

An Exploration of Length Generalization in Transformer-Based Speech Enhancement

QiQuan Zhang^{1,2}, Hongxu Zhu^{4,*}, Xinyuan Qian^{5,*}, Eliathamby Ambikairajah¹, Haizhou Li^{3,2}

¹The University of New South Wales, Australia, ²National University of Singapore, Singapore

³SDS, SRIBD, The Chinese University of Hong Kong, Shenzhen, China

⁴ASTRI, Hong Kong, ⁵University of Science and Technology Beijing, China

{zhangqiquan_hit;e.ambikairajah}@unsw.edu.au, hongxuzhu@astri.org

Abstract

The use of Transformer architectures has facilitated remarkable progress in speech enhancement. Training Transformers using substantially long speech utterances is often infeasible as self-attention suffers from quadratic complexity. It is a critical and unexplored challenge for a Transformer-based speech enhancement model to learn from short speech utterances and generalize to longer ones. In this paper, we conduct comprehensive experiments to explore the length generalization problem in speech enhancement with Transformer. Our findings first establish that position embedding provides an effective instrument to alleviate the impact of utterance length on Transformer-based speech enhancement. Specifically, we explore four different position embedding schemes to enable length generalization. The results confirm the superiority of relative position embeddings (RPEs) over absolute PE (APEs) in length generalization.¹

Index Terms: speech enhancement, Transformer, length generalization, position embedding

1. Introduction

Speech enhancement seeks to reconstruct the clean speech signal from the noisy mixture contaminated by surrounding background noise, with a wide range of applications including hearing aids, audio-video conference, and automatic speech recognition (ASR). Traditionally, many approaches leverage the assumed statistical properties of noise and speech signals to derive an estimator of clean speech [1, 2, 3]. These approaches fail to eliminate rapidly varying noise signals.

In recent years, the field of speech enhancement has been dominated by supervised learning methods that optimize deep neural networks (DNNs) to perform a non-linear mapping from the noisy speech to the designed objective [4, 5]. These methods mainly fall into waveform domain [6, 7] and spectral domain schemes. Waveform domain methods often utilize a DNN with an explicit encoder-decoder architecture to directly separate the clean waveform given a noisy waveform input. Spectral domain methods, in contrast, first transform the waveform representation into spectral representation, such as magnitude spectrum (MS) [4], complex spectrum (CS) [8], and log-power spectrum (LPS) [9]. Then, DNNs are trained to estimate the spectral feature of clean speech, or a spectral mask such as the ideal ratio mask (IRM) [10], spectral magnitude mask (SMM) [10], complex ideal ratio mask (cIRM) [11], and phase-sensitive mask (PSM) [12].

¹This work is supported by ARC Discovery Grant DP1900102479; NSFC Grant No. 62271432; Internal Project Fund from Shenzhen Research Institute of Big Data (Grant No. T00120220002); Shenzhen Science and Technology Research Fund (Fundamental Research Key Project Grant No. JCYJ20220818103001002. * Corresponding author

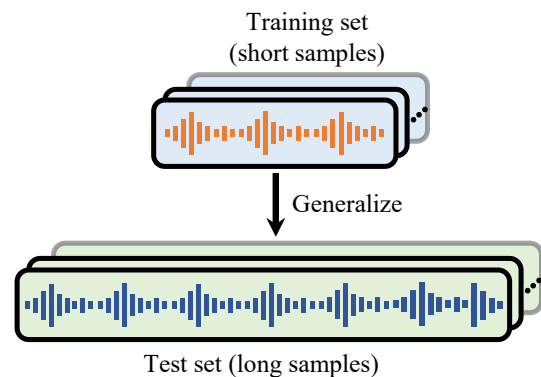


Figure 1: Length generalization in speech enhancement: Train on short speech samples and inference on long speech samples.

Numerous network architectures have been employed for speech enhancement. Multi-layer perceptrons (MLPs) are only capable of leveraging local information with a context window. With the capability to model the long-range dynamic dependencies of speech signals, long short-term memory (LSTM) networks have been exploited for speech enhancement [13, 14]. However, recurrent neural networks (RNNs) suffer from the sequential modeling nature, excluding their use in some applications. Subsequently, temporal convolution networks (TCNs) are successfully applied for speech enhancement and demonstrate impressive performance [15, 16, 17, 18, 19]. TCNs utilize stacked dilated 1-D convolution layers to build a large receptive field, which allows the model to leverage long-term contextual dependencies [20]. Transformer architectures have recently demonstrated state-of-the-art results across various signal processing tasks, including speech enhancement [21, 22, 23, 24] and image recognition [25]. As the core piece in Transformer, the multi-head self-attention (MHSA) module learns the representation of each element by capturing the interaction amongst all elements in a sequence. It grants Transformers the capability to effectively model global contextual information.

