ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 978-1-7281-7605-5/20/$31.00 ©2021 IEEE | DOI: 10.1109/ICASSP39728.2021.9413679

JOINT ALIGNMENT LEARNING-ATTENTION BASED MODEL FOR GRAPHEME-TO-PHONEME CONVERSION

Yonghe Wang, Feilong Bao, Hui Zhang, Guanglai Gao*

College of Computer Science, Inner Mongolia University, Hohhot, China
Inner Mongolia Key Laboratory of Mongolian Information Processing Technology, Hohhot, China
cswyh92@163.com, {csfeilong,cszh,csggl}@imu.edu.cn

ABSTRACT

Sequence-to-sequence attention-based models for grapheme-to-phoneme (G2P) conversion have gained significant interests. The attention-based encoder-decoder framework learns the mapping of input to output tokens by selectively focusing on relevant information, and has been shown well performance. However, the attention mechanism can result in non-monotonic alignments, resulting in poor G2P conversion performance. In this paper, we present a novel approach to optimize the G2P conversion model directly alignment grapheme-phoneme sequence by using alignment learning (AL) as the loss function. Besides, we propose a multi-task learning method that uses a joint alignment learning model and attention model to predict the proper alignments and thus improve the accuracy of G2P conversion. Evaluations on Mongolian and CMUDict tasks show that alignment learning as the loss function can effectively train G2P conversion model. Further, our multi-task method can significantly outperform both the alignment learning-based model and attention-based model.

Index Terms— grapheme-to-phoneme conversion, alignment learning, attention, multitask learning

1. INTRODUCTION

Grapheme-to-phoneme conversion (G2P) is the task of converting the spelling of a word (a grapheme sequence) to its phonetic transcription (a phoneme sequence) (e.g. STUDY → S T AH D IY). It is a crucial component in automatic speech recognition (ASR) and text-to-speech (TTS) systems. Manually creating and maintaining an expert-crafted pronunciation dictionary is a tedious task. Therefore, phoneme lexicons created by experts are finite, and they are used to train a G2P model that can automatically handle new words.

G2P conversion has been studied for a long time. The early research of G2P system was rule-based, but linguistic expertise is required to establish rules, and these rules are difficult to cover most possible situations [1, 2]. Subsequently, to solve the rule-based method cannot effectively convert out of vocabulary (OOV) words, joint sequence models for G2P conversion were researched [3, 4]. In [4], the publicly available tool Sequitur is based on the initial grapheme-phoneme sequence alignment to further calculates the joint n-gram language model and observed dramatically improved performance.

Recently, sequence-to-sequence learning based on encoder-decoder model has been successfully applied to a wide range of tasks, including neural machine translation [5–7], automatic speech recognition [8–10], text-to-speech [11–13], phrase break prediction [14, 15], and grapheme-to-phoneme conversion [16–18]. Such models learn a direct mapping between the variable lengths of sequence pairs, and directly transcribes the graphemes to phonemes without requiring predefined alignment.

For the traditional encoder-decoder architecture, the encoder module receives the source data features and transforms it into an intermediate representation and the decoder module for generating conversion output. At each training time step, given the current input of the grapheme sequence, the model produces a phoneme in the current time-step with maximal probability conditioned on the ground-truth of the phoneme sequences in the previous time-steps. In this process, the model cannot learn the alignment between grapheme-phoneme sequences. In [19], the encoder-decoder model combined with attention mechanism was investigated for the G2P task and obtained an effective performance improvement without explicit alignment. In [20], the combination of sequitur G2P and Multi-language trained seq2seq-attention can further improve performance. Although the attention mechanism can selectively focus on relevant information. However, G2P is clearly a sequence classification task with a monotonic input-output relationship. The attention mechanism can result in non-monotonic alignment, which makes it difficult for attention-based models to estimate the proper alignment.

To overcome the above incomplete alignment problem, this paper first proposes a novel G2P conversion method based on alignment learning (AL) criterion. Inspired by a joint CTC-attention model within the multi-task learning method of the end-to-end ASR system [21]. A novel sequence-to-sequence G2P conversion method is intended to improve performance by using a joint alignment learning-attention based encoder-decoder model within the multi-task learning (MTL) framework. The core of our proposed method is to train the shared-encoder network by using the alignment learning criterion and the attention model objective simultaneously, and obtain the advanced alignment representation. The alignment learning objective criterion can effectively compensate for the problem that the attention model is difficult to train the encoder network with proper alignments. Experiments are conducted in Mongolian and English dataset. The results show that the proposed method can effectively improve G2P conversion performance.

