}$

for all $n\ from\ 1\ to\ N$ **do**

$$A_n e^{j\Omega_n} \leftarrow stft(stft^{-1}(A_0 e^{j\Omega_{n-1}}))$$

end for

$return\ stft^{-1}(A_0 e^{j\Omega_N})$

Appendix. A (cont.)

Clean Magnitude + Noisy Phase

VoiceBank-DEMAND Test Dataset

- Window Size: 511
- Hop Length: 63
- Window Function: Hann Window

Appendix. A (cont.)

Clean Magnitude + Noisy Phase

iter N	PESQ	STOI	SISDR
0	3.956	0.989	20.15
1	4.191	0.993	20.91
2	4.280	0.994	21.30
3	4.336	0.995	21.55
4	4.364	0.996	21.73

Appendix. B

The reason why the PESQ of C-T-GSA is **lower** than that of T-GSA

Author's guess:

「Difficulty in predicting the phase spectrum」or
「Overfitting due to the larger parameter size」

Appendix. B

The reason why the PESQ of C-T-GSA is **lower** than that of T-GSA

Author's guess:

「Difficulty in predicting the phase spectrum」or
「Overfitting due to the larger parameter size」

Really?

Appendix. B (cont.)

w/o Output Phase

Methods\Metrics	SISDR	PESQ	STOI
SISDR (UNet)	18.57	2.59	0.935
SISDR (Complex UNet)	18.67	2.43	0.934
PMSQE (UNet)	13.07	3.05	0.930
PMSQE (Complex UNet)	13.55	3.00	0.931

Parameter Size: 3.1M (Complex UNet == UNet)