In supervised learning tasks, the ability of models to generalize to cases unseen at training time is critical. Extensive works have been conducted to study the generalization performance of neural speech enhancement models on unseen noise, speaker, and SNR conditions. Since self-attention has quadratic complexity, it is infeasible to train Transformers on all possible speech lengths. Hence, it is desirable to enable length generalization: “Can a Transformer-based speech enhancement model generalize from short speech signals to longer ones?” See Fig. 1 for an illustrative example of length generalization in speech enhancement. Position embedding appears to be a primary factor in the length generalization of Transformers. There have

been some attempts to explore position embedding to enable Transformer-based language models with the length generalization property [26, 27], such as T5 [28] and KERPLE [29]. To our knowledge, the length generalization problem in Transformer speech enhancement has not yet been explored. This paper is the first to study the pathologies that occur when Transformers are asked to generalize to enhance longer noisy inputs. We explore four advanced position embedding methods to enable length generalization for Transformer speech enhancement and show that relative position embedding (RPE) is more robust to input length change than absolute position embedding (APE).

This paper is organized as follows. We formulate the problem of neural time-frequency speech enhancement in Section 2. We present positional encoding methods in Section 3. Section 4 presents speech enhancement with position-aware Transformers. Section 5 introduces the experimental setup and analyzes the results. Section 6 concludes this paper.

2. Problem Formulation

Given the clean speech waveform $\mathbf{s} \in \mathbb{R}^{1 \times N}$ and additive noise waveform $\mathbf{d} \in \mathbb{R}^{1 \times N}$, the observed noisy waveform \mathbf{y} can be formulated as $\mathbf{y} = \mathbf{s} + \mathbf{d}$, with N denoting the number of time samples. The short-time Fourier transform (STFT) is applied to transform the raw waveform into the time-frequency representation (or spectrogram), given by $Y_{t,k} = S_{t,k} + D_{t,k}$, where $Y_{t,k}$, $S_{t,k}$, and $D_{t,k}$ respectively represents complex-valued STFT spectra of the noisy mixture, clean speech, and noise, at the time frame t and the frequency bin k . One typical neural speech enhancement scheme is to optimize a DNN model to predict a spectral mask $\widehat{M}_{t,k}$ from the noisy spectral magnitude. Then, the clean spectrum is obtained by applying the spectral mask to the noisy spectrum, given as $\widehat{S}_{t,k} = Y_{t,k} \cdot \widehat{M}_{t,k}$. In this work, we adopt the commonly used phase-sensitive mask (PSM) to carry out speech enhancement experiments for the exploration of length generalization. The PSM is formulated as [12]:

$$\text{PSM}_{t,k} = \frac{|S_{t,k}|}{|Y_{t,k}|} \cos(\Phi_{S_{t,k}-Y_{t,k}}) \quad (1)$$

where $|\cdot|$ extracts the magnitude spectrum, and $\Phi_{S_{t,k}-Y_{t,k}}$ denotes the difference between the clean and noisy spectral phase.

3. Position Embedding

There is a variety of position embedding methods explored to encode position information into Transformers, which can fall into absolute position embedding (APE) and relative position embedding (RPE).

3.1. Absolute Position Embedding

APE assigns a unique position embedding \mathbf{P}_t to each position t in a sequence. The input embeddings are summed with \mathbf{P} to form the input fed to the actual Transformers. The two common variants of APE are fixed or learnable position embedding.

Sinusoidal Position Embedding. In the original Transformer [30], the fixed sinusoidal functions of different frequencies are used to compute the position embeddings. The position embedding \mathbf{P}_t assigned to the position t is defined as

$$\mathbf{P}_{t,d} = \begin{cases} \sin\left(10000^{-\frac{d}{d_{model}}} \cdot t\right), & \text{if } d \text{ is even} \\ \cos\left(10000^{-\frac{(d-1)}{d_{model}}} \cdot t\right), & \text{if } d \text{ is odd} \end{cases} \quad (2)$$

where d_{model} denotes the dimension of input embeddings, with $d = \{1, \dots, d_{model}\}$ and $t = \{1, \dots, T\}$.

Learnable APE. Alternatively, the position embeddings can be learnable [31], where each position is assigned a fully learnable position vector. They are randomly initialized and jointly updated with the model's parameters.

3.2. Relative Position Embedding

Instead of encoding absolute position, RPE encodes the distance between each pair of elements in the input sequence. RPE commonly learned a position bias for each relative position, which is then involved in calculating the attention matrix.