2. PROPOSED MODEL

In this section, we introduce our proposed approach that joint alignment learning-attention based sequence-to-sequence G2P conversion model. We first review the attention-based encoder-decoder G2P conversion method in Section 2.1 and describe the proposed alignment learning optimization method in Section 2.2. Then, our joint alignment learning-attention based G2P conversion framework will be described in Section 2.3.

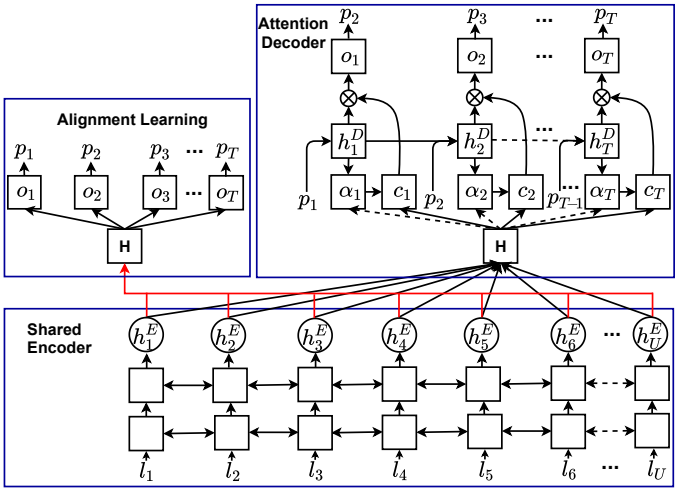

Fig. 1: The proposed joint alignment learning-attention based end-to-end model within the multi-task learning framework.

2.1. Attention-based encoder-decoder model

For the G2P conversion approach of the attention-based encoder-decoder architecture, we consider an input $\boldsymbol{l} = l_1, ..., l_U$ is the source sequence of grapheme with length U, and $\boldsymbol{p} = p_1, ..., p_T$ is the target sequence of phoneme with length T. The model directly model the conditional probability of $P(\boldsymbol{p}|\boldsymbol{l})$. Specifically, given both the grapheme sequence $\boldsymbol{l}$ and the previous inference labels $p_{1:t-1}$, the model emits the posterior probability of each label sequence at t conditioning:

$$p(\boldsymbol{p}|\boldsymbol{l}) = \prod_t p(p_t|\boldsymbol{l}, p_{1:t-1}). \tag{1}$$

In this process, the encoder subnetwork implemented as a multi-layer bidirectional RNN (such as LSTM) transforms the input grapheme sequence $\boldsymbol{l}$ into a high-level representation $\boldsymbol{h} = \{h_u\}_{u=1}^{U}$:

$$\boldsymbol{h} = \text{Encoder}(\boldsymbol{l}). \tag{2}$$

Then, the attention-based decoder takes $\boldsymbol{h}$ and all previously seen labels $p_{1:t-1}$ as input, producing the probability distribution of the label p_t:

$$p(p_t|\boldsymbol{l}, p_{1:t-1}) = \text{Decoder}(\boldsymbol{h}, p_{1:t-1}). \tag{3}$$

Specifically, at each decoder time step t, the posterior distribution of the predicted output is generated from the cascade of decoder states s_t and context vector c_t. The s_t is calculated by a 1-layer of unidirectional LSTM. The c_t is the context information produced by the attention module based on the hidden state of the encoder and decoder. We summarize this processing pipeline as follows:

$$c_t = \sum_{u=1}^{U} \alpha_{t,u} h_u^{enc} \tag{4}$$

$$\alpha_{t,u} = \text{Align}(h_u^{enc}, h_t^{dec}) = \frac{exp(\beta(h_u^{enc}, h_t^{dec}))}{\sum_{u=1}^{U} exp(\beta(h_u^{enc}, h_t^{dec}))} \tag{5}$$

where $\beta(\cdot)$ is the score function of the attention module, which can be in the form of dot, general, concat, or perception.

