Mongolian Grapheme to Phoneme Conversion by Using Hybrid Approach

Zhinan Liu[1], Feilong Bao[1(✉)], Guanglai Gao[1], and Suburi[2]

[1] College of Computer Science,
Inner Mongolia University, Huhhot 010021, China
lzn_bung@163.com, {csfeilong, csggl}@imu.edu.cn
[2] Inner Mongolia Public Security Department, Huhhot 010021, China
sunbuer@163.com

Abstract. Grapheme to phoneme (G2P) conversion is the assignment of converting word to its pronunciation. It has important applications in text-to-speech (TTS), speech recognition and sounds-like queries in textual databases. In this paper, we present the first application of sequence-to-sequence (Seq2Seq) Long Short-Term Memory (LSTM) model with the attention mechanism for Mongolian G2P conversion. Furthermore, we propose a novel hybrid approach of combining rules with Seq2Seq LSTM model for Mongolian G2P conversion, and implement the Mongolian G2P conversion system. The experimental results show that: Adopting Seq2Seq LSTM model can obtain better performance than traditional methods of Mongolian G2P conversion, and the hybrid approach further improves G2P conversion performance. The word error rate (WER) relatively reduces by 10.8% and the phoneme error rate (PER) approximately reduces by 1.6% through comparing with the Mongolian G2P conversion method being used based on the joint-sequence models, which completely meets the practical requirements of Mongolian G2P conversion.

Keywords: Mongolian · Grapheme-to-phoneme · Sequence-to-sequence LSTM

1 Introduction

Grapheme-to-phoneme conversion (G2P) refers to the task of converting a word from the orthographic form (sequence of letters/characters/graphemes) to its pronunciation (a sequence of phonemes). It has a wide range of applications in speech synthesis [1–3], automatic speech recognition (ASR) [4–6] and speech retrieval [7, 8].

One of the challenges in G2P conversion is that the pronunciation of any grapheme depends on a variety of factors including its context and the etymology of the word. Another complication is that output phone sequence can be either shorter than or longer than the input grapheme sequence. Typical approaches to G2P involve using rule-based methods and joint-sequence models. While rule-based methods are effective to handle new words, they have some limitations: designing the rules is hard and requires specific linguistic skills, and it is extremely difficult to capture all rules for natural languages. To overcome the above limitations, another called statistics-based method are proposed,

M. Zhang et al. (Eds.): NLPCC 2018, LNAI 11108, pp. 40–50, 2018.
https://doi.org/10.1007/978-3-319-99495-6_4

in which joint-sequence models are well performing and popular. In joint-sequence models, the alignment is provided via some external aligner [9–11]. However, since the alignment is a latent variable—a means to an end rather than the end itself, it is worthy to consider whether we can do away with such explicit alignment.

In recent years, some work on the G2P problem has used neural network-based approaches. Specifically, long short-term memory (LSTM) networks have recently been explored [12]. LSTMs (and, more generally, recurrent neural networks) can model varying contexts ("memory") and have been successful for a number of sequence prediction tasks. When used in a sequence-to-sequence (Seq2Seq) model, as in [13], which includes an encoder RNN and a decoder RNN, the encoder RNN encoder input sequence token by token, they in principle require no explicit alignments between the input (grapheme sequence) and output (phoneme sequence), as the model is trained in an end-to-end fashion. Bahdanau et al. [14] proposed a related model with an attention mechanism for translation that makes the model better, and Toshniwal et al. [15] introduce an attention mechanism and improve performance of G2P conversion.

For Mongolian G2P conversion, Bao et al. [16] proposed a rule-based method and the method based on joint-sequence model, where the latter method showed better performance than the former method. However, performance of current Mongolian G2P conversion is inferior to other languages. In this paper, we first introduce a Seq2Seq LSTM model with attention mechanism, which proved is useful in other sequence prediction tasks. We obtain better performance than traditional methods for Mongolian G2P conversion. Seq2Seq LSTM model is generative language model, conditioned on an input sequence, the model using an attention mechanism over the encoder LSTM states will not overfit and generalize much better than the plain model without attention mechanism. Taking account of the shortcomings of statistics-based method that can't exactly decode all words in the dictionary and Mongolian characteristics is the majority of Out Of Vocabulary (OOV) words with suffixes connected to the stem using Narrow Non-Break Space (NNBS), we proposed a novel hybrid approach combining of rules with Seq2Seq LSTM model to covert Mongolian word, and we obtain better performance for Mongolian G2P conversion.

