IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received December 22, 2020, accepted December 31, 2020, date of publication January 4, 2021, date of current version January 15, 2021.

Digital Object Identifier 10.1109/ACCESS.2020.3049073

Document-Level Neural TTS Using Curriculum Learning and Attention Masking

SUNG-WOONG HWANG AND JOON-HYUK CHANG, (Senior Member, IEEE)
Department of Electronic Engineering, Hanyang University, Seoul 04763, South Korea

Corresponding author: Joon-Hyuk Chang (jchang@hanyang.ac.kr)

This work was supported by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant by Korean Government through the Ministry of Science and ICT (MSIT) (Deep learning multi-speaker prosody and emotion cloning technology based on a high quality end-to-end model using small amount of data) under Grant 2020-0-00059.

ABSTRACT Speech synthesis has been developed to the level of natural human-level speech synthesized through an attention-based end-to-end text-to-speech synthesis (TTS) model. However, it is difficult to generate attention when synthesizing a text longer than the trained length or document-level text. In this paper, we propose a neural speech synthesis model that can synthesize more than 5 min of speech at once using training data comprising a short speech of less than 10 s. This model can be used for tasks that need to synthesize document-level speech at a time, such as a singing voice synthesis (SVS) system or a book reading system. First, through curriculum learning, our model automatically increases the length of the speech trained for each epoch, while reducing the batch size so that long sentences can be trained with a limited graphics processing unit (GPU) capacity. During synthesis, the document-level text is synthesized using only the necessary contexts of the current time step and masking the rest through an attention-masking mechanism. The Tacotron2-based speech synthesis model and duration predictor were used in the experiment, and the results showed that proposed method can synthesize document-level speech with overwhelmingly lower character error rate, and attention error rates, and higher quality than those obtained using the existing model.

INDEX TERMS Speech synthesis, document-level neural TTS, curriculum learning, attention masking, Tacotron2, MelGAN, DeepVoice3, ParaNet, MultiSpeech.

I. INTRODUCTION

Speech synthesis (text-to-speech synthesis, TTS), which produces natural speech from text, is an active research area. With the advent of an end-to-end speech synthesis model based on a deep neural network (DNN), the quality of synthesized speech has significantly improved compared with that generated using the previous concatenative synthesis model [1], [2] and statistical parametric speech synthesis model [3]–[6]. Tacotron [7] is a representative end-to-end speech synthesis model based on DNN that simplifies the complex structure used to generate linguistic and acoustical features in the previous model; it is achieved by generating a mel spectrogram from the text sequence through a single neural network and synthesizing speech using the Griffin and Lim [8] algorithm as a vocoder. The result of speech synthesis through Tacotron2 [9], an enhanced Tacotron model [7], has improved to a level substantially similar to the level of natural human speech. However, this model cannot synthesize sentences longer than the trained speech length, and various problems such as missing or repeated words and incomplete synthesis occur when attempting document-level speech synthesis.

The current end-to-end natural speech synthesis system uses a sequence-to-sequence model comprising two structures: an encoder and a decoder. In the encoder, the input sequence is compressed into a fixed-size vector, and in the decoder, the output sequence is generated using the context vector output from the encoder. When the encoder tries to compress all information into a fixed-sized vector, information loss occurs, and an attention mechanism is used to solve this problem. The attention mechanism is one of the most important factors in enabling document-level speech synthesis. In the original Tacotron [7] system, the content-based attention mechanism introduced in [10] is used to align the target text and output a spectrogram. However, with this mechanism, it is difficult to synthesize speech longer than the trained text length. In the Tacotron2 [9] system, it is possible

The associate editor coordinating the review of this manuscript and approving it for publication was Li He.

to synthesize sentences longer than the length of the trained text by applying location-sensitive attention [11], an attention mechanism that generates alignment using not only the encoder and decoder information but also the information of the previous time step. However, this approach is still insufficient to synthesize text. Recently, in [12], a new mechanism was proposed that can synthesize long sentences by training only short sentences by modifying the previously used attention mechanism. The study proposed GMMv2 attention and a deep convolution attention model by modifying the location-sensitive attention used in Tacotron2 and the pure location-based GMM attention introduced in [13]. Through this approach, it is possible to synthesize natural speech of approximately 80 s by training with a speech of 10 s or less; however, it is still insufficient to synthesize document-level text into speech.