In the model training, the decoder completes hypothesis generation when end-of-sentence($<sos>$) is emitted. The loss function of the encoder-decoder model is trained using the cross-entropy (CE) criterion:

$$\mathcal{L}_{Attention} = -\ln P(\boldsymbol{p}^*|\boldsymbol{l}) = -\sum_t \ln P(p_t^*|\boldsymbol{l}, p_{1:t-1}^*) \tag{6}$$

where $p_{1:t-1}^*$ is the ground truth of the previous labels.

2.2. Alignment Learning G2P Conversion Model

We consider that G2P conversion is a sequence alignment optimization task. Give the input sequence of grapheme $\boldsymbol{l}$ with length U and the target sequence of phoneme $\boldsymbol{p}$ with length T. The critical challenge of G2P conversion is that the length of the input sequence and the target sequence may not be the same, and there is no a prior way of aligning them. We propose to address this problem by defining a distance function $d(l_i, p_j)$, which is used measure the cost of aligning the grapheme l_i with the phoneme p_j. And use alignment learning as the loss function to minimize the overall distance cost matrix to find the best possible alignment.

Corresponding to the distance cost matrix is the alignment matrix. We define $\mathcal{D} \subset \{0, 1\}^{U \times T}$ to be the set of possible binary alignment matrices. For the alignment matrix $D, \forall D \in \mathcal{D}, D_{ij} = 1$ if l_i is labeled as $p_j^{'}$ and $D_{ij} = 0$ otherwise. The purpose of G2P alignment is to align each grapheme with a single phoneme, so we need to impose rigorous constraints on the eligible warping path to ensure the $\boldsymbol{p}$ order is strictly consistent with the $\boldsymbol{l}$ timeline. After getting an alignment matrix D, we can derive its corresponding label $p_{1:T}$ as: $p_i = p_j^{'}$, if $D_{ij} = 1$.

Given a set of eligible alignments $\mathcal{D}$, the goal of AL is to consider the cost of all possible alignment matrices to find the best alignment $D^* \in \mathcal{D}$:

$$D^* = \arg\max_{D \in \mathcal{D}} \langle D, \Delta(\boldsymbol{l}, \boldsymbol{p}) \rangle , \tag{7}$$

where $\Delta(\boldsymbol{l}, \boldsymbol{p}) := [\delta(l_i, p_j)]_{ij} \in \mathbb{R}^{U \times T}$ is the cost matrix between input graphemes $\boldsymbol{l}$ and target phonemes $\boldsymbol{p}$. $\langle D, \Delta(\boldsymbol{l}, \boldsymbol{p}) \rangle$ is the inner product between the eligible alignment matrix D and the cost matrix $\Delta(\boldsymbol{l}, \boldsymbol{p})$.

The distance cost matrix values $\Delta(\boldsymbol{l}, \boldsymbol{p})$ in equation (7) are the negative log-probabilities of the model output corresponding to each $(\boldsymbol{l}, \boldsymbol{p})$ index. Therefore, among many possible alignments, the training goal of the AL criterion is to optimize the model to minimize the cost of the optimal alignment.

$$\mathcal{L}_{\text{AL}} = \min_\gamma \{\langle D, \Delta(\boldsymbol{l}, \boldsymbol{p}) \rangle, D \in \mathcal{D}\}, \tag{8}$$

where the generalized $\min_\gamma\{\}$ operator is formulated as equation (9) with a hyper-parameter $\gamma \leqslant 1$:

$$\min_\gamma\{a_1, ..., a_n\} := \begin{cases} \min_{i \leqslant n} a_i, & \gamma = 0, \\ -\gamma \log \sum_{i=1}^{n} e^{-a_i/\gamma}, & \gamma > 0. \end{cases} \tag{9}$$

the higher value of the hyper-parameter γ means that more non-minimum values are mixed into the output.

2.3. Joint Alignment Learning-Attention using MTL

In Section 2.2, we proposed the G2P conversion model using alignment learning as the loss function, which can effectively perform monotonic alignment of input graphemes and output

phonemes. Meanwhile, in Section 2.1, the sequence-to-sequence model based on the attention mechanism can selectively focus on the related input tokens when generating each output token. Taking these characteristics into account, we propose to train the shared-encoder network using both alignment learning objective and attention model objective simultaneously within the multi-task learning (MTL) framework.