T5-RPE [28] first bucketize relative distances $i-j$ between the i -th and j -th elements with a logarithmic bucket function, and the relative positions in the same bucket share the same scalar position bias across layers. A bucket of 32 parameters \mathbf{B} is learned for each attention head and the relative position bias $\mathbf{P}_{i,j}$ is assigned as

$$\mathbf{P}_{i,j} = \begin{cases} \mathbf{B}[\min(15, 8 + \lfloor \frac{\log((|i-j|)/8)}{\log(128/8)} \cdot 8 \rfloor) + 16], & i-j \leq -8 \\ \mathbf{B}[|i-j| + 16], & -8 < i-j < 0 \\ \mathbf{B}[i-j], & 0 \leq i-j < 8 \\ \mathbf{B}[\min(15, 8 + \lfloor \frac{\log((i-j)/8)}{\log(128/8)} \cdot 8 \rfloor)], & i-j \geq 8 \end{cases} \quad (3)$$

where $\lfloor \cdot \rfloor$ denotes the flooring operation.

KERPLE [29] kernelizes positional differences by using conditionally positive definite kernels. There are two variants of KERPLE, the power variant and the logarithmic variant, with the logarithmic variant showcasing a better performance:

$$\mathbf{P}_{i,j} = -r_1 \cdot \log(1 + r_2 |i-j|) \quad (4)$$

where $r_1, r_2 > 0$ denote learnable scalars in each head.

4. Speech Enhancement with Position-aware Transformer

4.1. Model Architecture

In Fig. 2, we depict the position-aware Transformer backbone network. The input to the network is the STFT magnitude spectrum of the noisy mixture, denoted by $|\mathbf{Y}| \in \mathbb{R}^{T \times K}$, where T and K respectively represent the number of time frames and frequency bands. The input embedding layer is a fully connected (FC) layer preactivated by frame-wise layer normalization (LN) followed by the ReLU function, which projects the input into a latent embedding, denoted as $\mathbf{Z} \in \mathbb{R}^{T \times d_{model}}$. In APE methods, the position embedding $\mathbf{P} \in \mathbb{R}^{T \times d_{model}}$ is added to \mathbf{Z} to

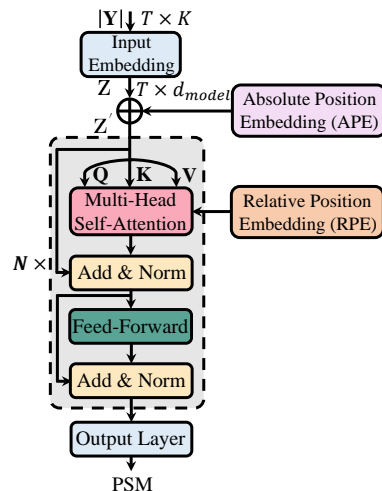


Figure 2: Diagram of position-aware Transformer model for speech enhancement. \oplus represents the element-wise addition.

form the input \mathbf{Z}' to N stacked Transformer layers. Each Transformer layer comprises two sub-layers, an MHSA module and a two-layer feedforward network (FFN). A residual connection is applied around each sub-layer, followed by LN. The output layer is an FC layer with a sigmoidal function, which generates the estimated PSM.

4.2. Position-Aware Self-Attention

Rather than incorporating position information in the input embedding (APE), RPE integrates relative position information in self-attention. The position-aware MHSA employs h attention heads to facilitate the model to jointly attend to different aspects of information. For each attention head \mathbf{Head}^i ($i \in \{1, \dots, h\}$), given a hidden representation $\mathbf{U} \in \mathbb{R}^{T \times d_{model}}$ as the input, it is firstly transformed into the queries, keys, and values with three separate project matrices ($\mathbf{W}_Q^i \in \mathbb{R}^{d_{model} \times d_k}$, $\mathbf{W}_K^i \in \mathbb{R}^{d_{model} \times d_k}$, $\mathbf{W}_V^i \in \mathbb{R}^{d_{model} \times d_v}$): $\mathbf{Q}^i = \mathbf{U}\mathbf{W}_Q^i$, $\mathbf{K}^i = \mathbf{U}\mathbf{W}_K^i$, $\mathbf{V}^i = \mathbf{U}\mathbf{W}_V^i$, with dimensions of d_k , d_k , and d_v , respectively. The attention scores are computed with the scaled dot-product in each head, where the relative position bias matrix $\mathbf{P}^i \in \mathbb{R}^{T \times T}$ is integrated into self-attention after query-key dot product:

$$\mathbf{Head}^i = \text{Softmax} \left(\frac{\mathbf{Q}^i \mathbf{K}^{i\top}}{\sqrt{d_k}} + \mathbf{P}^i \right) \mathbf{V}^i. \quad (5)$$

where $d_k = d_v = d_{model}/h$. Then, the outputs of all h attention heads are concatenated together and passed through a linear feed-forward layer, which is formulated as follows:

$$\text{MHSA}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Concat}\{\mathbf{Head}^1, \dots, \mathbf{Head}^h\} \mathbf{W}_O \quad (6)$$

where $\mathbf{W}^O \in \mathbb{R}^{d_{model} \times d_{model}}$ is the linear projection matrix. A two-layer FNN is further applied to the output of the position-aware MHSA sub-layer for two linear transformations.