In this paper, we propose a document-level neural TTS model that synthesizes document-level text into speech using curriculum learning [14] and attention masking. First, through curriculum learning, whenever the epoch increases, the longer sentence is trained to make the proposed model robust to the document-level text, and then attention masking is used when synthesizing. In [15]–[17], attention masking was used to overcome attention errors such as repetition and pronunciation errors in speech synthesis; however, in our model, we aim to use it to synthesize document-level text into speech. Reference [15] used the method that involved measuring which part of the context is given the highest attention weight when generating the current mel spectrogram, and then masking the rest except the certain window behind that part. In [16], the part to mask is determined by an approximate ratio of the mel spectrogram to the alignment length. After determining which part of the alignment is to be used to generate the mel spectrogram at the current time step based on the ratio, the method masks the rest of the alignment, except for a certain part. In [17], a method similar to that of [15] was used; however when the part with the highest attention weight moves to the next context three times, the reference point is changed and the remaining parts, except for the first one and the last four, are masked. In the existing models, two methods have been used to determine a reference point for attention masking. The first method is to use the highest attention weight as a reference point [15], [17], and the second method is to determine the approximate ratio of the mel spectrogram length and the alignment length in advance, and then determine the reference point for attention masking by dividing the length of the mel spectrogram generated so far when synthesizing speech by this ratio [16]. At this time, since the entire mel spectrogram length of the synthesized sentence cannot be known in advance, the exact ratio cannot be known and the accuracy of attention masking is degraded. To improve the accuracy of attention masking used in the second method, we do not simply arbitrarily specify the ratio of the length of the mel spectrogram and the alignment. Instead, by training the duration for each character of the text through the duration predictor used in the non-autoregressive model [16], [18], the length of the entire mel spectrogram can be predicted by using the input text only. Through this, the exact length ratio of the mel spectrogram to the alignment can be known, and more accurate attention masking is possible.

II. MODEL

As depicted in Fig. 1, the proposed system includes a model that trains long texts through curriculum learning by combining the input sentences whenever the epoch increases. As shown in Fig. 2, our model prevents synthesis failure even if the length of the text to be synthesized is longer, while using only the alignment of the essential part for generating the mel spectrogram in the current time step with attention masking during synthesis. The accuracy of attention masking is increased by predicting the length of the mel spectrogram with the text to be synthesized using the duration predictor.

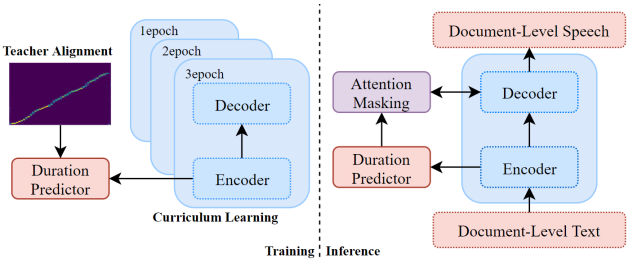

FIGURE 1. Block diagram of document-level neural TTS system. In the training part of the proposed model, we start from short sentences and train longer sentences through curriculum learning, and through the duration predictor, we are able to predict the length of the output mel spectrogram by using the input text only (left side of figure). Also, the inference part uses a duration predictor to mask the attention of unnecessary parts and synthesize document-level speech (right side of figure).

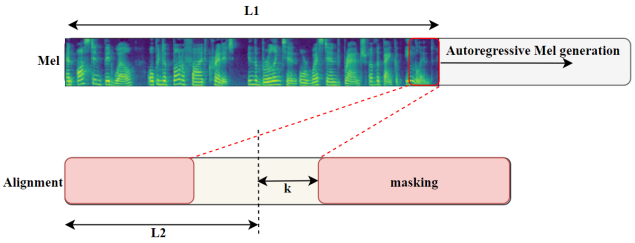

FIGURE 2. System architecture of attention masking. The synthesis of document-level text fails when a mel spectrogram is generated considering the entire alignment. Predicting the length of the entire mel spectrogram through the duration predictor and dividing it by the alignment length is equivalent to dividing the length of the mel spectrogram generated up to the current time step (L1) by the position of the alignment needed to generate the mel spectrogram of the current time step (L2). Through this process, the L2 value, which is the position of the required alignment, can be known, and the speech is synthesized by referring only to the alignment within the range of an arbitrary *k* value (25 used in this study).

