Wei ZHAO, Li XU, 2022. Efficient decoding self-attention for end-to-end speech synthesis. *Frontiers of Information Technology & Electronic Engineering*, 23(7):1127-1138. https://doi.org/10.1631/FITEE.2100501

Efficient decoding self-attention for end-to-end speech synthesis

Key words: Efficient decoding; End-to-end; Self-attention; Speech synthesis

Corresponding author: Li XU
E-mail: xupower@zju.edu.cn
ORCID: https://orcid.org/0000-0001-7874-629X

Motivation

1. Self-attention has been extensively used in text-to-speech (TTS) because of its parallel structure and superior strength in modeling sequential data.

2. The inference procedure becomes relatively slow when employing a self-attention decoder for end-to-end speech synthesis since the number of operations required by self-attention scales quadratically in sequence length.

3. The original self-attention may be substituted by a more efficient alternative with only linear computation complexity.

Main idea

1. The self-attention in the TTS model’s decoder is replaced by the stack of a global average attention (GAA) module and a local predictive attention (LPA) module.

2. The GAA module assigns the same average attention weight to each token to build stable global dependencies. The LPA module combines lightweight and dynamic convolution to construct flexible local dependencies.

3. The modified decoder can be accelerated via dynamic programming to have only linear computation complexity in inference.

Method

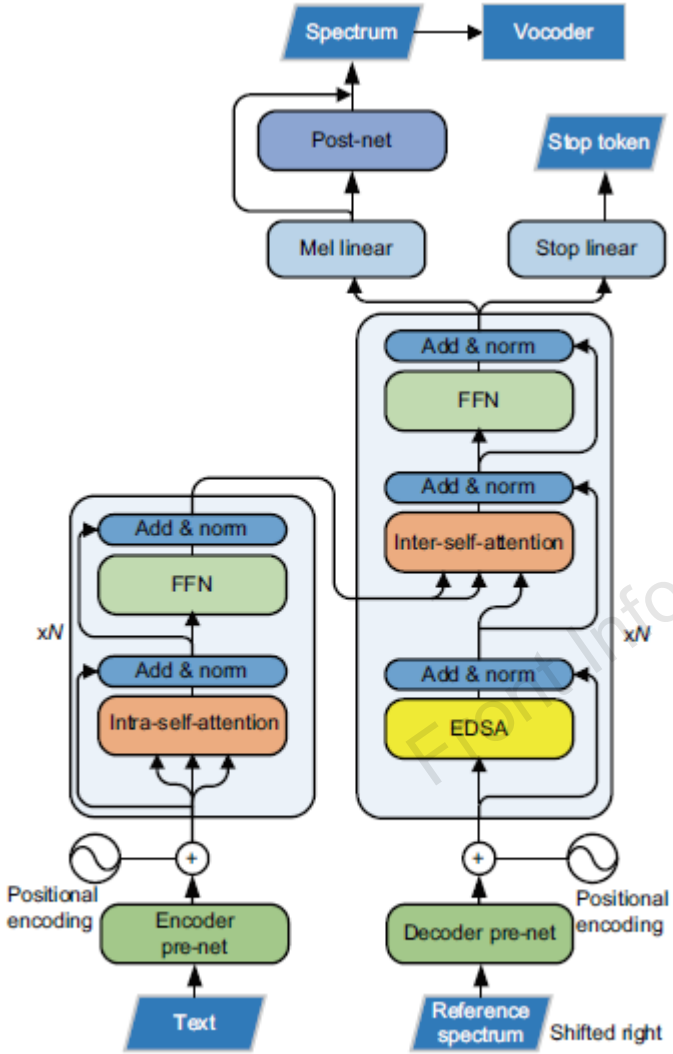

Fig. 3 Transformer-based TTS system with the efficient decoding self-attention (EDSA) module

The overall framework of our speech synthesis model and the structure of the proposed efficient decoding self-attention (EDSA):

