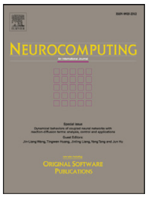
Neurocomputing 358 (2019) 44–52

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom



Hybrid Attention for Chinese Character-Level Neural Machine Translation

Feng Wang[a,b,*], Wei Chen[b], Zhen Yang[b], Shuang Xu[b], Bo Xu[b]

[a] *University of Chinese Academy of Sciences, Beijing 100190, P.R. China*
[b] *Institute of Automation, Chinese Academy of Sciences, No.95 ZhongGuanCun East Road, Beijing 100190, P.R. China*

ARTICLE INFO

Article history:
Received 30 January 2018
Revised 31 March 2019
Accepted 12 May 2019
Available online 16 May 2019

Communicated by Dr. Tie-Yan Liu

Keywords:
Neural machine translation
Hybrid attention
Character
Word segmentation

ABSTRACT

This paper proposes a novel character-level neural machine translation model which can effectively improve the Neural Machine Translation (NMT) by fusing word and character attention information. In our work, the bidirectional Gated Recurrent Unit (GRU) network is utilized to compose word-level information from the input sequence of characters automatically. Contrary to traditional NMT models, two kinds of different attentions are incorporated into our proposed model: One is the *character-level attention* which pays attention to the original input characters; The other is the *word-level attention* which pays attention to the automatically composed words. With the two attentions, the model is able to encode the information from the character level and word level simultaneously. We find that the composed word-level information is compatible and complementary to the original input character-level information. Experimental results on Chinese-English translation tasks show that the proposed model can offer a boost of up to +1.92 BLEU points over the traditional word based NMT models. Furthermore, our translation performance is also comparable to the latest outstanding models, including the state-of-the-art.

1. Introduction

Neural Machine Translation (NMT) [1–3] has gained popularity in solving the machine translation problem in recent years. Its architecture typically consists of an encoder and a decoder, which are typically parameterized as Long Short-Term Memory (LSTM) [4] or GRU [5] networks, often with residual connections [6,7] to enable stacking of several layers. Neural machine translation has achieved state-of-the-art performance on several parallel corpus of multiple languages [8,9].

Most of previous work in machine translation focuses on word-level translation. Since the word-level based NMT system usually uses the top-N frequent words in the training corpus and regards other words as unseen words, one of the most remarkable limitations for the word-level based translation is that the NMT system is usually weak in handling the Out-Of-Vocabulary(OOV) and rare words, which may damage its translation performance [10–13]. Some approaches towards this problem seek to cover more words by adopting an identity translation dictionary in a post-processing step [14] or using a larger vocabulary [15,16]. These methods could be called *vocabulary-specific* approaches. Intuitively, only if the vocabulary can be expanded large enough, the OOV problem can be alleviated to a certain extent by these approaches. However, the vocabulary is always limited so that these approaches are incapable of solving the OOV problem completely. The other branch of approaches is *unit-specific* [17], such as sub-word units [18] or even character-level units [19,20], which use more fine-grained processing unit than words. Taking character as the basic processing unit is a new trend, which has been widely used in many NLP tasks [21–23].

Due to these limitations associate with word-level translation, developing character-level NMT models is attractive for multiple reasons. Firstly, the OOV problem will be relieved or vanished. And the model can be trained with less vocabulary, which enables the character NMT model to solve many scalability issues, both in terms of computational speed and memory requirements. Secondly, since each character occurs frequently in the training corpus, all of the character embeddings are able to get full trained.

The character-level neural machine translation models have achieved some successes by composing word information from the input character sequence. However, they heavily rely on the composed word information yet pay less attention to the character-level information. We conjecture that both of the word-level and character-level information are vital for the translation model. If we combine the composed word-level information and the in-

* Corresponding author at: Institute of Automation, Chinese Academy of Sciences, No.95 ZhongGuanCun East Road, Beijing 100190, P.R. China.

E-mail addresses: feng.wang@ia.ac.cn (F. Wang), wei.chen.media@ia.ac.cn (W. Chen).

https://doi.org/10.1016/j.neucom.2019.05.032

put character-level information, the translation performance of the NMT model may get boosted. Luong and Manning [24] propose a hybrid architecture for NMT that translates mostly at the word level and consults the character components for rare words when necessary. Their model achieves significant improvement by incorporating character-level information for the rare words. However, they do not combine all of the character-level information for all the words in their dictionary.