A. CURRICULUM LEARNING

Curriculum learning is a deep learning method first proposed in [14]; it starts with simple data and then the difficulty of learning increases with increasingly complex data. For example, in a shape-recognition task, curriculum learning

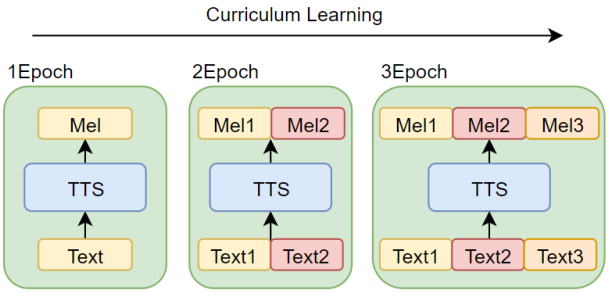

FIGURE 3. Block diagram of the curriculum learning.

is performed using accurately shaped circles, squares, etc., as simple data and rectangles and ellipses as complex data. In the speech task, learning complexity typically increases from less noisy to more noisy data.

Because our purpose is to synthesize document-level text into speech, we begin training with short sentences and gradually adopt long sentences based on curriculum learning. In the first epoch, the model is trained using the original training data, and in the second epoch, the input data are randomly combined by two sentences so that a longer sentence can be used for training. Both the text to be trained and the true mel spectrogram must be combined; however if they are simply joined, an unnatural voice is generated in the joining step. To prevent this, we insert a t-token (text token) and an m-token (mel spectrogram token) between each text and the mel spectrogram. The t-token plays the role of distinguishing between texts, and at the same time, it learns the m-token, which allows the speech to lead naturally between mel spectrograms. Therefore, a new character that is not used in learning is used for the t-token. The m-token is structured to allow for a gap in speech for approximately a second because sentences must be naturally linked. The longer the input text, the greater the capacity of the GPU required for training, and therefore the batch size automatically decreases in proportion to the longer input so that it can be trained within the limited GPU capacity. Similarly, in the third epoch, we arbitrarily combine three input sentences and mel spectrograms in the training data, use t-tokens and m-tokens to maintain the naturalness of the speech, and reduce the batch size over the length to enable curriculum learning with a limited GPU capacity. The process of training by combining text and mel spectrograms while considering their GPU capacity is repeated, and when the maximum capacity is reached, it is possible to return to the process of training a pair of texts and mel spectrograms. For example, in the third epoch, if the batch size becomes too small or the GPU capacity is exceeded, then from the fourth epoch, one can train the data one sentence at a time and repeat the process.

B. ATTENTION MASKING USING THE DURATION PREDICTOR

If we apply attention masking as shown in Fig. 2, it is possible to mask the alignment of parts that are not needed when generating the output at the current time step, thus improving the efficiency of the synthesis and reducing the attention error. If the synthesized text becomes longer, the length of the alignment to be referenced when generating the mel spectrogram at a specific time step increases, and it becomes difficult to accurately determine where to focus attention. We attempt to overcome this problem through attention masking, and then we implement a model that synthesizes document-level text into speech well.

Tacotron2 [9], which is used as the base speech synthesis model in our study, generates mel spectrograms autoregressively; therefore, a part of the alignment required to generate the mel spectrogram is different for each time step. Owing to the characteristics of speech synthesis, monotonous alignment is generated because the order of the output sequence corresponding to each input sequence is the same. In other words, when the mel spectrogram is generated at the beginning of a sentence, the front part of the alignment is important, and therefore, by masking the back part, one can discard the unnecessary parts and can emphasize the front part by adding weight. Conversely, when generating the backward part of the mel spectrogram, one can mask the front of the alignment. We use the ratio of the total length of the mel spectrogram to the length of the alignment to find the location of the alignment necessary to generate the mel spectrogram at a specific time step. First, the duration predictor proposed by FastSpeech [18] is used to estimate the length of the output sequence of the mel spectrogram using only the input text.

The original purpose of the duration predictor is to better predict the output of the decoder by predicting the length of the mel spectrogram and changing it to the input of the decoder through the length regulator in the non-autoregressive speech synthesis model [16], [18]. However, in this study, we used the duration predictor to determine which part of the alignment considered in generating the mel spectrogram of the current time step. First, the alignment is extracted from the Tacotron2 model previously trained as a teacher model, and the duration of the alignment is extracted through a duration extractor. Teacher alignment includes information about the number of frames generated by each sequence of characters extracted from the text. As shown in [18], the character sequence, H_{char} can be expressed as follows:

$$H_{char} = [h_1, h_2, \ldots, h_n] \tag{1}$$

The output from the duration extractor, D can be expressed as:

$$D = [d_1, d_2, \ldots, d_n] \tag{2}$$

Thus, the length of the mel spectrogram is obtained as:

$$m = \sum_{i=1}^{n} d_i \tag{3}$$

When training the proposed model, we pass the encoder output through two 1D convolutional layers and one linear layer to predict the duration of the character sequence.