Fig.1 illustrates the overall architecture of our proposed framework. The network includes a shared encoder subnetwork, whose outputs correspond to alignment learning model and attention model, respectively. Our proposed joint training method can efficiently enforce the desired monotonic alignment without an initial rough grapheme-phoneme sequence alignment. The objective function of MTL is defined as a weighted sum of the losses propagated from both AL and attention model:

$$\mathcal{L}_{MTL} = \lambda \mathcal{L}_{AL} + (1-\lambda)\mathcal{L}_{Attention} \tag{10}$$

where hyper-parameter λ is weight smoothing factor, and $0 \leqslant \lambda \leqslant 1$.

3. EXPERIMENTS

3.1. Datasets

We carry out experiments on Mongolian and English G2P conversion tasks to verify the performance of the proposed method. For Mongolian, the dictionary contains about 39K manual pronunciation entries, we randomly selected 5% and 10% of the full vocabulary for development and testing, and used the other 85% for training. There are 43 graphemes set containing 26 lowercase alphabet symbols, 12 uppercase alphabet symbols, 5 control symbols (e.g., hyphen (-), underline (_), etc.). The phonemes in this dataset contain 57. For English, we use CMUDict pronunciation 0.7b [1] with about 138k manual pronunciation entries using the ARPAbet phoneme set. We use the sequence-to-sequence G2P example of the CNTK toolkit2 [2] to split the dataset into training, development and test sets in the same way, including a 108,952-word training set and a 12,855-word test set. 5,447 words are used as development set to determine stopping criteria while training. There are 27 graphemes (uppercase alphabet symbols plus the apostrophe) and 41 phonemes in this dataset.

We use $<sos>$ and $<eos>$ tokens as beginning-of-graphemes (beginning-of-phonemes) and end-of-graphemes (end-of-phonemes) tokens in both datasets. For inference, the decoder uses the past phoneme sequence to predict the next phoneme, and it stops predicting after token $<eos>$.

3.2. Evaluation

The G2P results are performed in terms of word error rate (WER) and phoneme error rate (PER). WER is calculated as the total number of mispredicted words divided by the total number of words. PER is calculated as a normalized minimum edit distance on phonemes between the conversion output sequence and ground truth sequence. Edit distance is also known as Levenshtein distance, which is calculated using a dynamic programming algorithm. Both WER and PER are in the percentage format. For words with multiple real pronunciations, following [4, 16, 19], we choose the prediction output that results in the lowest PER to calculate the overall PER results. Moreover, word error is calculated only if the predicted pronunciation does not match any real pronunciations.

[1] http://www.speech.cs.cmu.edu/cgi-bin/CMUDict

[2] https://github.com/Microsoft/CNTK/tree/master/Examples/SequenceToSequence/CMUDict/Data

3.3. Training

For the alignment learning-based methods. We use an encoder network for modeling, which is a multi-layers Bidirectional Long Short-Term Memory LSTMs (Bi-LSTM). Dropout operation is applied to the output of each Bi-LSTM layer by a rate of 30%. We used a 128-dimensional embedding matrix to transform the input graphemes into a continuous vector as encoder input. The embeddings are trained as part of the model training.

For attention-based sequence-to-sequence and multi-task learning methods. The encoder network is the same as the alignment learning-based method. On the decoder side, we used a 128-dimensional embedding matrix to transform the input phonemes into a continuous vector, followed by one-unidirectional LSTMs. For the scorer function inside the attention module, we used MLP scorers and the number of hidden units is the same as the decoder LSTM.

In the training phase, all of the model parameters were initialized randomly as uniform(-0.1,0.1). We train for 100 epochs using Adam algorithm with gradient clipping [22]. A simple learning rate schedule is employed, we start with a learning rate of 0.0005, after 20 epochs, the learning rate gets decayed by a factor of 0.9 when the model does not improve over the validation data. Our mini-batch size is set to 32 for all tasks. For our alignment learning-based method, we tried four scenarios using different discount factors $\gamma = \{1, 0.1, 0.01, 0.001\}$. For our MTL, we tested three different task weights, λ: 0.8, 0.5, and 0.2. Our framework is implemented with the PyTorch [23].