In this work, we propose a novel character-level NMT model which implements a hybrid attention by fusing the word-level information and character-level information at the same time. The proposed hybrid attention model has two attention layers: one attention layer is used to focus on the original character input sequence; the other one pays its attention to the composed word-level information. We do not assume the specific language pairs, because the model can accept any language by taking its smallest text granularity as input. For languages without explicit boundaries, such as Chinese, text can be fed into the model in character granularity directly and characters can be combined into words with the segmentation information extracted from large-scale monolingual data. For languages with explicit boundaries, such as English, text can be fed character by character until the word boundary. We still keep the target side being represented as a sequence of words. This paper has two main contributions:

- We propose a novel NMT model which views the input as character sequence. We introduce the word segment information as a kind of significant feature to encode the character vector, which does not suffer from the OOV problem in the source encoder.
- Owing to the hybrid attention, the proposed model can incorporate the character-level and word-level information at the same time. We find that the composed word-level information is compatible and complementary to the original input character-level information. Experimental results show that the proposed model achieves significant improvement than the strong baseline models.

2. Related work

Taking character as the basic processing unit is a new trend in the field of NLP. And the character-level models have been widely used in NLP tasks [21–23]. Based on the state-of-the-art attention based encoder-decoder framework, some character-level NMT models have been proposed recently. Chung et al. [20] focuses on representing the target side as a character sequence with a bi-scale recurrent neural network. In [17,19], a character-based encoder is proposed in the source side. And a full character-level model which maps the source character sequence to a target character sequence without any segmentation has been proposed in [25].

We draw the inspiration for our hybrid attention character-level NMT from [24]. However, different from that, we suppose that word segmentation is a significant feature for the machine learning. We make an attempt at generating the word segmentation embedding from the simple character input for the neural machine translation, which can be integrated to the translation system as an additional feature afterwards. Contrary to their hybrid scheme that focuses on the style of the source and target, our model pays more attention to the alignment for the word and character attention information.

2.1. Chinese word segmentation

Unlike English and other western languages, most east Asian languages, including Chinese, are written without explicit word delimiters. Therefore, word segmentation is an essential step for processing those languages. In traditional word segmentation task, Xue et al. [26,27] propose to consider it as a sequence label problem, which can be handled with supervised learning algorithms such as Maximum Entropy (ME) [28] and Conditional Random Fields (CRFs) [29].

In recent years, Chinese Word Segmentation (CWS) has undergone great development and neural networks have been widely used. [30] minimizes the effort in feature engineering. The subsequent ideas vary in the network architecture. Zheng et al. [31] propose a perceptron style algorithm to speed up the training process. In particular, feed-forward neural network [31], recursive neural network [32], LSTM [33], have been used to derive contextual representations from input sequences. Full word information has shown its effectiveness for word based CWS system [34–36]. Liu et al. [37,38] utilize word embeddings to boost the performance of word-based CWS method. Previous work shows that character-based and word-based CWS models can complement each other [39,40]. Therefore, we conjecture that incorporating the character-based CWS model in the neural machine translation system may be tremendously beneficial to improve the performance of translation.

2.2. Attention-based NMT

Attention mechanism is a widely studied task in the NLP community. This framework has been widely used as baseline, which effectively improves the performance of the sequence to sequence task. This subsection briefly describes the attention-based encoder-decoder framework.

The attention mechanism simultaneously produces the translation sentence by generating one target word at every time step, which is used to conduct dynamic alignment. At each time i, given an input sequence $x = (x_1, \ldots, x_{T_x})$ and previous translated words $(y_1, \ldots, y_{i-1})$ as the forward RNN input, the probability of next word y_i is:

$$p(y_i|y_1, \ldots, y_{i-1}, x) = g(y_{i-1}, c_i, s_i) \tag{1}$$

Where s_i is a decoder hidden state for time step i, which is computed as:

$$s_i = f(s_{i-1}, c_i, y_{i-1}) \tag{2}$$

Here g and f are nonlinear transform functions, which can be implemented as GRU or LSTM. And c_i is a distinct context vector at time step i, which is calculated as a weighted sum of the input annotations h_j:

$$c_i = \sum_{j=1}^{T_x} a_{i,j} h_j \tag{3}$$

Where h_j is the annotation of x_j from a bidirectional RNN. The weight a_{ij} for h_j is calculated as:

$$a_{i,j} = \frac{exp(e_{ij})}{\sum_{t=1}^{T_x} exp(e_{i,t})} \tag{4}$$

Where

$$e_{i,j} = v_a tanh(W s_{i-1} + U h_j) \tag{5}$$

3. Hybrid attention model

In this section, we describe the proposed character-level based translation model with hybrid attention in detail. Fig. 1 is the graphical illustration of our model. Since we consider the settings where the source language has no explicit boundaries, we use a word segmentation module to get the segmentation information automatically. Then, a bidirectional GRU is adopted to compose