In the next section, we will discuss traditional methods for Mongolian G2P conversion. In the remainder we will focus on Seq2Seq LSTM model with an attention mechanism for Mongolian G2P conversion. We will lay the theoretical foundations and undertake a detailed exposition of this model in Sect. 3, and then we will introduce the hybrid approach in Sect. 4. Section 5 presents experimental results demonstrating the better performance of the proposed method, and analyze the consequences of the method. Finally, in Sect. 6 we conclude this paper and look forward to the future of Mongolian G2P conversion technology.

2 Related Work

The Mongolian G2P conversion that was firstly considered in the context of Mongolian text-to-speech (TTS) applications. In this section, we will summarize two traditional approaches to Mongolian G2P conversion.

The one approach is rule-based Mongolian G2P conversion. The written form and spoken form of Mongolian are not one-to-one, and the vowels and consonants may increase, fall off and change. Through in-depth study of Mongolian pronunciation rules, three rules, vowels pronunciation variation rule, consonant binding rule and vowel-harmony rule, are employed in Mongolian G2P conversion. Firstly, Mongolian word is converted by using vowels pronunciation variation rule, and then conversion is followed according to the consonant binding rule, finally, the vowel-harmony rule is used. The rule-based method overcomes the limitations of simple dictionary look-up. However, this method consists of two drawbacks: firstly, designing the rules is hard and requires specific linguistic skills. Mongolian frequently show irregularities, which need to be captured by exceptional rules or exceptional lists. Secondly, the interdependence between rules can be quite complex, so rule designers have to cross-check if the outcome of applying the rules is correct in all cases. This makes development and maintenance of rule systems very tedious in practice. Moreover, a rule-based G2P system is still likely to make mistakes when presented with an exceptional word, not considered by the rule designer.

Another Mongolian G2P conversion approach is based on joint-sequence model. The model needs to find a joint vocabulary of graphemes and phonemes (named graphone) by aligning letters and phonemes, and uses graphone sequence to generate the orthographic form and pronunciation of a Mongolian word. The probability of a graphone sequence is

$$p(C = c_1 \ldots c_T) = \prod\nolimits_{t=1}^{T} p(c_t | c_1 \ldots c_{t-1}) \tag{1}$$

Where each c is a graphone unit. The conditional probability $p(c_t|c_1 \ldots c_{t-1})$ is estimated using an n-gram language model.

To date, this model has produced the better performance on common benchmark datasets. Sequitur G2P is a good established G2P conversion tool using joint-sequence n-gram modelling so that it is very convenient to perform an experiment. In the next section, we will introduce Seq2Seq LSTM model with attention mechanism.

3 Seq2Seq LSTM Model with an Attention Mechanism

Neural Seq2Seq model has recently shown promising results in several tasks, especially translation [17, 18]. Because the G2P problem is in fact largely analogous to the translation problem, with a many-to-many mapping between subsequences of input labels and subsequences of output labels and with potentially long-range dependencies, so this model is also frequently used in G2P conversion [13, 15, 19]. We first apply the Seq2Seq LSTM model with attention mechanism to Mongolian G2P conversion, here, we describe in detail the Seq2Seq LSTM model used in Mongolian G2P conversion.

The Seq2Seq LSTM model follow the LSTM Encoder-Decoder framework [20], the encoder reads the input Mongolian letters sequence, a sequence of vectors $x = (x_1, \ldots, x_2)$, the LSTM computes the $h_1, \ldots, h_T(h_t$ is control state at timestep $t)$ and $m_1, \ldots, m_T(m_t$ is memory state at timestep $t)$ as follows.

$$i_t = sigm(W_1 x_t + W_2 h_{t-1}) \tag{2}$$

$$i_t^{'} = tanh(W_3 x_t + W_4 h_{t-1}) \tag{3}$$

$$f_t = sigm(W_5 x_t + W_6 h_{t-1}) \tag{4}$$

$$o_t = sigm(W_7 x_t + W_8 h_{t-1}) \tag{5}$$

$$m_t = m_{t-1} \odot f_t + i_t \odot i_t^{'} \tag{6}$$

$$h_t = m_t \odot o_t \tag{7}$$

Where the operator $\odot$ represents element-wise multiplication, the matrices $W_1, \ldots, W_8$ and the vector h_0 are the parameters of the model, and all the nonlinearities are computed element-wise. The above equations are merged as:

$$h_t = f(x_t, h_{t-1}) \tag{8}$$

In above equation, f represents an LSTM.