3.4. Inference

For inferring of the attention and MTL models, we use the simple greedy approach to report the 1st pass results directly. That is, we infer the first phoneme by the argmax of the softmax output. Then, the inferred phoneme is fed back into the G2P model to generate the next phoneme. This process is continued until the end token $<eos>$ is output or the maximum number of inferences is reached. For inferring of alignment learning-based model, we took the sequence of most likely outputs. That is, at each inference time step, we consider the output phone to be the highest probability output. Beam search was not applied in this work.

3.5. Results

Table 1 gives the PER and WER results obtained with our alignment learning-based G2P models, the AL criterion hyper-parameter γ is set to 1 for all experiments. We compared the performance of different model structures with various components. As can be seen from the table, for the CMUDict dataset, when the model has the same number of hidden layers, a large number of biLSTM hidden units are crucial. Meanwhile, a smaller number of stacked layers can cause performance loss. The best performance is obtained for the 3-layer stacked LSTM with 512 hidden units and can achieve 7.08% on the PER and 30.49% on WER. For the Mongolian dataset, a large number of biLSTM hidden units can also improve model performance. However, the high number of hidden layers sometimes does not result in better performance, such as AL-biLSTM (512x3) didn't exceed AL-biLSTM (512x2). The reason is that due to the small amount of Mongolian training data, data sparseness would be triggered when the model parameters are large, which would cause

Table 1: The PER and WER of alignment learning-based G2P models. Results shown here are for γ=1.

Data	Method	PER	WER
Mongolian	AL-biLSTM (256x3)	3.16	13.79
	AL-biLSTM (256x4)	2.91	13.00
	AL-biLSTM (512x2)	**2.56**	**11.55**
	AL-biLSTM (512x3)	2.80	12.57
CMUDict	AL-biLSTM (256x3)	7.44	32.15
	AL-biLSTM (256x4)	7.22	30.92
	AL-biLSTM (512x2)	7.31	31.64
	AL-biLSTM (512x3)	**7.08**	**30.49**

Table 2: The results of alignment learning-based G2P model trained with different hyper-parameters γ.

Data	Method	PER	WER
Mongolian	AL-biLSTM (512x2, γ=0.1)	1.81	8.51
	AL-biLSTM (512x2, γ=0.01)	**1.74**	**8.13**
	AL-biLSTM (512x2, γ=0.001)	1.82	8.51
CMUDict	AL-biLSTM (512x3, γ=0.1)	6.44	29.19
	AL-biLSTM (512x3, γ=0.01)	**6.36**	**28.83**
	AL-biLSTM (512x3, γ=0.001)	6.62	29.61

Table 3: PER and WER results from alignment learning-based model, attention-based model, and MTL with (λ = 0.8, 0.5, 0.2).

Data	Method	PER	WER
Mongolian	AL-biLSTM (256x3)	1.91	11.29
	seq2seq-attn (256x3)	1.87	7.97
	MTL (256x3, λ=0.8)	1.84	7.14
	MTL (256x3, λ=0.5)	1.65	6.93
	MTL (256x3, λ=0.2)	**1.60**	**6.53**
	AL-biLSTM (512x2)	1.74	8.13
	seq2seq-attn (512x2)	1.75	6.81
	MTL (512x2, λ=0.8)	1.63	6.75
	MTL (512x2, λ=0.5)	1.58	6.30
	MTL (512x2, λ=0.2)	**1.50**	**6.20**
CMUDict	AL-biLSTM (256x3)	6.80	30.47
	seq2seq-attn (256x3)	6.21	26.05
	MTL (256x3, λ=0.8)	5.85	25.54
	MTL (256x3, λ=0.5)	5.74	24.51
	MTL (256x3, λ=0.2)	**5.55**	**24.05**
	AL-biLSTM (512x3)	6.36	28.83
	seq2seq-attn (512x3)	6.19	24.91
	MTL (512x3, λ=0.8)	5.49	23.71
	MTL (512x3, λ=0.5)	5.35	23.23
	MTL (512x3, λ=0.2)	**5.26**	**22.96**

the model cannot learn enough information. The best performance is obtained using the AL-biLSTM (512x2) system. It achieved 2.56% PER and 11.55% WER. Overall, the experimental results indicate that alignment learning as the loss function can effectively train the G2P conversion model, making the input phonemes and output phonemes are monotonically aligned.