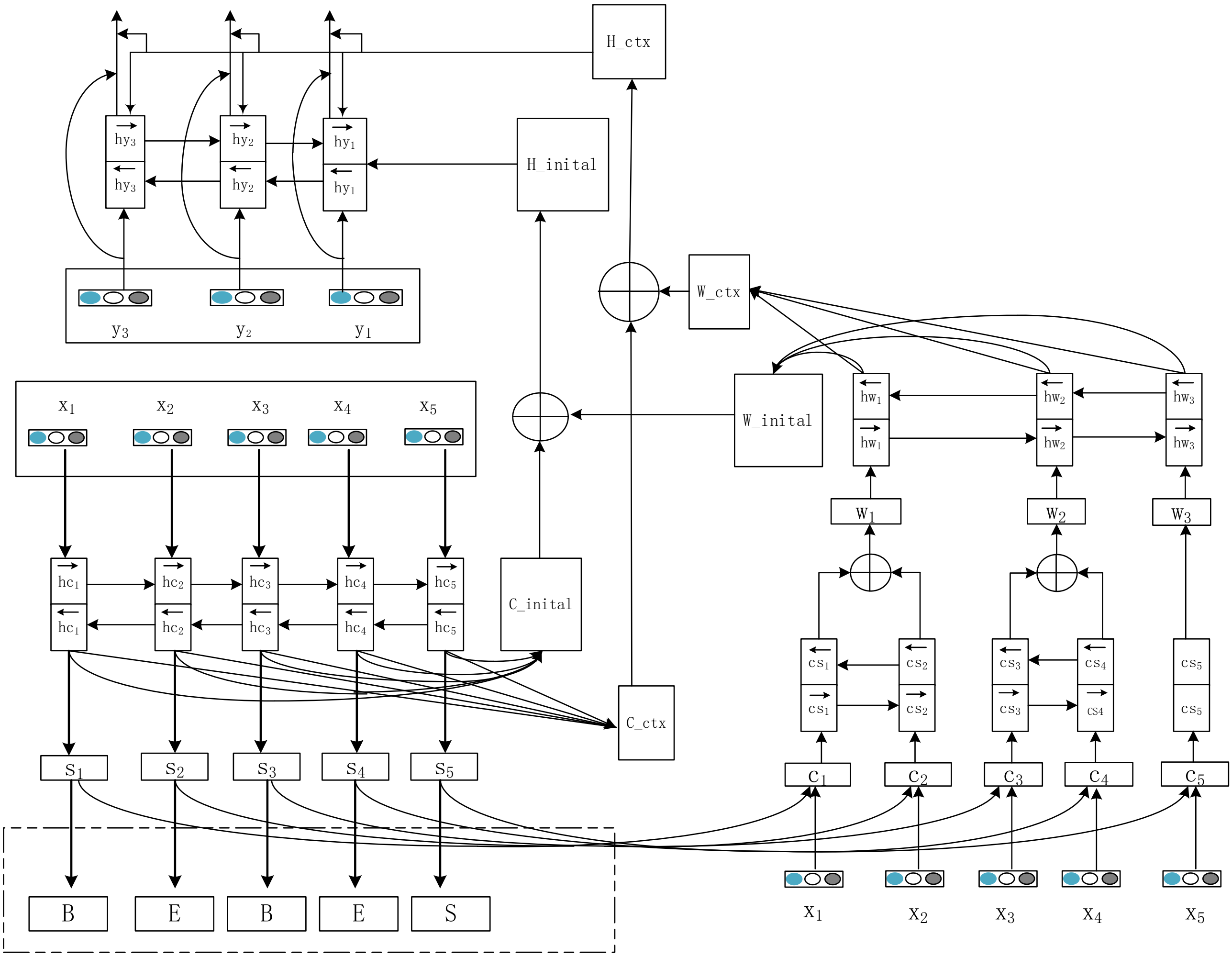


Fig. 1. The architecture of the proposed model.

the word representation from the input character sequence. Based on the representations of words and characters, we propose a decoder with hybrid attention to dynamically build the context representation from the word level and character level at the same time.