The decoder is another LSTM to produce the output sequence (phonemes sequence y $(y_1, \ldots, y_{T_B})$) and trained to predict the next phoneme y_t given the attention vector c_t and all the previously predicted phonemes sequence $\{y_1, \ldots, y_{t-1}\}$, each conditional probability is modeled as

$$p(y_t | y_1, \ldots, y_{t-1}, x) = g(y_{t-1}, s_t, c_t) \tag{9}$$

Where g is a nonlinear, potentially multi-layered, function that outputs the probability of y_t, and s_t represents the hidden state of the LSTM at timestep t, the attention vector c_t [21] concatenating with s_t became the new hidden state to predict y_t. To computed the attention vector c_t at each output time t over the input Mongolian letters sequence $x(x_1, \ldots, x_{T_A})$ as following:

$$u_i^t = v^T \tanh\left(W_1^{'} h_i + W_2^{'} s_t\right) \tag{10}$$

$$a_i^t = softmax\left(u_i^t\right) \tag{11}$$

$$c_t = \sum\nolimits_{i=1}^{T_A} a_i^t h_i \tag{12}$$

The vector v and matrices $W_1^{'}, W_2^{'}$ are learnable parameters of the model. The vector u_i^t has length T_A and its $i - th$ item contains a score of how much attention should be put on the $i - th$ hidden encoder state h_i. These scores are normalized by softmax to create the attention mask a^t over encoder hidden decoder.

4 Hybrid Approach to Mongolian G2P Conversion

Taking account of the shortcomings of statistics-based method that can't exactly decode all words in the dictionary and Mongolian characteristics is the majority of Out Of Vocabulary (OOV) words with suffixes connected to the stem using Narrow Non-Break Space (NNBS), we proposed a novel hybrid approach of combining rules with Seq2Seq LSTM model to covert Mongolian word.

4.1 Rules

The rules include two parts. The first part overcomes the disadvantages of the method based on Seq2Seq LSTM model that this method can't ensure that all words in the dictionary are exactly decoded. In rules, for those words in the dictionary, their accurate phonemes sequences can be got by dictionary look-up. Because Mongolian is Agglutinative Language, the majority of Mongolian word with suffixes connected to the stem using Narrow Non-Break Space (NNBS), following work in [22], we also called those NNBS suffixes. The NNBS suffixes refer to case suffixes, reflexive suffixes and partly plural suffixes. They are used very flexible that each stem can add several NNBS suffixes to change Mongolian word form. The another part is to handle Mongolian word with NNBS suffixes, the pronunciation of Mongolian word with NNBS suffixes follow two rules, one rule called NNBS suffixes' rules is that NNBS suffixes' pronunciation depends on the form of stem's phoneme sequence, which is different due to varying form of stem's phonemes sequence. Another rule named stem's rules is that stem's phonemes sequence can be changed according to NNBS suffixes' phonemes sequence. We define four forms of stem's phonemes sequence for NNBS suffixes' rules as following:

- Form 1: The word-stem is a positive word and stem's phonemes sequence ends with a vowel, there are two cases. The first case is that stem's phoneme sequence ending with a vowel in the set {al, vl, ael, vi, vae, va, av}, then check whether Il or I exists in the stem's phoneme sequence. The second case is similar to the first case except that stem's phoneme sequence ending with a vowel in the set {wl, oel, wi, w}.
- Form 2: The word-stem is a negative word and stem's phonemes sequence ends with a vowel, there are two cases. The first case is that stem's phoneme sequence ending with a vowel in the set {el, ul, El, ui, ue, Yl, e, u}, then check whether il or i exists in the stem's phoneme sequence. The second case is similar to the first case except that stem's phoneme sequence ending with a vowel in the set {ol, Ol, o}.
- Form 3: The word-stem is a positive word and stem's phonemes sequence ends with a consonant, whether the first vowel searched from back to front exists in the set {a, v, Y, ae, as1, as2, vi, al, vl, ael, va, vae} or in the set {w, oe, wi, wl, oel, ws}, if Il or I is encountered, the next vowel should be searched forward from Il or I.
- Form 4: The word-stem is a negative word and stem's phonemes sequence ends with a consonant, whether the first vowel searched from back to front exists in the set {e, u, es, ui, El, el, ul, Yl, ue} or in the set {o, os, ol, Ol}, if il or i is encountered, the next vowel should be searched forward from il or i.