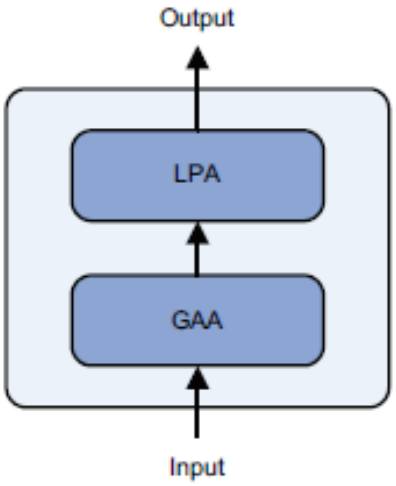

Fig. 4 Structure of the EDSA module

Method (Cont'd)

The GAA module:

$$\begin{pmatrix} v_1' \\ v_2' \\ v_3' \\ \vdots \\ v_n' \end{pmatrix} = \begin{pmatrix} v_1 \\ (v_1+v_2)/2 \\ (v_1+v_2+v_3)/3 \\ \vdots \\ (v_1+v_2+...+v_n)/n \end{pmatrix}$$

$$= \begin{pmatrix} 1 & 0 & 0 & ... & 0 \\ 1/2 & 1/2 & 0 & ... & 0 \\ 1/3 & 1/3 & 1/3 & ... & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ 1/n & 1/n & 1/n & ... & 1/n \end{pmatrix} \times \begin{pmatrix} v_1 \\ v_2 \\ v_3 \\ \vdots \\ v_n \end{pmatrix}, \quad (6)$$

Method (Cont'd)

The LPA module:

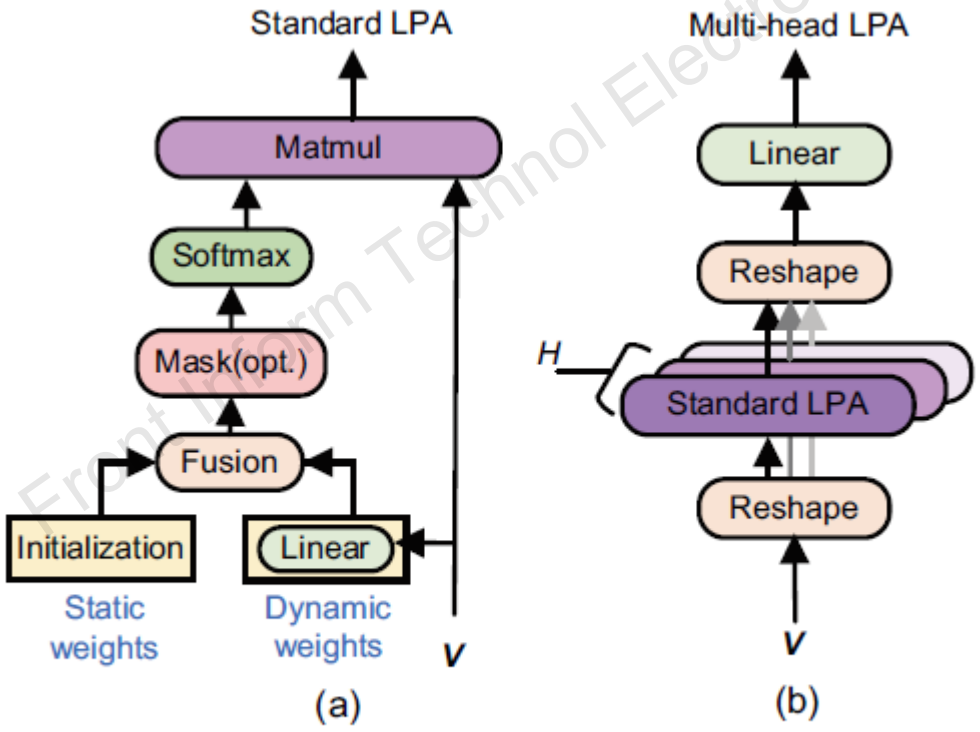

Fig. 5 Standard (a) and multi-head (b) local predictive attention networks

Method (Cont'd)