We also explore several configurations by using AL criterion with different $\gamma \in \{0.1, 0.01, 0.001\}$ values. We conduct experiments using the model of the best performance in Table 1. As can be seen from Table 1 and 2, the different γ leads to different performance of the model. Compared to the highest critical value γ =1, all other values give better results. Furthermore, we found that the hyper-parameter factor γ = 0.01 gives the best performance on both datasets. On Mongolian datasets, the PER is reduced from 2.56% to 1.74% (relative 32%), and the WER is reduced from 11.55% to 8.13% (relative 29.6%). On CMUDict datasets, the PER is reduced from 7.08% to 6.32% (relative 10.2%), and the WER is reduced from 30.49% to 28.83% (relative 5.3%). In the rest of the experiment, we fixed the hyper-parameter factor γ of the AL criterion to 0.01.

Table. 3 shows the PER and WER comparison of different models on the Mongolian and CMUDict corpus. To verify the generalization performance of our proposed method, we conducted experiments on the best and worst models in Table 1. Our proposed MTL model shows consistent behavior, the G2P conversion accuracy is significantly outperformed both alignment learning-based model and attention-based model. We observed that our joint alignment learning-attention achieved the best performance when using λ = 0.2 on both Mongolian and CMUDict tasks. Comparison of AL-biLSTM (512x2) series of experiments on Mongolian dataset, the proposed MTL (512x2, λ=0.2) model achieves 9.0% relative WER reduction compared to the seq2seq-attn (512x2) model, and achieves 23.7% relative WER reduction compared to the AL-biLSTM (512x2) model. Similarly, on CMUDict datasets, the proposed MTL (512x3, λ=0.2) model achieves 7.8% relative WER reduction compared to the seq2seq-attn (512x3) model, and achieves 20.4% relative WER reduction compared to the AL-biLSTM (512x3) model.

4. CONCLUSIONS

In this work, we have proposed a novel approach to optimize end-to-end G2P conversion systems directly alignment graphemes-phonemes sequence by using alignment learning (AL) as the loss function. The AL-based loss function minimizes all the costs spanned by all possible alignments between the two-time series of graphemes and phonemes, and the best alignment model means the optimal conversion system. We also proposed joint training of an alignment learning-based model with an attention-based model using the multi-task learning approach, which shared an encoder subnetwork during the training phase. On two benchmark corpora, Mongolian and CMUDict, proposed alignment learning as the loss function can effectively train the G2P conversion model. Further, our MTL method can significantly outperform both the alignment learning-based model and attention-based model. The best system achieved up to 1.50% PER and 6.20% WER in Mongolian, and 5.26% PER and 22.96% WER in CMUDict.

5. ACKNOWLEDGMENTS

This research is supported by the National Key Research and Development Program of China (No.2018YFE0122900), China National Natural Science Foundation (No.61773224, No.62066033), Inner Mongolia Natural Science Foundation (No.2018MS06006), Achievement Transformation Program of Inner Mongolia Autonomous Region (No.CGZH2018125) and Applied Technology Research and Development Program of Inner Mongolia Autonomous Region (No.2019GG372, No.2020GG0046).

6. REFERENCES

[1] Honey S. Elovitz, Rodney Johnson, Astrid McHugh, and John E. Shore, "Letter-to-sound rules for automatic translation of english text to phonetics," *IEEE Transactions on Acoustics Speech Signal Processing*, vol. 24, no. 6, pp. 446–459, 1977.

[2] Marc Schrder, "Issues in building general letter to sound rules," in *International Speech Communication Association*, 2018, pp. 77–80.

[3] Lucian Galescu and James F Allen, "Pronunciation of proper names with a joint n-gram model for bi-directional grapheme-to-phoneme conversion.," in *International Conference on Spoken Language Processing*, 2002, pp. 109–112.

[4] Maximilian Bisani and Hermann Ney, "Joint-sequence models for grapheme-to-phoneme conversion," *Speech Communication*, vol. 50, no. 5, pp. 434–451, 2008.