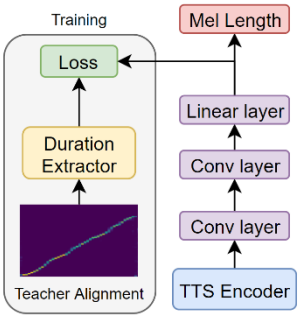

FIGURE 4. Block diagram of the duration predictor.

We train the predicted duration with the true duration obtained through teacher alignment, and predict the length of the mel spectrogram when synthesizing.

By following the approach described above, we can obtain the ratio of the mel spectrogram length to the alignment length, which is equal to the ratio of the length of the mel spectrogram generated for the current time step, shown in Fig. 2 (L1), to the part of the alignment that we should consider (L2). Through this, the mel spectrogram of the current time step is generated while considering only the alignment of a specific length $2k$, centered on the obtained L2, and the remaining alignment is masked. When attention masking is applied to an input text that is too short, sufficient alignment cannot be considered when synthesizing speech, and therefore, a threshold value is needed so that attention masking can be applied only to texts of at least a certain length.

III. EXPERIMENTS & EVALUATION

A. EXPERIMENTAL CONDITIONS IN THE TTS EVALUATION

We used approximately 24 h of the LJSpeech dataset [19], which contains 13,100 English audio clips recorded by a female speaker. We used 12,700 random sentences for training and 400 sentences for evaluation. As we implemented and tested various speech synthesis models, Tacotron2 yielded the best sound quality and the highest accuracy of speech synthesis. Thus, we used Tacotron2 [9] as a base model, with the addition of a curriculum learning and an attention masking model, and a single Nvidia GeForce RTX2080 GPU. The batch size started at 12, and the model was set to automatically reduce the batch size to $1/n$ when learning n sentences combined to synthesize long sentences through curriculum learning within a limited GPU capacity. In the case of attention masking, the specific range k that did not perform masking was set to 25, which was the character sequence length of approximately half sentences, so that speech could be synthesized by referring only to the alignment, which was approximately a sentence long. A character sequence length of 300 was set as the threshold value of the minimum sentence length to which attention masking was applied, so that the input sequence could be judged as document-level text when it exceeded approximately three sentences. MelGAN [20] was trained and used as a vocoder for speech synthesis.

B. DOCUMENT-LEVEL NEURAL TTS

The main advantage of our proposed document-level neural TTS is that it can synthesize long sentences at once. To verify this, we conducted a test using the script of a Harry Potter novel and evaluated it according to the length and time of the synthesized speech. As shown in Fig. 5(a), to compare the length of the sentence that could be synthesized with the existing model, we used a Tacotron model [7] using content-based attention [10] and a Tacotron2 model [9] using location-sensitive attention [11]. We also conducted a performance comparison experiment of the existing model according to the application of curriculum learning in the proposed model. For this experiment, the synthesized speech was converted to text through the Google Cloud Speech-to-Text API service and compared it with the original text, and the character error rate (CER) was measured. With the CER, we used the attention error rate (AER) to assess the accuracy