5. Experiments

5.1. Datasets and Feature Extraction

The training data consists of 28 539 clean speech utterances from the LibriSpeech *train-clean-100* corpus [32], with approximately 100 hours. We collect noise recordings from the QUT-NOISE dataset [33], the Environmental Noise dataset [34], the noise recordings in MUSAN dataset [35], the RSG-10 dataset, the UrbanSound dataset [36], the Nonspeech dataset [37], and the colored noise recordings [38]. Four noise sources are drawn from collected noise data for evaluation: *F16*, *factory welding*, and *voice babble* from the RSG-10 dataset, and *street music* from the UrbanSound dataset. This results in a noise set comprising 6 809 noise clips. For validation experiments, we randomly draw 1 000 noise recordings and clean speech utterances to create a validation set of 1 000 noisy mixtures, where each clean speech utterance is mixed with a random clip from one noise recording at an SNR randomly sampled from -10 to 20 dB (in 1 dB increments). All audio signals are with a sampling frequency of 16 kHz. We extract the STFT magnitude spectrum using a square-root-Hann window of length 32 ms (512 time samples) with a frame shift of 16 ms (256 time samples).

Length Generalization. We segment the clean speech utterances in the training set into 1-second (1s) and 2-second (2s) speech clips for training, respectively. For evaluation experiments, we generate six test sets, each comprising 400 noisy mixtures of 1s, 2s, 5s, 10s, 15s, and 20s, respectively. The process of generating mixtures is as follows: For each of the four test noise recordings, we randomly take 20 clean speech (over durations of 1s, 2s, 5s, 10s, 15s, and 20s, respectively) from

the LibriSpeech *test-clean-100* corpus. A random clean clip (1s, 2s, 5s, 10s, 15s, and 20s in length) is mixed with a random noise clip (with the same length) from the noise recording, across SNRs from -5 and 15 dB, in 5 dB increments. The validation lengths have a broad range (diversity) and most of the speech utterances are more than four times (4s or 8s) the length of the training set (1s or 2s).

5.2. Implementation Details

To explore the impact of position embeddings in length generalization, we employ a Transformer backbone without position embedding (No-Pos) as a base. The Transformer backbone contains $N=4$ Transformer layers, with the parameter configurations as in [21]: $h=8$, $d_{model}=256$, and $d_{ff}=1024$. Our experiments explore length generalization across four commonly used position embeddings, Sinusoidal [30], learnable-APE [31], T5-RPE [28], and KERPLE [29]. The original KERPLE is proposed for causal language modeling. In this work, a non-causal KERPLE is derived to investigate length generalization.

Training Strategy. For each training iteration, we select 10 clean speech utterances from the training data and segment them into 1s and 2s clean speech clips, respectively. The speech clips are dynamically mixed with noise clips during the training stage. Specifically, each clean clip in a mini-batch is mixed with a random clip from a random noise recording at a random SNR sampled between -10 and 20 dB (in 1 dB increments). We employ mean-square error (MSE) as the objective function for mask estimation. The Adam algorithm is adopted for the optimization of gradients, with the hyper-parameter settings as [30], $\beta_1=0.9$, $\beta_2=0.98$, and $\epsilon=1 \times 10^{-9}$. The models are trained for 150 epochs. The gradient clipping technique is utilized to keep the gradients between -1 and 1. A warm-up scheduler [39, 30] is employed to dynamically adapt the learning rate, which is formulated as: $lr = d_{model}^{-0.5} \cdot \min(n_{itr}^{-0.5}, n_{itr} \cdot w_{itr}^{-1.5})$, where n_{itr} and w_{itr} denote the number of training iterations and warm-up stage iterations, respectively. Following the study [16], w_{itr} is set to 40 000. We employ an NVIDIA Tesla P100-PCIe-16GB graphics processing unit to run our experiments.

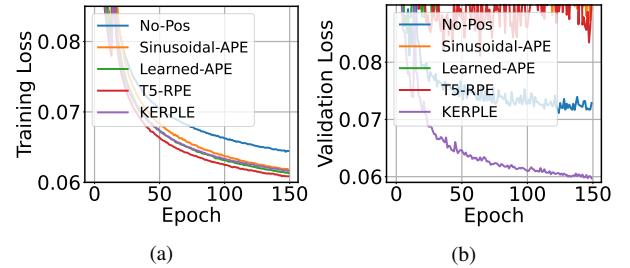


Figure 3: The (a) training loss and (b) validation loss of the models trained using speech of length 1s.