We list parts NNBS suffixes and their different phoneme sequence corresponding to varying form of the sequence of the stem in Table 1. The stem's rules include two parts, one part named as stem_rule1 is to determine whether NNBS suffixes' phonemes sequence starts with a long vowel, another part is to judge that whether stem's phonemes follows the stem_rule2. The stem's rules are showed as following in Table 2.

Table 1. Parts NNBS suffixes and their different phoneme sequence corresponding to varying form of the sequence of stem.

Mongolian	Latin	Form 1	Form 2	Form 3	Form 4
ᠶᠢᠨ	-yin	g il l	g il l	il l	Il l
ᠳᠤ	-dv	d	d	as1 d	ws d
ᠪᠠᠷ	-bar	g ar r	g wr r	ar r	wr r
ᠢᠶᠡᠷ	-iyer	al r	el r	ol r	wl r
ᠲᠡᠢ	-tei	t El	t Ol	t ael	t oel
ᠠᠴᠠ	-aqa	al s	wl s	el s	ol s
ᠠᠴᠠ ᠪᠠᠨ	-aqa-ban	g al s al n	g wl s wl n	al s al n	wl s wl n
ᠶᠢᠨ ᠢᠶᠡᠨ	-yin-iyen	g il n h el n	g il n h ol n	il n h el n	il n h ol n

Table 2. The stem's rules. C1, C2, V1, C3 and V, C1, V1, C2 are the last four phonemes of the stem's phonemes sequence, C1, C2 and C3 are the consonants, V and V1 are the vowels, A represents the phonemes sequence of the NNBS suffixes of Mongolian word.

Stem's phonemes	Stem_rule1	Stem_rule2	Word' s Phonemes
—C_1 C_2 V_1 C_3	A stars with a long vowel	V1 belongs to the set {as1, as2, es, ws, os}, C1 and C2 form a composite consonant.	—C_1 C_2 C_3 + A
—V C_1 V_1 C_2	A stars with a long vowel	V1 don't belong to the set {j, q, x, y, I, i}.	—V C_1 C_2 + A
Others	—	—	— + A

4.2 Combining Rules with Seq2Seq LSTM Model

Combining rules with Seq2Seq LSTM model for Mongolian G2P conversion are shown in Fig. 1. The procedures mainly comprise the following parts. Firstly, we transliterate all Mongolian words and its phonemes sequence in the dictionary to the Latin form, and transliterate input Mongolian word to the Latin form, if input Mongolian word exists in the dictionary, we can directly get word's phonemes sequence through the dictionary look-up. Secondly, Mongolian word does not exist in dictionary, if Mongolian word is with NNBS suffixes, we make use of combining rules with Seq2Seq LSTM model to handle this word. Instead of with NNBS suffixes, we decode Mongolian word by using Seq2Seq LSTM model. Finally, Mongolian word's phonemes sequence can be generated.

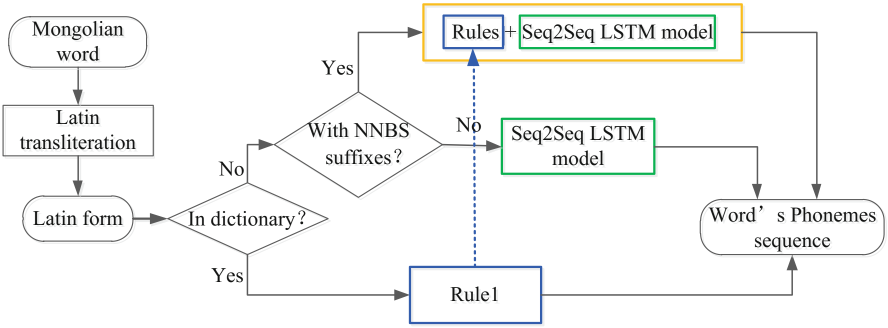

Fig. 1. The structure of combination rules and Seq2Seq LSTM model. The blue boxes represent that using rules to handle word, and Rule1 included in Rules is dictionary look-up. The green boxes represent that using Seq2Seq LSTM model to decode word, and the yellow box represents using hybrid approach. (Color figure online)