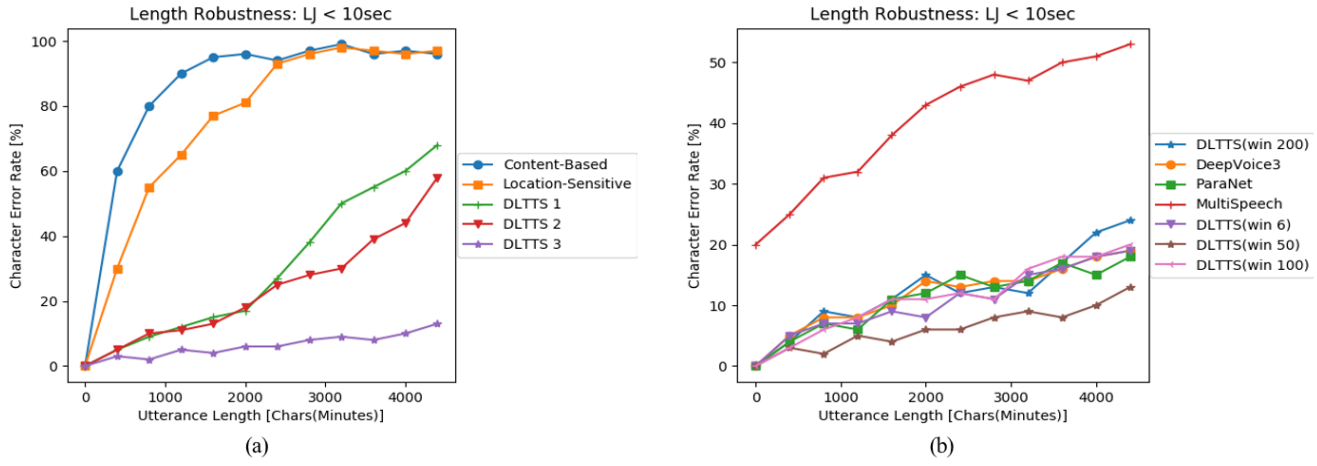

FIGURE 5. Utterance length robustness of the proposed model. The document-level neural TTS model confirmed that the character error rate (CER) remains low even when synthesizing long sentences and that it can synthesize more accurate speech than that synthesized using other models.

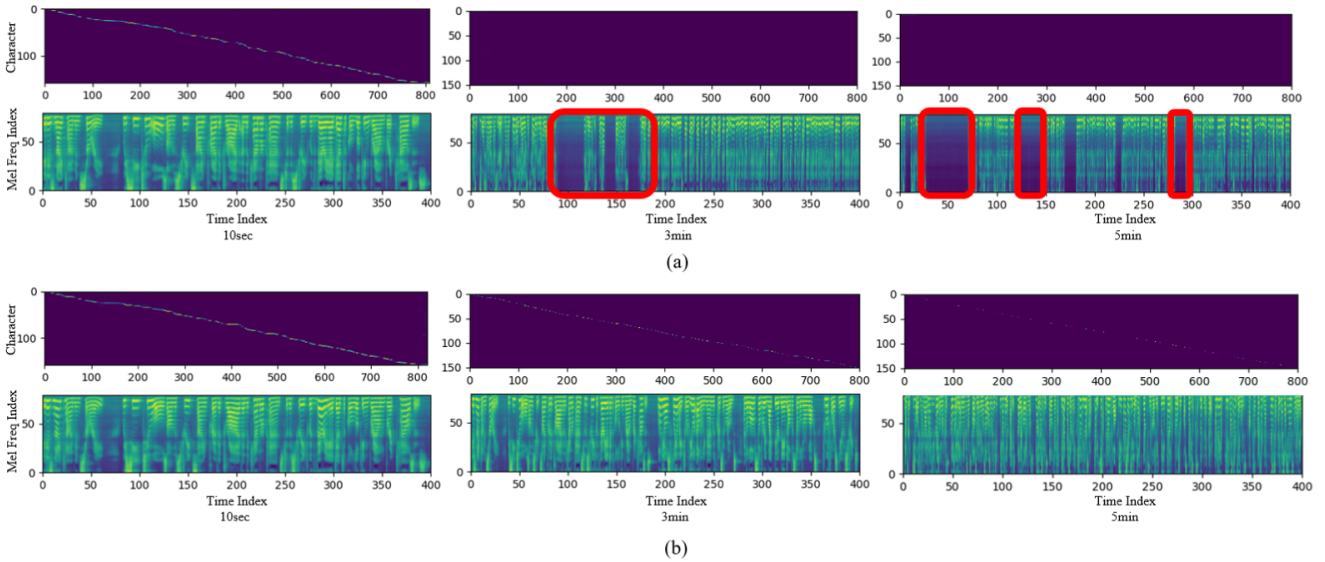

FIGURE 6. Generation of the alignment and mel spectrograms according to the length of synthesized speech in the proposed model. (a) From the result of the synthesis in the Tacotron2 model based on location-sensitive attention, we can see that long sentences to be synthesized generate poor alignment, and the mel spectrogram is destroyed. (b) From the result of the synthesis in the document-level neural TTS model we propose, we can confirm that both the alignment and the mel spectrogram are well generated.