5.3. Training and Validation Loss

Fig. 3-4 give the train and validation loss curves of the models trained on 1s and 2s speech clips, respectively. It can be found that all the models converge well on the training data. Sinusoidal-APE and Learned-APE fail to perform length generalization. In comparison to No-Pos, they consistently achieve lower training loss but much higher validation loss. We can also observe significant length generalization issues with T5-RPE trained using 1s speech utterances. In contrast, KERPLE consistently demonstrates remarkable length generalization ca-

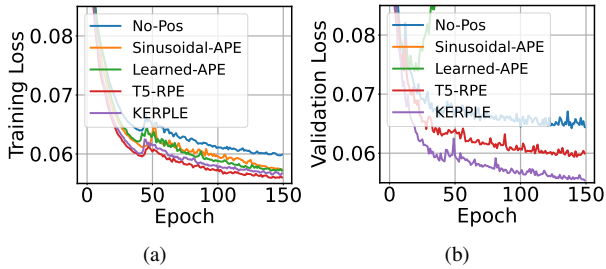


Figure 4: The (a) training loss and (b) validation loss of the models trained using speech of length 2s.

Table 1: The comparison results of PESQ, ESTOI (in %), CSIG, CBAK, and COVL. All the models are trained using 1s noisy-clean speech pairs and tested on noisy mixtures of lengths 1s, 5s, 10s, 15s, and 20s.

| Test Len. | Model | Metrics | | | | |
|-----------|----------------|-------------|--------------|-------------|-------------|-------------|
| | | PESQ | ESTOI | CSIG | CBAK | COVL |
| 1s | Noisy | 1.88 | 54.26 | 2.33 | 1.86 | 1.71 |
| | No-Pos | 2.59 | 68.17 | 3.18 | 2.60 | 2.48 |
| | Sinusoidal-APE | 2.70 | 70.96 | 3.29 | 2.68 | 2.59 |
| | Learned-APE | 2.72 | 71.12 | 3.30 | 2.68 | 2.60 |
| | T5-RPE | 2.73 | 71.60 | 3.34 | 2.70 | 2.63 |
| | KERPLE | 2.71 | 70.51 | 3.29 | 2.69 | 2.60 |
| 5s | Noisy | 1.88 | 54.26 | 2.33 | 1.86 | 1.71 |
| | No-Pos | 2.49 | 66.02 | 3.10 | 2.50 | 2.38 |
| | Sinusoidal-APE | 2.42 | 62.50 | 2.94 | 2.41 | 2.26 |
| | Learned-APE | 2.11 | 55.49 | 2.61 | 2.21 | 1.96 |
| | T5-RPE | 2.52 | 65.61 | 3.12 | 2.48 | 2.40 |
| | KERPLE | 2.69 | 70.27 | 3.32 | 2.64 | 2.59 |
| 10s | Noisy | 1.92 | 53.19 | 2.40 | 1.87 | 1.75 |
| | No-Pos | 2.46 | 63.89 | 3.09 | 2.45 | 2.36 |
| | Sinusoidal-APE | 2.38 | 60.17 | 2.92 | 2.35 | 2.22 |
| | Learned-APE | 2.08 | 53.44 | 2.54 | 2.15 | 1.91 |
| | T5-RPE | 2.42 | 61.01 | 3.01 | 2.37 | 2.29 |
| | KERPLE | 2.70 | 68.62 | 3.32 | 2.63 | 2.59 |
| 15s | Noisy | 1.90 | 52.68 | 2.32 | 1.83 | 1.71 |
| | No-Pos | 2.43 | 63.18 | 3.01 | 2.39 | 2.30 |
| | Sinusoidal-APE | 2.35 | 58.24 | 2.83 | 2.28 | 2.15 |
| | Learned-APE | 2.05 | 51.42 | 2.46 | 2.09 | 1.85 |
| | T5-RPE | 2.35 | 58.89 | 2.89 | 2.29 | 2.19 |
| | KERPLE | 2.67 | 68.24 | 3.25 | 2.58 | 2.54 |
| 20s | Noisy | 1.91 | 51.42 | 2.36 | 1.82 | 1.72 |
| | No-Pos | 2.43 | 61.50 | 3.05 | 2.41 | 2.32 |
| | Sinusoidal-APE | 2.39 | 56.71 | 2.87 | 2.29 | 2.19 |
| | Learned-APE | 2.03 | 50.19 | 2.44 | 2.08 | 1.83 |
| | T5-RPE | 2.40 | 57.02 | 2.88 | 2.28 | 2.18 |
| | KERPLE | 2.68 | 66.88 | 3.30 | 2.61 | 2.57 |

pability, with substantially lower training loss and validation loss compared to No-Pos. Overall, among these methods, RPE methods show a better length generalization property than APE.

5.4. Results and Analysis

We evaluate the performance of models with five objective metrics, i.e., perceptual evaluation of speech quality (PESQ) [40], extended short-time objective intelligibility (ESTOI) [41], and three composite metrics (CSIG, CBAK, and COVL) [42].