Paired with the EDSA module, a dynamic programming decoding algorithm is proposed to accelerate the inference:

$$\boldsymbol{a}_t = \boldsymbol{a}_{t-1} + \boldsymbol{v}_t, \tag{10}$$

$$\tilde{\boldsymbol{w}}_t,\ \tilde{\boldsymbol{g}}_t = \mathrm{Linear}(\boldsymbol{a}_t / t), \tag{11}$$

$$\boldsymbol{w}_t = \mathrm{Sigmoid}(\tilde{\boldsymbol{g}}_t) \cdot \tilde{\boldsymbol{w}}_t + \overline{\boldsymbol{w}}, \tag{12}$$

$$\boldsymbol{o}_t = \sum_{\boldsymbol{v}_i \in B_t} \mathrm{Dropout}(\mathrm{Softmax}(\boldsymbol{w}_t))_i \cdot \boldsymbol{v}_i, \tag{13}$$

Major results

Table 1 Mean opinion score (MOS) and Mel cepstrum distortion (MCD) with 95% confidence intervals

Model	MOS		MCD	
	CSMSC	LJSpeech	CSMSC	LJSpeech
Ground truth	4.48±0.06	4.51±0.05	N/A	N/A
Tacotron 2	4.26±0.07	4.15±0.07	6.78±0.09	7.62±0.11
Transformer	**4.31±0.07**	**4.20±0.08**	6.73±0.09	**7.40±0.10**
Ours	4.28±0.08	4.19±0.07	**6.67±0.08**	7.43±0.10

Bold values indicate the better results of the Transformer baseline and our model. N/A: not applicable

Table 2 Comparison mean opinion score (CMOS) with 95% confidence intervals

Model	CMOS	
	CSMSC	LJSpeech
EDSA	0	0
Without GAA	−0.14±0.17	−0.11±0.16
Without LPA	−0.18±0.16	−0.23±0.15

GAA: global average attention; LPA: local predictive attention

Major results (Cont'd)

Table 3 Real-time factors (RTFs) of different models with 95% confidence intervals

Model	RTF with about 400 frames		RTF with about 800 frames	
	CPU	GPU	CPU	GPU
Tacotron 2	4.044±0.015	15.702±0.079	3.926±0.023	16.051±0.217
Transformer	0.248±0.012	2.186±0.023	0.164±0.006	1.776±0.056
Ours	**1.370±0.010**	**2.643±0.026**	**1.350±0.005**	**2.656±0.009**

Bold values indicate the better results of the Transformer baseline and our model

Table 4 Number of employed floating-point operations (FLOPs) of different models in inference

Model	Number of parameters ($\times 10^6$)	Number of FLOPs ($\times 10^{12}$)			
		400	(400, SA)	800	(800, SA)
Tacotron 2	28.158	0.010	N/A	0.021	N/A
Transformer	52.949	2.183	0.654	10.032	3.534
Ours	48.245	**0.162**	**0.003**	**0.418**	**0.006**

Bold values indicate the better results of the Transformer baseline and our model. N/A: not applicable; SA: self-attention

Conclusions

1. An efficient decoding self-attention module has been presented for end-to-end speech synthesis.

2. The proposed alternative can model both long- and short-range dependencies with the number of operations scaling linearly in the sequence length.

3. Experimental results on both Mandarin and English datasets demonstrate that the suggested method can accelerate the inference remarkably with negligible performance loss.

Wei ZHAO is currently pursuing his PhD degree in the College of Electrical Engineering, Zhejiang University, Hangzhou, China. His research interests include time series analysis, speech synthesis, and speaker recognition.

Li XU received his BS, MS, and PhD degrees from Zhejiang University, Hangzhou, China, in 1986, 1989, and 1997, respectively. Since 1989, he has been with Zhejiang University, and now he is a professor of electrical engineering. From 1997 to 1998, he was a visiting associate professor with the Department of Electrical Engineering, The Ohio State University, Columbus, OH, USA. His research interests include intelligent control and intelligent systems, intelligent transportation, and industrial automation.