For Mongolian word with NNBS suffixes, we handle this word by using the hybrid method (see in Fig. 2). We firstly segment Mongolian word to stem and NNBS suffixes. If stem exists in the hash table, we can get stem's phonemes sequence from hash table, instead of this situation, we decode stem by using Seq2Seq LSTM model. Depending on stem's phonemes sequence and NNBS suffixes' rules, we can get NNBS suffixes' phonemes sequence, and then modify stem's phonemes sequence through NNBS suffixes' phonemes sequence and stem's rules. Mongolian word's phonemes

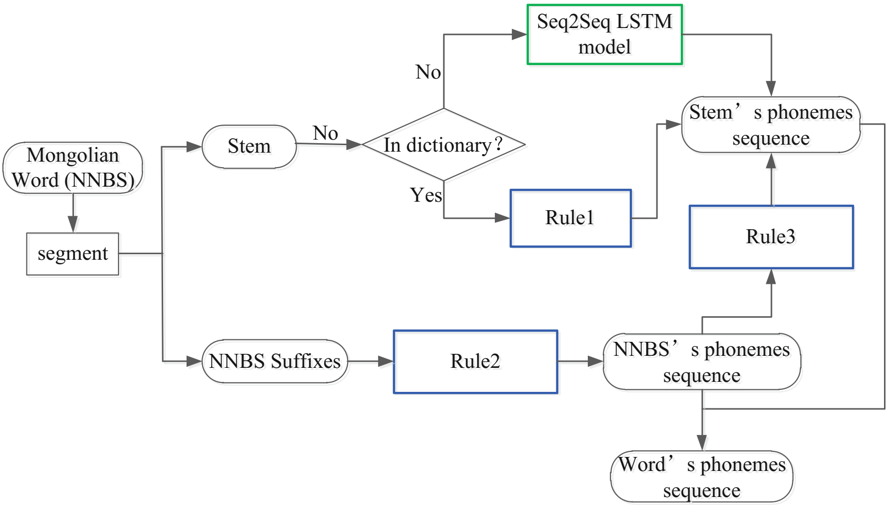

Fig. 2. The procedure of using the proposed method to handle Mongolian word with NNBS suffixes. The blue boxes represent rules included Rule1 (hash table look-up), Rule2 (NNBS suffixes' rules), and Rule3 (stem's rules). The green boxes represent that using Seq2Seq LSTM model to decode stem. (Color figure online)

sequence can be produced by jointing stem's phonemes sequence modified with NNBS suffixes' phonemes sequence.

5 Experiments

5.1 Data Set

This paper uses the Mongolian Orthography dictionary as an experimental dataset, Mongolian word with NNBS suffixes accounts for 15 percent of this dictionary. The dataset consisted of a training set of 33483 pairs of word and its phonemes sequence, a validation set of 1920 pairs of word and its phonemes and a test set of 3940 pairs of word and its phonemes. The evaluation criteria of the model are the word error rate (WER) and the phoneme error rate (PER).

$$WER = 1 - \frac{NW_{correct}}{NW_{wtotal}} \tag{13}$$

$$PER = 1 - \frac{NP_{ins} + NP_{del} + NP_{sub}}{NP_{ptotal}} \tag{14}$$

Where $NW_{correct}$ represents number of correctly decoded Mongolian words, NW_{wtotal} is the number of Mongolian words, NP_{ptotal} is the total number of phonemes corresponding to all the Mongolian words converted, NP_{ins}, NP_{del} and NP_{sub} is the quality of insertion errors, deletion errors and substitute errors of total phonemes, respectively.

5.2 Setting and Result

The baseline systems used in this paper are a rule-based G2P conversion system and a G2P conversion system based on the joint-sequence model. The performance of the rule-based G2P conversion system is that WER is 32.3% and PER is 7.6%, apparently, its result is not good. For Sequitur G2P, we tune the model order (n-gram) on the development set used to adjust the discount parameters of joint-sequence model. We found that the experimental result is better when the order is between 6 and 10, however, WER and PER are difficult to lower when the order is more than 10. The Table 3 shows the experimental result of Sequitur G2P.

