A language model for Amdo Tibetan speech recognition

Taiben Suan[1, 3, 4, 5*], *Rangzhuoma* Cai[1, 2, 4, 5], *Zhijie* Cai[1, 2, 4, 5], *Ba* Zu[1, 4, 5], and *Baojia* Gong[1, 4, 5]

[1]College of Computer Science and Technology, Qinghai Normal University, Xining, Qinghai 810016, China
[2]School of Computer Science and Technology ，Southwest Minzu University, Sichuan Chengdu 610041, China
[3]Xinlong county Meteorological Bureau, Xinlong county, Sichuan 626800 , China
[4]Qinghai Provincial Key Laboratory of Tibetan Information Processing and Machine Translation, Xining, Qinghai 810008, China
[5]Key Laboratory of Tibetan Information Processing, Ministry of Education, Xining, Qinghai 810008, China

Abstract. We built a language model which is based on Transformer network architecture, used attention mechanisms to dispensing with recurrence and convalutions entirely. Through the transliteration of Tibetan to International Phonetic Alphabets, the language model was trained using the syllables and phonemes of the Tibetan word as modeling units to predict corresponding Tibetan sentences according to the context semantics of IPA. And it combined with the acoustic model as the Tibetan speech recognition was compared with end-to-end Tibetan speech recognition.

1 Introduction

The research on the Tibetan language model is still in its infancy [1], and there are some research based on the deep neural network [2-3] but least for speech recognition. In speech recognition, the use of deep learning algorithms can achieve end-to-end speech recognition with word or phrase as the modeling unit [4-7]. The neural network model is better than traditional models in speech recognition but it depends on large volumes of data, requiring a lot of speech data for training to realize its potential. Tibetan is a minority language with a relatively small population in China. It is mainly divided into three dialects; U-Tsang, Kamba, and Amdo. Thus the speech data required for the end-to-end Tibetan speech recognition model training is more difficult to collect than the corpus of text data. Therefore, Tibetan speech recognition still uses syllables or phonemes as modeling units, and the combination of acoustic models and language models has a better performance. The content of this paper is a language model for Amdo Tibetan speech recognition, how to transliterate Tibetan sentences into corresponding IPA, and train language models using syllables or phonemes as modeling units for speech recognition tasks.

[*] Corresponding author: aiswoboo@gmail. com

2 Transformer component

Transformer was originally used in the field of machine translation [8]. It is different from RNN (Recurrent Neural Network) and CNN (Convolutional Neural Network) structures. It uses a self-attention mechanism for relating different positions of sequence in order to compute a representation of one word in sequence, and, at the same time, process the sequence in parallel. It's entire model framework is completely built with attention mechanism and feed-forward neural network, and Transformer's training speed and performance are much better than RNN [9].

2.1 Multi-head attention

First, dot-product of the query sequence and all the keys sequence is divided by the scaling factor $\sqrt{d_k}$, and then a softmax function is applied to obtain the weights of the values sequence to computing the scaled dot-product attention. The scaling factor plays an adjustment role, so that the dot product grow large, resulting in pushing the softmax function to an area with an extremely small gradient. The output matrix is:

$$Attention(Q,K,V) = soft\max(\frac{QK^T}{\sqrt{d_k}})V \tag{1}$$

Multi-head attention can be understood as performing Scaled Dot-product Attention multiple times without sharing parameters, projecting Q, K, and V through h times different linear transformations, and then concatenate different results, finally output through a linear mapping. The multi-head attention compute as:

$$MultiHead(Q,KV) = Concat(head_1,...,head_h)W^O \tag{2}$$

$$\text{Where } head_i = Attention(QW_i^Q, KW_i^K, VW_i^V) \tag{3}$$

where the projections are parameter matrices:

$$W_i^Q \in \mathrm{R}^{d_{\mathrm{model}\times \mathrm{d}_k}}, W_i^K \in \mathrm{R}^{d_{\mathrm{model}\times \mathrm{d}_k}}, W_i^V \in \mathrm{R}^{d_{\mathrm{model}\times \mathrm{d}_v}},\quad W^O \in \mathrm{R}^{hd_v\times d_{\mathrm{model}}}$$

2.2 Feed-forward neural network and positional decoding

The feed-forward neural network consists of two linear transformations with a ReLU activation in between.

$$FFN(x) = \max(0, xW_1 + b_1)W_2 + b_2 \tag{4}$$

x represents the input; W_1 represents the parameter matrix of the first linear transformation; b_1 represents the bias vector of the first linear transformation; W_2 represents the parameter matrix of the second linear transformation; b_2 represents the bias vector of the second linear transformation.