of the generated speech. The AER is the ratio of speeches with attention error such as repeating, skipping words or mispronouncing among all generated speeches. Our model did not exceed the initial 20% of the CER until we synthesized 5 min and 30 s of the speech (about 4400 characters); in contrast, the content-based attention model was found to exceed 20% in 10 s and the location-sensitive attention model exceeded 20% in 30 s. When curriculum learning was not applied (DLTTS1), or two sentences were added together (DLTTS2), we confirmed that the CER rose to the 60% range in the speech synthesis environment of more than 5 min; however, but the CER fell to the lower 10% range when curriculum learning was applied by adding three sentences together (DLTTS3). Fig. 5(b) shows a comparison of the performance in a document-level speech synthesis environment by applying it to a model that suggests alternative attention masking of previously used models. When the attention masking used in MultiSpeech [17] was applied to a speech synthesis environment of more than 5 min, the CER was more than 50% and the speech was not synthesized properly. In the model, the reference point for attention masking is the part with the highest attention weight, and then the attention weight of the following context is three times greater than the weights of the previous context, passing the reference point to the next. The delay that occurs at this time will not occur in a document-level speech synthesis environment. On the other hand, the application of attention masking used in DeepVoice3 or ParaNet resulted in a similar level of CERs to that of the proposed model. In addition, from the result of the experiment where the window size of attention masking was changed in the proposed model, it was confirmed that when the size was 50, the lowest CER and attention error rate (AER) were obtained and good performance was achieved. In addition, we compared the mel spectrogram and attention synthesized from the document level to those of the existing model. We found that in the existing model, for longer sentences, part of the mel spectrogram was destroyed, and the attention was not generated. In contrast, in the proposed model, the results were generated without any problems, as shown in Fig. 6.

C. ROBUSTNESS

When document-level text is synthesized by speech, attention errors such as repeated and skipped words occur if the attention is not established between the encoder and the decoder. Table 1 shows that the proposed model has a very low AER at the document-level compared with that of the existing model. We tested 200 random documents for each sentence length and measured the number of times an attention error occurred. The Tacotron model using content-based attention had a high AER when synthesizing sentences longer than 30 s, and the Tacotron2 model using location-sensitive attention showed a high error rate when the length of the synthesized sentences exceeded 1 min.

Moreover, as shown in Table 2, the AER was measured by applying the attention masking proposed in the existing model. First, we found that the attention masking used in MultiSpeech is not suitable for the document-level speech synthesis model. In this model, when attention masking is used, the reference point is set as the point with the highest attention weight, and when this reference point is passed to the next, the weight of the next reference point must be three times higher than that of the previous reference point. This delay is not well applied when synthesizing document-level speech. In fact, from the result of the experiment with

TABLE 1. Comparison of the proposed model (DLTTS) by applying the attention masking used in each model.

Method	Text length	Attention error rate
Tacotron	10 s	4/200
	30 s	200/200
	1 min	200/200
Tacotron2	10 s	0/200
	30 s	24/200
	1 min	189/200
DeepVoice3	1 min	0/200
	3 min	7/200
	5 min	12/200
ParaNet	1 min	0/200
	3 min	5/200
	5 min	9/200
MultiSpeech	1 min	80/200
	3 min	178/200
	5 min	200/200
DLTTS	1 min	0/200
	3 min	0/200
	5 min	2/200

TABLE 2. Comparison of the proposed model with the existing model by applying curriculum learning.

Method	Text length	Attention error rate
without curriculum learning	1 min	1/200
	3 min	44/200
	5 min	101/200
curriculum learning with 2 sentences	1 min	0/200
	3 min	8/200
	5 min	57/200
curriculum learning with 3 sentences	1 min	0/200
	3 min	0/200
	5 min	2/200

this delay removed, it was confirmed that the document-level speech was synthesized well with a very low AER. It can be seen that the attention masking model used in DeepVoice3 or ParaNet shows a sufficiently low error rate when the proposed model was applied. However, our proposed document-level neural TTS model showed a relatively lower error rate even when synthesizing sentences longer than 5 min. This means that our model could stably synthesize document-level texts into speech.

In Table 3, we show the AER measured by changing the window size of the attention masking. The result showed that when the window size is 50, document-level speech can be synthesized with the lowest error rate. If the window size is too large, the length of the attention to be referred to while synthesizing the speech will be too long to be properly synthesized. Conversely, if the window size is too small, loss of information occurs because it does not refer to the required attention. Therefore, it can be seen that it is important to synthesize the speech by applying the appropriate window size to the attention masking.

D. SYNTHESIS QUALITY

To evaluate the synthesis quality, we conducted a listening test with 12 participants to measure the mean opinion score (MOS). We measured the MOS by applying the attention of the models in Table 4 to the proposed model. Because Tacotron and Tacotron2 do not synthesize document-level

TABLE 3. Comparison of our model by changing the window size of attention masking.