Tables 1-2 respectively report the evaluation results of models trained on 1s and 2s speech clips in terms of PESQ, ESTOI, CSIG, CBAK, and COVL, across five test lengths. For matched test lengths (1s or 2s), all the position embedding methods provide substantial improvements over the No-Pos and exhibit similar performance, with T5-RPE showcasing the best results, which is consistent with the loss trends in Fig. 3-4. For the test lengths from 5s to 20s, among these methods, overall

Table 2: The comparison results of PESQ, ESTOI (in %), CSIG, CBAK, and COVL. All the models are trained using 2s noisy-clean speech pairs and tested on noisy mixtures of lengths 2s, 5s, 10s, 15s, and 20s.

| Test Len. | Model | Metrics | | | | |
|-----------|----------------|-------------|--------------|-------------|-------------|-------------|
| | | PESQ | ESTOI | CSIG | CBAK | COVL |
| 2s | Noisy | 1.88 | 53.94 | 2.36 | 1.87 | 1.74 |
| | No-Pos | 2.58 | 67.19 | 3.23 | 2.61 | 2.50 |
| | Sinusoidal-APE | 2.71 | 70.23 | 3.33 | 2.69 | 2.61 |
| | Learned-APE | 2.70 | 70.55 | 3.33 | 2.70 | 2.62 |
| | T5-RPE | 2.74 | 71.15 | 3.37 | 2.71 | 2.65 |
| | KERPLE | 2.71 | 70.34 | 3.34 | 2.69 | 2.62 |
| 5s | Noisy | 1.88 | 54.26 | 2.33 | 1.86 | 1.71 |
| | No-Pos | 2.56 | 67.40 | 3.19 | 2.57 | 2.47 |
| | Sinusoidal-APE | 1.89 | 53.44 | 2.31 | 1.91 | 1.71 |
| | Learned-APE | 2.06 | 59.48 | 2.61 | 2.10 | 1.92 |
| | T5-RPE | 2.74 | 71.29 | 3.39 | 2.70 | 2.66 |
| | KERPLE | 2.74 | 71.38 | 3.37 | 2.67 | 2.64 |
| 10s | Noisy | 1.92 | 53.19 | 2.40 | 1.87 | 1.75 |
| | No-Pos | 2.52 | 65.35 | 3.19 | 2.53 | 2.45 |
| | Sinusoidal-APE | 1.94 | 53.23 | 2.37 | 1.92 | 1.76 |
| | Learned-APE | 1.99 | 54.68 | 2.47 | 1.97 | 1.82 |
| | T5-RPE | 2.69 | 67.99 | 3.35 | 2.64 | 2.62 |
| | KERPLE | 2.74 | 69.95 | 3.39 | 2.67 | 2.65 |
| 15s | Noisy | 1.90 | 52.68 | 2.32 | 1.83 | 1.71 |
| | No-Pos | 2.47 | 64.44 | 3.10 | 2.46 | 2.37 |
| | Sinusoidal-APE | 1.92 | 52.78 | 2.28 | 1.88 | 1.70 |
| | Learned-APE | 1.96 | 52.49 | 2.40 | 1.89 | 1.76 |
| | T5-RPE | 2.60 | 66.62 | 3.24 | 2.54 | 2.51 |
| | KERPLE | 2.72 | 69.59 | 3.35 | 2.64 | 2.62 |
| 20s | Noisy | 1.91 | 51.42 | 2.36 | 1.82 | 1.72 |
| | No-Pos | 2.49 | 62.73 | 3.13 | 2.47 | 2.40 |
| | Sinusoidal-APE | 1.91 | 51.37 | 2.33 | 1.86 | 1.72 |
| | Learned-APE | 1.94 | 51.55 | 2.40 | 1.87 | 1.75 |
| | T5-RPE | 2.61 | 64.20 | 3.26 | 2.54 | 2.52 |
| | KERPLE | 2.74 | 68.36 | 3.39 | 2.65 | 2.65 |

KERPLE exhibits the best length generalization property. In the 10s and 20s test lengths case (1s training length, shown in Table 1), KERPLE attain the gains of 0.24 and 0.25, 4.73% and 5.38%, 0.23 and 0.25, 0.18 and 0.20, 0.23 and 0.25, on PESQ, ESTOI, CSIG, CBAK, and COVL, respectively, over No-Pos. Sinusoidal-APE and Learned-APE exhibit similar length generalization issues, with substantially lower performance scores compared to No-Pos. Taking the 10s test length case (2s training length), compared to No-Pos, Sinusoidal-APE and Learned-APE show decreases of 0.58 and 0.53, and 12.12% and 10.67% in PESQ and ESTOI, respectively. T5-RPE also shows a similar length generalization issue for 1s training length.