Transformer model does not contain any RNN and CNN structure, but the position of each word is closely related to the final output. So encode the position of each word to implement the model using sequence order information, The specific calculation formula is expressed as:

$$PE(pos,2i) = \sin(pos/1000^{2i/d_{\mathrm{model}}}) \tag{5}$$

$$PE(pos,2i+1) = \cos(pos/1000^{2i/d_{\mathrm{model}}}) \tag{6}$$

where pos is position of the word in the sequence; i represents the i-th dimension of the word vector; d_{model} is the dimension of the word vector. Use sin and cos to encode position information, such an encoding method can express both the absolute and relative position of the word.

3 Tibetan language model

3.1 Tibetan phonetic transcription

In order to make the language model predict Tibetan sentences according to the context semantics of IPA, and then combine with the acoustic model to play a role in speech recognition, we need to transliterate Tibetan into the corresponding IPA sequence as the input of the Transformer. Tibetan words are used as the output of the transformer to training model. Tibetan script is a horizontal and vertical two-dimensional phonetic script composed of consonants and vowels. It is composed of 7 basic components according to strict Tibetan grammar rules. According to the spelling order, they are Prefix, Superscript, Root Consonant, Subscript, Vowel sign, Suffix, and Second Suffix [10]. There is a many-to-one mapping relationship between Tibetan words and corresponding phonetic symbols. Tibetan word are separated by a tsek '་'. Usually a Tibetan word is a syllable [11], consisting of single or multiple consonants and monophones or a combination of monophones and final consonants [12]. In this paper, the four components of the Tibetan syllables:Prefix, Superscript, Root Consonant, Subscript are used as consonant phonemes, Vowel sign, Suffix, and Second Suffix are used as vowel phonemes. For example, the Tibetan sentence"བོད་སྐད་སྨྲ་བརྗོད" (Tibetan speech) is transcribed into "wot ʰkat ʰma ɟʷot". But in fact, the recognition result of the acoustic model cannot be so accurate. For example, sometimes the "སྨྲ་བརྗོད" voice signal in a certain context is recognized as "ʁʰma ɟot" or even can be "ma ɟot". This is because the acoustic characteristics of some Tibetan word are similar, or the contextual voice will affect the acoustic characteristics of the current word. To this end, we transcribed a small part of the training data in a broad way, ignores as many details as possible, or transcribed them according to the acoustic characteristics of the context.

Table 1. Broad and narrow transcription.

Broad way	Narrow way	IPA
ཀ་ག་བཀའ	ཀ་ག	k a
ཁ་མཁའ་འཁའ	ཁ	kʰ a
ཁྲ་ཆ་མཁྲ་འཁྲ་མཆའ་འཆའ་འཕྲ	ཁྲ་ཆ	cʰ a
དགྲ་རྒྱ་ཇ་ལྗ་སྒྲ་བགྲ་བརྒྱ་བསྒྲ་བཇ་སྦྲ	རྒྱ་ཇ་ལྗ་སྒྲ	ɟ a
. . .	. . .	. . .

3.2 Model architecture

The model framework is mainly composed of encoder and decoder. The encoder is composed of 6 network blocks, all network blocks are the same in structure, but they do not share parameters. The network block of the decoder is the same as the network block of the encoder. It is also composed of 6 network blocks. In order to optimize the training process, the entire network uses residual connections and normalizes the layers.