We use the same dataset to train Seq2Seq LSTM model. We choose the width of the network' LSTM layers from the set {64,128,256,512,1024,2048}, number of layers from {1,2,3,4}, and choose stochastic gradient descent (SGD) as optimization method for network training, and learning rate is 0.5. In our experiment, we found that WER and PER are increasing as the width of LSTM layers and the number of layers increasing. We take better results (in Table 4) out of experimental results.

Comparing Table 3 with Table 4, we find that the performance of Mongolian G2P conversion based on Seq2Seq LSTM model is better than based on the joint-sequence model. We take the Seq2Seq LSTM model (1024x1) whose performance is best in

Table 3. Sequitur G2P's experimental result, where we test the joint-sequence model by decoding the test set and the train set. Best result for single model in bold.

Model	The test set		The train set	
	WER	PER	WER	PER
Model order 6	16.4%	3.4%	4.1%	0.8%
Model order 7	16.4%	3.4%	3.4%	0.7%
Model order 8	16.3%	3.6%	3.3%	0.7%
Model order 9	**16.3%**	**3.2%**	**2.9%**	**0.5%**
Model order 10	16.3%	3.2%	2.9%	0.5%

Table 4. The results of different Seq2Seq LSTM models tested, best result in bold

Model	The test set		The train set	
	WER	PER	WER	PER
LSTM model(512 × 1)	8.7%	2.4%	2.3%	0.6%
LSTM model(1024 × 1)	**8.0%**	**2.2%**	**1.0%**	**0.3%**
LSTM model(128 × 2)	9.7%	2.6%	3.5%	0.9%
LSTM model(256 × 2)	8.8%	2.4%	2.4%	0.6%

Table 4 to combine with rules for Mongolian G2P conversion. We firstly randomly take the same amount of the test dataset of pairs of Mongolian words and their phoneme sequences, and write them in a dictionary for look-up. We get the experimental result to compare with above methods (see in Table 5).

Table 5. The comparing result of testing the best joint-sequence model (order 9), the best seq2seq LSTM model (1024 × 1) and the hybrid method.

Method	The test set		The train set	
	WER	PER	WER	PER
Model order 9	16.3%	3.2%	2.9%	0.5%
LSTM model(1024 × 1)	8.0%	2.2%	1.0%	0.3%
LSTM model(1024 × 1) + rules	**5.5%**	**1.6%**	**0.7%**	**0.2%**

We can see from Table 5, the performance using Seq2Seq model (1024 × 1) is better than using the best joint-sequence model, the WER and the PER reduce by 8.3% and 1.0%, respectively. Although using the same Seq2Seq LSTM model (1024 × 1), the method based on combing rules with Seq2Seq LSTM model performs better, it's WER and PER is 5.5% and 1.6%, respectively. There are two reasons for performance improvement after combining rules. Firstly, if Mongolian word exists in dictionary, we can get exact word's phonemes sequence through rules (dictionary look-up), this approach is apparently more accurate than Seq2Seq LSTM model. Secondly, Mongolian words with NNBS suffixes are ordinary, because of characteristics of stem's pronunciation and NNBS suffixes' pronunciation, it is difficult to exactly cope with this situation by only using Seq2Seq LSTM model, combing rules (stem's rules and NNBS suffixes' rules) can get more accurate phonemes sequence.

6 Conclusion

In this paper, we present the first application of Seq2Seq LSTM model with attention mechanism for Mongolian G2P conversion, the experimental results show that the Mongolian G2P conversion based on Seq2Seq model can get better performance than the previous methods. We continuously adjusted the parameters of the model. We obtain a best Seq2Seq LSTM model (1024x1), and we use the best model to combine with rules for Mongolian G2P conversion, and experimental results became better than the method only using Seq2Seq model. This method proposed is of profound significance to Mongolian G2P conversion, meantime, it is greatly beneficial to the study of Mongolian speech synthesis, speech retrieval and speech recognition. When we go further, and try model fusion for Mongolian G2P conversion, we assume that model fusion may make significant advances.

Acknowledgement. This research was supported by the China national natural science foundation (No. 61563040, No. 61773224) and Inner Mongolian nature science foundation (No. 2016ZD06).