6. Conclusion

In this paper, we establish and explore the length generalization problem in speech enhancement models with Transformers. We introduce four position embedding methods to enable Transformer-based speech enhancement models with the property to generalize to longer utterances. The comprehensive experiment results in five metrics underscore the superiority of relative position embedding (RPE) over absolute position embedding (APE) in terms of length generalization. APE methods, i.e., Sinusoidal-APE and Learned-APE are not able to generalize from shorter speech to long ones. We believe our work sets the stage for future research on length generalization for speech enhancement and some other speech processing tasks with Transformers.

7. References

- [1] P. C. Loizou, *Speech enhancement: theory and practice*. CRC press, 2013.
- [2] Q. Zhang, M. Wang, Y. Lu, L. Zhang, and M. Idrees, “A novel fast nonstationary noise tracking approach based on mmse spectral power estimator,” *Digital Signal Processing*, vol. 88, pp. 41–52, 2019.
- [3] Q. Zhang, M. Wang, Y. Lu, M. Idrees, and L. Zhang, “Fast nonstationary noise tracking based on log-spectral power mmse estimator and temporal recursive averaging,” *IEEE Access*, vol. 7, pp. 80 985–80 999, 2019.
- [4] D. Wang and J. Chen, “Supervised speech separation based on deep learning: An overview,” *IEEE/ACM TASLP*, vol. 26, no. 10, pp. 1702–1726, 2018.
- [5] X. Zhang, Q. Zhang, H. Liu, T. Xiao, X. Qian, B. Ahmed, E. Ambikairajah, H. Li, and J. Epps, “Mamba in speech: Towards an alternative to self-attention,” *arXiv*, 2024.
- [6] A. Defossez, G. Synnaeve, and Y. Adi, “Real time speech enhancement in the waveform domain,” in *INTERSPEECH*, 2020.
- [7] Z. Kong, W. Ping, A. Dantrey, and B. Catanzaro, “Speech denoising in the waveform domain with self-attention,” in *Proc. ICASSP*, 2022, pp. 7867–7871.
- [8] K. Tan and D. Wang, “Learning complex spectral mapping with gated convolutional recurrent networks for monaural speech enhancement,” *IEEE/ACM TASLP*, vol. 28, pp. 380–390, 2019.
- [9] Y. Xu, J. Du, L.-R. Dai, and C.-H. Lee, “A regression approach to speech enhancement based on deep neural networks,” *IEEE/ACM TASLP*, vol. 23, no. 1, pp. 7–19, 2014.
- [10] Y. Wang, A. Narayanan, and D. Wang, “On training targets for supervised speech separation,” *IEEE/ACM TASLP*, vol. 22, no. 12, pp. 1849–1858, 2014.
- [11] D. S. Williamson, Y. Wang, and D. Wang, “Complex ratio masking for monaural speech separation,” *IEEE/ACM TASLP*, vol. 24, no. 3, pp. 483–492, 2015.
- [12] H. Erdogan, J. R. Hershey, S. Watanabe, and J. Le Roux, “Phase-sensitive and recognition-boosted speech separation using deep recurrent neural networks,” in *Proc. ICASSP*, 2015, pp. 708–712.
- [13] J. Chen and D. Wang, “Long short-term memory for speaker generalization in supervised speech separation,” *JASA*, vol. 141, no. 6, pp. 4705–4714, 2017.
- [14] F. Weninger, H. Erdogan, S. Watanabe, E. Vincent, J. L. Roux, J. R. Hershey, and B. Schuller, “Speech enhancement with lstm recurrent neural networks and its application to noise-robust asr,” in *LVA/ICA*. Springer, 2015, pp. 91–99.
- [15] Q. Zhang, Q. Song, A. Nicolson, T. Lan, and H. Li, “Temporal convolutional network with frequency dimension adaptive attention for speech enhancement,” *Proc. INTERSPEECH*, pp. 166–170, 2021.
- [16] Q. Zhang, X. Qian, Z. Ni, A. Nicolson, E. Ambikairajah, and H. Li, “A time-frequency attention module for neural speech enhancement,” *IEEE/ACM TASLP*, vol. 31, pp. 462–475, 2023.
- [17] A. Pandey and D. Wang, “TCNN: Temporal convolutional neural network for real-time speech enhancement in the time domain,” in *Proc. ICASSP*, 2019, pp. 6875–6879.
- [18] Q. Zhang, A. Nicolson, M. Wang, K. K. Paliwal, and C. Wang, “Monaural speech enhancement using a multi-branch temporal convolutional network,” *arXiv preprint arXiv:1912.12023*, 2019.
- [19] Q. Zhang, Q. Song, Z. Ni, A. Nicolson, and H. Li, “Time-frequency attention for monaural speech enhancement,” in *Proc. ICASSP*, 2022, pp. 7852–7856.