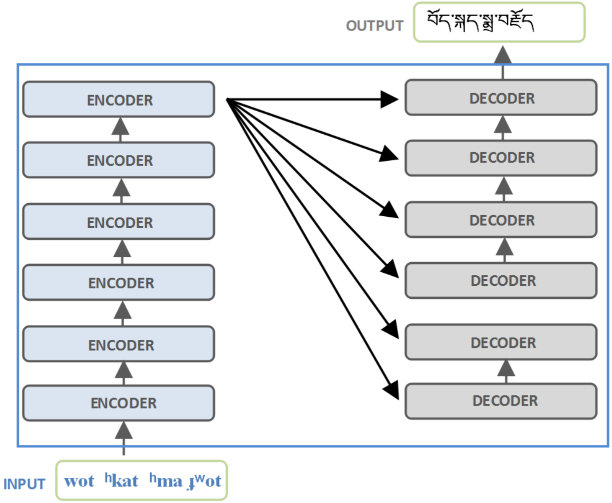

Fig. 1. Model architecture.

3.2.1 Encoder

Each encoder can be divided into two sublayers:self-attention sublayer and feed-forward neural network sublayer.

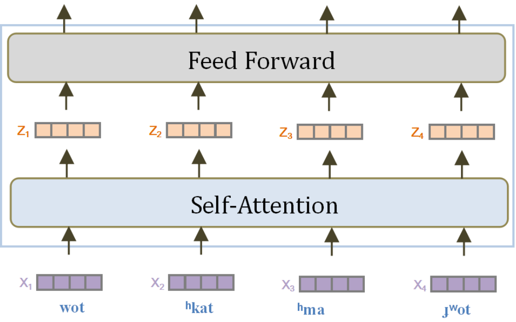

Fig. 2. Encoder architecture.

After transcribing a Tibetan sentence into an IPA sequence, it must be vectorized by the embeding layer. The vector sequence dimension is 512. When the transcribed IPA is a syllable or phoneme, the sequence length is set to 25 and 60 respectively. Then it encodes each vector in the position coding layer to obtain the relative or absolute position information in sequence. Next, it is linearly transformed into three vectors q, k, and v, and passed to the self-attention layer, processed with multi-head self-attention, and each group of vectors is processed through 8 heads to feature extract for the model to jointly attend to information from different representation subspaces at different positions with considering the whole sequence. Then concatenates and compresses the results into one vector sequence through linear transformation, giving it to the feed-forward neural network layer, and then input to the next block of encoder. Finally, the Encoder output sequence converts it into Attention matrices (K , V) and sends it to the Decoder.

3.2.2 Decoder

Similarly, before input to the decoder, word embedding and position encoding are also required. The decoder consists of a self-attention sublayer, an encoder-decoder attention sublayer and a feed-forward neural network sublayer. The difference is that there is an additional masked multi-head attention layer in the decoder.

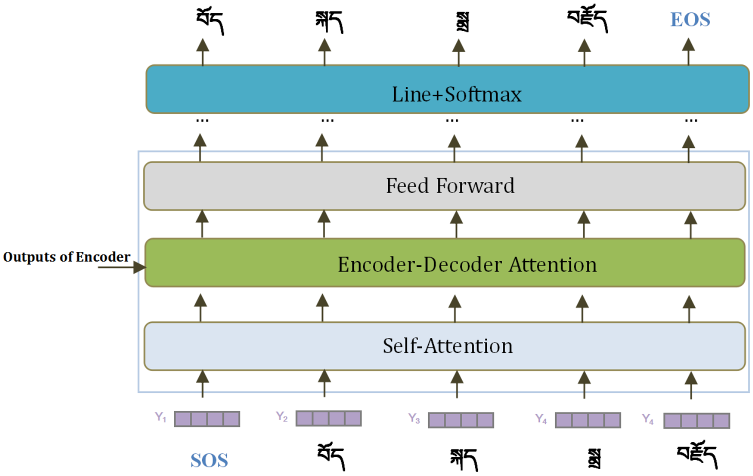

Fig. 3. Decoder architecture.

When training the model, the first sub-layer of the decoder uses masked multi-head self-attention, so that the input only contains the word information before the current position to achieve sequential decoding, and the current output can only be based on the outputed part. The input of the encoder-decoder attention layer is to create the Q matrix from the self-attention sublayer, and obtain the K matrix and V matrix from the output of the encoder. Therefore, the input to the decoder includes not only the output of the encoder, but also the output of the previous decoder. Finally, each step of the decoding stage through the linear layer and the softmax layer will output a Tibetan word. During the test, the first input of the decoder is the start symbol SOS, and the Tibetan output of each step is provided to the bottom decoder in the next time step until the special end symbol EOS is reached.