Method	Text length	Attention error rate
with a window size of 200	1 min	0/200
	3 min	4/200
	5 min	21/200
with a window size of 100	1 min	0/200
	3 min	0/200
	5 min	3/200
with a window size of 50	1 min	0/200
	3 min	0/200
	5 min	2/200
with a window size of 6	1 min	0/200
	3 min	1/200
	5 min	4/200

TABLE 4. Comparison of the MOSs by applying the attention used in other models to the proposed model.

System	MOS
Tacotron	3.42 ± 0.37
Tacotron2	3.98 ± 0.10
DeepVoice3	3.63 ± 0.07
ParaNet	3.49 ± 0.14
MultiSpeech	3.28 ± 0.09
DLTTS	3.94 ± 0.19
Ground truth	4.58 ± 0.05

TABLE 5. Comparison of the MOSs for the window size of attention masking in the proposed model.

System	MOS
with a window size of 200	3.61 ± 0.13
with a window size of 100	3.71 ± 0.24
with a window size of 50	3.94 ± 0.19
with a window size of 6	3.52 ± 0.11

TABLE 6. Hyperparameters used for proposed document-level neural TTS model.

Parameter	Document-level neural TTS
Audio Sample Rate	16000
Reduction Factor r	1
Mel Bands	80
Character Embedding Dim.	512
Encoder Conv. Layers	3
Decoder RNN Dim.	1024
Decoder Prenet Dim.	256
Decoder Dropout Probability	0.1
Stop Token Threshold	0.5
Attention RNN Dim.	1024
Attention Hidden Size	128
Attention Dropout Probability	0.1
Postnet Embedding Dim.	512
Postnet Conv. Layers	5
Learning Rate	0.001
Batch Size	12

speech, the sound quality was measured for sentences of 10 s or less, and the rest for speech of 5 min. All models used MelGAN as a vocoder. It was confirmed that the sound quality of ParaNet was better than the masking used in MultiSpeech, and that the sound quality of the DeepVoice3 masking model was better than that of ParaNet. In addition, as shown in Table 5, from the result of measuring the MOS measured while changing the window size in the proposed

masking, it was confirmed that the MOS measured was the highest at a size of 50, whereas the AER and CER were the lowest. Hence, it was confirmed that the proposed model can synthesize with high accuracy and sound quality similar to those of the Tacotron2 model when synthesizing document-level speech longer than 5 min.

IV. CONCLUSION

In this paper, we proposed a document-level neural TTS model that synthesizes document-level text into speech. Recently in [12], GMM attention and location-sensitive attention, which were previously used, were modified to create a model capable of synthesizing speech of 80 s by training only about 10 s of speech. However, we achieved better results in a slightly different way through the proposed model. First, short sentences were trained through curriculum learning, and then the proposed model learned longer sentences to become robust with document-level text. If the sentence to be synthesized exceeded a certain length, attention masking was used to generate the mel spectrogram while considering only the essential parts of the alignment. From the experiments, we found the exact part of the alignment needed to generate the mel spectrogram in a given time step by taking the teacher alignment from the pre-trained Tacotron2 model so that the length of the mel spectrogram could be estimated by looking only at the text. Moreover, a result of the experiment using the LJSpeech dataset, we confirmed that the proposed document-level neural TTS model could synthesize a much longer sentence (more than 5 min) with lower attention error, and character error rates and higher quality than that achieved using the existing model. The proposed model took about 50 s to generate a speech over 5 min. To overcome this limitation, we will study a model that synthesizes document-level speech in real-time by applying the proposed model to a non-autoregressive model.