- [20] S. Bai, J. Z. Kolter, and V. Koltun, “An empirical evaluation of generic convolutional and recurrent networks for sequence modeling,” *arXiv preprint arXiv:1803.01271*, 2018.
- [21] Q. Zhang, H. Zhu, Q. Song, X. Qian, Z. Ni, and H. Li, “Ripple sparse self-attention for monaural speech enhancement,” in *Proc. ICASSP*, 2023, pp. 1–5.
- [22] D. de Oliveira, T. Peer, and T. Gerkmann, “Efficient transformer-based speech enhancement using long frames and STFT magnitudes,” in *Proc. INTERSPEECH*, 2022, pp. 2948–2952.
- [23] J. Kim, M. El-Khomy, and J. Lee, “T-GSA: Transformer with gaussian-weighted self-attention for speech enhancement,” in *Proc. ICASSP*, 2020, pp. 6649–6653.
- [24] Q. Zhang, M. Ge, H. Zhu, E. Ambikairajah, Q. Song, Z. Ni, and H. Li, “An empirical study on the impact of positional encoding in transformer-based monaural speech enhancement,” in *Proc. ICASSP*, 2024, pp. 1001–1005.
- [25] S. Ren, X. Yang, S. Liu, and X. Wang, “Sg-former: Self-guided transformer with evolving token reallocation,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 6003–6014.
- [26] O. Press, N. Smith, and M. Lewis, “Train short, test long: Attention with linear biases enables input length extrapolation,” in *Proc. ICLR*, 2021.
- [27] J. Su, Y. Lu, S. Pan, A. Murtadha, B. Wen, and Y. Liu, “Ro-former: Enhanced transformer with rotary position embedding,” *arXiv preprint arXiv:2104.09864*, 2021.
- [28] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, P. J. Liu *et al.*, “Exploring the limits of transfer learning with a unified text-to-text transformer,” *J. Mach. Learn. Res.*, vol. 21, no. 140, pp. 1–67, 2020.
- [29] T.-C. Chi, T.-H. Fan, P. J. Ramadge, and A. Rudnicky, “Kerple: Kernelized relative positional embedding for length extrapolation,” *Proc. NeurIPS*, vol. 35, pp. 8386–8399, 2022.
- [30] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” in *NeurIPS*, 2017, pp. 5998–6008.
- [31] J. D. M.-W. C. Kenton and L. K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” in *Proc. of NAACL-HLT*, 2019, pp. 4171–4186.
- [32] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, “Librispeech: an asr corpus based on public domain audio books,” in *Proc. ICASSP*, 2015, pp. 5206–5210.
- [33] D. B. Dean, S. Sridharan, R. J. Vogt, and M. W. Mason, “The QUT-NOISE-TIMIT corpus for the evaluation of voice activity detection algorithms,” in *Proc. INTERSPEECH*, 2010.
- [34] F. Saki, A. Sehgal, I. Panahi, and N. Kehtarnavaz, “Smartphone-based real-time classification of noise signals using subband features and random forest classifier,” in *Proc. ICASSP*, 2016, pp. 2204–2208.
- [35] D. Snyder, G. Chen, and D. Povey, “MUSAN: A music, speech, and noise corpus,” *arXiv preprint arXiv:1510.08484*, 2015.
- [36] J. Salamon, C. Jacoby, and J. P. Bello, “A dataset and taxonomy for urban sound research,” in *ACM-MM*, 2014, pp. 1041–1044.
- [37] G. Hu and D. Wang, “A tandem algorithm for pitch estimation and voiced speech segregation,” *IEEE TASLP*, vol. 18, no. 8, pp. 2067–2079, 2010.
- [38] Q. Zhang, A. Nicolson, M. Wang, K. K. Paliwal, and C. Wang, “DeepMMSE: A deep learning approach to mmse-based noise power spectral density estimation,” *IEEE/ACM TASLP*, vol. 28, pp. 1404–1415, 2020.
- [39] A. Nicolson and K. K. Paliwal, “Masked multi-head self-attention for causal speech enhancement,” *Speech Com.*, vol. 125, pp. 80–96, 2020.
- [40] R. I.-T. P. ITU, “862.2: Wideband extension to recommendation P. 862 for the assessment of wideband telephone networks and speech codecs. ITU-Telecommunication standardization sector, 2007.”
- [41] J. Jensen and C. H. Taal, “An algorithm for predicting the intelligibility of speech masked by modulated noise maskers,” *IEEE/ACM TASLP*, vol. 24, no. 11, pp. 2009–2022, 2016.
- [42] Y. Hu and P. C. Loizou, “Evaluation of objective quality measures for speech enhancement,” *IEEE TASLP*, vol. 16, no. 1, pp. 229–238, 2007.