References

1. Hojo, N., Ijima, Y., Mizuno, H.: An investigation of DNN-based speech synthesis using speaker codes. In: INTERSPEECH, pp. 2278–2282 (2016)
2. Merritt, T., Clark, R., Wu, Z.: Deep neural network-guided unit selection synthesis. In: IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 5145–5149. IEEE (2016)
3. Liu, R., Bao, F., Gao, G., Wang, Y.: Mongolian text-to-speech system based on deep neural network. In: Tao, J., Zheng, T.F., Bao, C., Wang, D., Li, Y. (eds.) NCMMSC 2017. CCIS, vol. 807, pp. 99–108. Springer, Singapore (2018). https://doi.org/10.1007/978-981-10-8111-8_10
4. Graves, A., Mohamed, A., Hinton, G.: Speech recognition with deep recurrent neural networks. In: IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 6645–6649. IEEE (2013)
5. Wang, Y., Bao, F., Zhang, H., Gao, G.: Research on Mongolian speech recognition based on FSMN. In: Huang, X., Jiang, J., Zhao, D., Feng, Y., Hong, Y. (eds.) NLPCC 2017. LNCS (LNAI), vol. 10619, pp. 243–254. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-73618-1_21
6. Zhang, H., Bao, F., Gao, G.: Mongolian speech recognition based on deep neural networks. In: Sun, M., Liu, Z., Zhang, M., Liu, Y. (eds.) CCL 2015. LNCS (LNAI), vol. 9427, pp. 180–188. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-25816-4_15
7. Bao, F., Gao, G., Bao, Y.: The research on Mongolian spoken term detection based on confusion network. Commun. Comput. Inf. Sci. **321**(1), 606–612 (2012)

8. Lu, M., Bao, F., Gao, G.: Language model for Mongolian polyphone proofreading. In: Sun, M., Wang, X., Chang, B., Xiong, D. (eds.) CCL/NLP-NABD -2017. LNCS (LNAI), vol. 10565, pp. 461–471. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-69005-6_38
9. Bisani, M., Ney, H.: Joint-sequence models for grapheme-to-phoneme conversion. Speech Commun. **50**(5), 434–451 (2008)
10. Chen, S.F.: Conditional and joint models for grapheme-to-phoneme conversion. In: European Conference on Speech Communication and Technology, INTERSPEECH 2003, Geneva, Switzerland, DBLP (2003)
11. Jiampojamarn, S., Kondrak, G., Sherif, T.: Applying many-to-many alignments and hidden markov models to letter-to-phoneme conversion. In: Proceedings of Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, USA, pp. 372–379 (2008)
12. Rao, K., Peng, F., Sak, H.: Grapheme-to-phoneme conversion using long short-term memory recurrent neural networks. In: IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 4225–4229. IEEE (2015)
13. Yao, K., Zweig, G.: Sequence-to-sequence neural net models for grapheme-to-phoneme conversion. Computer Science (2015)
14. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. Computer Science (2014)
15. Toshniwal, S., Livescu, K.: Jointly learning to align and convert graphemes to phonemes with neural attention models. In: Spoken Language Technology Workshop. IEEE (2017)
16. Bao, F., Gao, G.: Research on grapheme to phoneme conversion for Mongolian. Appl. Res. Comput. **30**(6), 1696–1700 (2013)
17. Luong, M.T., Sutskever, I., Le, Q.V.: Addressing the rare word problem in neural machine translation. Bull. Univ. Agric. Sci. Vet. Med. Cluj-Napoca Vet. Med. **27**(2), 82–86 (2014)
18. Jean, S., Cho, K., Memisevic, R.: On using very large target vocabulary for neural machine translation. Computer Science (2014)
19. Milde, B., Schmidt, C., Köhler, J.: Multitask sequence-to-sequence models for grapheme-to-phoneme conversion. In: INTERSPEECH, pp. 2536–2540 (2017)
20. Sutskever, I., Vinyals, O., Le, Q.V.: Sequence to sequence learning with neural networks. In: NIPS, pp. 3104–3112 (2014)
21. Vinyals, O., Kaiser, L., Koo, T.: Grammar as a foreign language. Eprint Arxiv, pp. 2773–2781 (2014)
22. Wang, W., Bao, F., Gao, G.: Mongolian named entity recognition system with rich features. In: The 26th International Conference on Computational Linguistics, pp. 505–512. Proceedings of the Conference, Japan (2016)