[5] Ilya Sutskever, Oriol Vinyals, and Quoc V Le, "Sequence to sequence learning with neural networks," in *Advances in neural information processing systems*, 2014, pp. 3104–3112.

[6] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio, "Neural machine translation by jointly learning to align and translate," in *International Conference on Learning Representations*, 2015.

[7] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al., "Google's neural machine translation system: Bridging the gap between human and machine translation," *arXiv preprint arXiv:1609.08144*, 2016.

[8] Alex Graves and Navdeep Jaitly, "Towards end-to-end speech recognition with recurrent neural networks," in *International Conference on Machine Learning*, 2014, pp. 1764–1772.

[9] William Chan, Navdeep Jaitly, Quoc Le, and Oriol Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 4960–4964.

[10] Chung-Cheng Chiu, Tara N Sainath, Yonghui Wu, Rohit Prabhavalkar, Patrick Nguyen, Zhifeng Chen, Anjuli Kannan, Ron J Weiss, Kanishka Rao, Ekaterina Gonina, et al., "State-of-the-art speech recognition with sequence-to-sequence models," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 4774–4778.

[11] Yuxuan Wang, R J Skerryryan, Daisy Stanton, Yonghui Wu, Ron Weiss, Navdeep Jaitly, Zongheng Yang, Ying Xiao, Zhifeng Chen, Samy Bengio, et al., "Tacotron: Towards end-to-end speech synthesis," in *INTERSPEECH 2017 18th Annual Conference of the International Speech Communication Association*. IEEE, 2017, pp. 4006–4010.

[12] Rui Liu, Berrak Sisman, Jingdong Li, Feilong Bao, Guanglai Gao, and Haizhou Li, "Teacher-student training for robust tacotron-based tts," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6274–6278.

[13] Rui Liu, Berrak Sisman, Feilong Bao, Guanglai Gao, and Haizhou Li, "Modeling prosodic phrasing with multi-task learning in tacotron-based tts," *IEEE Signal Processing Letters*, vol. 27, pp. 1470–1474, 2020.

[14] Rui Liu, Feilong Bao, Guanglai Gao, Hui Zhang, and Yonghe Wang, "Improving mongolian phrase break prediction by using syllable and morphological embeddings with bilstm model.," in *INTERSPEECH 2018 19th Annual Conference of the International Speech Communication Association*, 2018, pp. 57–61.

[15] Rui Liu, Berrak Sisman, Feilong Bao, Jichen Yang, Guanglai Gao, and Haizhou Li, "Exploiting morphological and phonological features to improve prosodic phrasing for mongolian speech synthesis," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 274–285, 2021.

[16] Kaisheng Yao and Geoffrey Zweig, "Sequence-to-sequence neural net models for grapheme-to-phoneme conversion," in *INTERSPEECH 2015 16th Annual Conference of the International Speech Communication Association*, 2015, pp. 3330–3334.

[17] Moonjung Chae, Kyubyong Park, Linhyun Bang, Soobin Suh, Longhyuk Park, Namju Kimt, and Longhun Park, "Convolutional sequence to sequence model with non-sequential greedy decoding for grapheme to phoneme conversion," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 2486–2490.

[18] Sevinj Yolchuyeva, Géza Németh, and Bálint Gyires-Tóth, "Grapheme-to-phoneme conversion with convolutional neural networks," *Applied Sciences*, vol. 9, no. 6, pp. 1143, 2019.

[19] Shubham Toshniwal and Karen Livescu, "Jointly learning to align and convert graphemes to phonemes with neural attention models," in *2016 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2016, pp. 76–82.

[20] Benjamin Milde, Christoph Schmidt, and Joachim Kohler, "Multitask sequence-to-sequence models for grapheme-to-phoneme conversion.," in *INTERSPEECH 2017 18th Annual Conference of the International Speech Communication Association*, 2017, pp. 2536–2540.

[21] Suyoun Kim, Takaaki Hori, and Shinji Watanabe, "Joint ctc-attention based end-to-end speech recognition using multi-task learning," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 4835–4839.

[22] Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio, "On the difficulty of training recurrent neural networks," in *International conference on machine learning*, 2013, pp. 1310–1318.

[23] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer, "Automatic differentiation in pytorch," in *Conference and Workshop on Neural Information Processing Systems (NIPS)*, 2017.