4 Experiment

The corpus data to training language model of this paper is mainly composed of 200M texts of Tibetan and corresponding phonetic symbols through transliteration. In order to test the performance of the speech recognition system when the language model is combined with the acoustic model, we used the speech data of 10-hour reading of Tibetan textbooks for primary and secondary schools by 1 female and 1 male (Channel:Mono, Sampling:16 bit, Bit rate:16kHz). we trained language model on 2 Tesla M60 GPSs with following hyperparaments:

Optimizer:Adam optimizer (learning_rate=0. 0001, β_1=0. 9, β_2=0. 98, ε=1e-8),

Num_heads=8, Num_blocks=6, Hidden_unite=512, Dropout=0. 2

The performance of the transformer based language model on the task is shown by WER(Word Error Rate) in the following table:

Table 2. Language model performance

Unit	phoneme	syllable
WER	0. 094	0. 105

We use the CNN+CTC based acoustic model trained by speech data to combine with the language model. The results on test dataset of selecting different modeling unite as follows:

Table 3. Speech Recognition System performance

Speech Recognition System		**Unit**	**WER**
AM	**LM**		
CNN+CTC	No LM	word	0. 569
	Transformer	syllable	0. 506
		phoneme	0. 415

5 Conclusion

The experimental results show that the performance of Tibetan speech recognition combined with an acoustic and language model is better than end-to-end speech recognition. Additionally, a speech recognition system where using the phoneme as modeling unit is better than using the syllable as modeling unit. Due to the small scale of training data, the generalization ability of the model is weak and cannot be applied in practice. In terms of phonetic transcription there are still some problems and room for improvement. For example, it is impossible to accurately transliterate Sanskrit-Tibetan words that conform to the Tibetan grammar rules.

This research was financially supported by the National Natural Science Foundation of China (61966031, 61866032); Ministry of Education "Chunhui Program" (Z2016077, Z2012093); Qinghai Province Science and Technology Project (2019-SF-129); Qinghai Province Key Laboratory Project (2014-Z-Y32, 2015-Z-Y03); Key Laboratory of Tibetan Information Processing and Machine Translation (2013-Y-17).

References

1. G. Yang, Y. Cuo. Research Status and Prospect of Tibetan Language Model [J]. Computer Knowledge and Technology. **16** 3(2020)
2. S. Tongtong. Research on Tibetan Language Model Based on Recurrent Neural Network [D]. Tianjin University. 12(2017)
3. H. Zhaxi. Research on Tibetan Word Spell Checking Technology based on LSTM [D]. Qinghai Normal University. 3(2020)
4. Geoffrey E. Hinton, Simon Osindero, Yee-Whye Teh. A Fast Learning Algorithm for Deep Belief Nets [J] . Neural Computation. 7(2006)
5. Tjandra A, Sakti S, Nakamura S. End-to-end speech recognition sequence training with reinforcement learning [J]. IEEE Access. (2019)

6. Rosenberg A, Audhkhasi K, Sethy A, et al. End-to-end speech recognition and keyword search on low-resource languages [C]. Speech and Signal Processing. 5280-5284. (2017)
7. Zhao Yue, Yue Jianjian, Xu Xiaona, et al. End-to-end-based Tibetan multitask speech recognition. IEEE Access. (2019)
8. Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need [C]. Advances in Neural Information Processing Systems. 5998-6008. (2017)
9. W. Junhao, L. Yifeng, Enriching image descriptions by fusing fine-grained semantic features with a transformer [J]. Journal of East China Normal University. (2020)
10. C. Zhuoma, C. Zhijie. An algorithm for word component decomposition in Tibetan character frequency statistics system [J]. Computer Engineering and Science. 33(3):159-162. **15**. (2011)
11. C. Zhijie. Research on Key Techniques of Tibetan Word Vector Representation [D]. Qinghai Normal University. (2018)
12. W. Shuangcheng, The characteristics of the complex vowels in Amdo Tibetan [J]. National language. 3(2004)