REFERENCES

[1] A. J. Hunt and A. W. Black, "Unit selection in a concatenative speech synthesis system using a large speech database," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. Conf.*, May 1996, pp. 373–376.
[2] A. W. Black and P. Taylor, "Automatically clustering similar units for unit selection in speech synthesis," in *Proc. Eurospeech*, Sep. 1997, pp. 601–604.
[3] K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi, and T. Kitamura, "Speech parameter generation algorithms for HMM-based speech synthesis," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Jun. 2000, p. 1315.
[4] H. Zen, K. Tokuda, and A. W. Black, "Statistical parametric speech synthesis," *Speech Commun.*, vol. 51, p. 1039–1064, Nov. 2009.
[5] H. Ze, A. Senior, and M. Schuster, "Statistical parametric speech synthesis using deep neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, May 2013, pp. 7962–7966.
[6] K. Tokuda, Y. Nankaku, T. Toda, H. Zen, J. Yamagishi, and K. Oura, "Speech synthesis based on hidden Markov models," *Proc. IEEE*, vol. 101, no. 5, pp. 1234–1252, May 2013.
[7] Y. Wang, R. J. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio, Q. Le, Y. Agiomyrgiannakis, R. Clark, and R. A. Saurous, "Tacotron: Towards end-to-end speech synthesis," in *Proc. Interspeech*, Aug. 2017, pp. 4006–4010.
[8] D. Griffin and J. Lim, "Signal estimation from modified short-time Fourier transform," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-32, no. 2, pp. 236–243, Apr. 1984.
[9] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. Skerrv-Ryan, R. A. Saurous, Y. Agiomvrgiannakis, and Y. Wu, "Natural TTS synthesis by conditioning wavenet on MEL spectrogram predictions," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2018, pp. 4779–4783.
[10] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *Proc. ICLR*, Sep. 2015, pp. 1–15.
[11] J. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech recognition," in *Proc. 28th Int. Conf. Neural Inf. Process. Syst.*, vol. 1, Jun. 2015, pp. 577–585.
[12] E. Battenberg, R. J. Skerry-Ryan, S. Mariooryad, D. Stanton, D. Kao, M. Shannon, and T. Bagby, "Location-relative attention mechanisms for robust long-form speech synthesis," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2020, pp. 6194–6198.
[13] A. Graves, "Generating sequences with recurrent neural networks," 2013, *arXiv:1308.0850*. [Online]. Available: https://arxiv.org/abs/1308.0850
[14] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, "Curriculum learning," in *Proc. 26th Annu. Int. Conf. Mach. Learn.*, Jan. 2009, pp. 41–48.
[15] W. Ping, K. Peng, A. Gibiansky, S. O. Arik, A. Kannan, S. Narang, J. Raiman, and J. Miller, "Deep voice 3: Scaling text-to-speech with convolutional sequence learning," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, vol. 79, 2018, pp. 1094–1099.
[16] K. Peng, W. Ping, Z. Song, and K. Zhao, "Non-autoregressive neural text-to-speech," in *Proc. Int. Conf. Mach. Learn.*, Jun. 2020, pp. 7586–7598.
[17] M. Chen, X. Tan, Y. Ren, J. Xu, H. Sun, S. Zhao, T. Qin, and T.-Y. Liu, "MultiSpeech: Multi-speaker text to speech with transformer," 2020, *arXiv:2006.04664*. [Online]. Available: http://arxiv.org/abs/2006.04664
[18] Y. Ren, Y. Ruan, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T. Y. Liu, "Fastspeech: Fast, robust and controllable text to speech," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, May 2019, pp. 3171–3180.
[19] K. Ito, "The lj speech dataset," Tech. Rep., 2017.
[20] K. Kumar, R. Kumar, T. de Boissiere, L. Gestin, W. Z. Teoh, J. Sotelo, A. de Brebisson, Y. Bengio, and A. C. Courville, "Melgan: Generative adversarial networks for conditional waveform synthesis," in *Proc. Neural Inf. Process. Syst.*, 2019, pp. 14881–14892.

SUNG-WOONG HWANG received the B.S. degree in mechanical engineering from Hanyang University, Seoul, South Korea, in 2018, where he is currently pursuing the M.S. degree in electronics and computer engineering. His research interest includes speech synthesis.

JOON-HYUK CHANG (Senior Member, IEEE) received the B.S. degree in electronics engineering from Kyungpook National University, Daegu, South Korea, in 1998, and the M.S. and Ph.D. degrees in electrical engineering from Seoul National University, South Korea, in 2000 and 2004, respectively. From 2000 to 2005, he was with Netdus Corporation, Seoul, as CTO. From 2004 to 2005, he held a postdoctoral position with the University of California at Santa Barbara, Santa Barbara, where he was involved in adaptive signal processing and audio coding. In 2005, he joined the Korea Institute of Science and Technology, Seoul, as a Research Scientist, where he was involved in speech recognition. From 2005 to 2011, he was an Assistant Professor with the School of Electronic Engineering, Inha University, Incheon, South Korea. He is currently a Full Professor with the School of Electronic Engineering, Hanyang University, Seoul, South Korea. His research interests include speech recognition, deep/machine learning, artificial intelligence (AI), speech processing, acoustic signal processing, and bio-medical signal processing. He was a recipient of the IEEE/IEEK IT Young Engineer of the Year, in 2011. He is serving on the Editorial Board of *Digital Signal Processing* (Elsevier).