3.1. Chinese word segmentation

The word segmentation module is implemented with the classical framework for end-to-end sequence labeling tasks as well. We define the {B,S,M,E} as the word segmentation tags, where the {B,M,E} represent begin, middle, end of a multi-character segmentation respectively and the {S} represents a single-character segmentation. The word segmentation module is composed of three specialized layers, i.e., character embedding layer, bidirectional GRU encoding layer and tag inference layer. In our practice, the bidirectional GRU encoding layer shares the same weight parameters with the encoder of the NMT model.

Formally, considering the input character sequence $x_1, \ldots, x_{T_x}$ (T_x equals five in Fig. 1), the model projects each character into a continuous d-dimensional character embedding x_i using a character lookup table. Feeding the character embedding sequence into a bidirectional GRU layer, we get the annotation h_i for x_i. With a simple forward neural network layer, h_i is transformed to the hidden state s_i, which is used for classifying the word segmentation tags directly.

3.2. Word composing

With the character embedding and the segmentation information, we can compose the word-level information with a bidirectional GRU layer. Segmentation tag for each input character can be obtained by the word segmentation module. Here we take the segmentation result B,E,B,E,S as an instance. According to the segmentation result, the input character sequence is segmented into three words which are represented as w_1, w_2, w_3. Since w_1 is composed by x_1 and x_2, it is computed as:

$$w_1 = \frac{1}{2} * (m_1 + m_2) \tag{6}$$

Where m_1 and m_2 are the hidden states of the bidirectional GRU with the input c_1 and c_2. c_i is computed as:

$$c_i = [s_i; x_i] \tag{7}$$

Where the colon represents the operation of concatenation. Since w_3 only includes the character x_5, it is calculated as:

$$w_3 = m_5 \tag{8}$$

Where m_5 is the hidden state of the bidirectional GRU with the input c_5.

3.3. Encoder

Different from traditional NMT models, the encoder in the proposed model encodes information from both the word-level and character-level. Specifically, the encoder contains two separate bidirectional GRUs: One is used to transform the character embedding sequence $x_1, \ldots, x_{T_x}$ into the hidden variable sequence $hc_1, \ldots, hc_{T_x}$; The other encodes the sequence of word representations $w_1, \ldots, w_{T_w}$ into the hidden variable sequence $hw_1, \ldots, hw_{T_w}$, where T_w varies to the segmentation result. Here a shared bidirectional GRU is adopted to encode the character embeddings into word embeddings according to the word segmentation module, which is a practical way to keep the simplicity.

3.4. Decoder with hybrid attention

During decoding, the traditional character-level NMT reads the hidden variables of the encoder and predicts the target sequence. Compared to [3], there are three main differences:

The first difference is the initialization method for the hidden state of the GRU. Unlike traditional NMT model which initializes the hidden state only with the average of the character-level context, the hidden state in the proposed model is initialized as:

$$H_{initial} = \alpha * c_{initial} + \beta * w_{initial} \tag{9}$$

Where $H_{initial}$ denotes the initial state, $c_{initial}$ and $w_{initial}$ are the character-level context and word-level context respectively. The $c_{initial}$ is computed as:

$$c_{initial} = \frac{1}{T_x} \sum_{i=1}^{T_x} hc_i \tag{10}$$

and $w_{initial}$ is calculated as:

$$w_{initial} = \frac{1}{T_w} \sum_{i=1}^{T_w} hw_i \tag{11}$$

The second difference is the computing for the context H_{ctx}. In the proposed model, the H_{ctx} contains the word-level context W_{ctx} and the character-level context C_{ctx}. It is calculated as:

$$H_{ctx} = \alpha * C_{ctx} + \beta * W_{ctx} \tag{12}$$

To compute the C_{ctx} and W_{ctx}, a hybrid attention is utilized. Specifically, one word-level attention is used to compute W_{ctx} and the character-level attention is used to compute C_{ctx}.

$$W_{ctxi} = \sum_{j=1}^{T_w} aw_{i,j} hw_j \tag{13}$$

Where $aw_{i,\,j}$ is calculated as:

$$aw_{i,j} = \frac{exp(ew_{i,j})}{\sum_{t=1}^{T} exp(ew_{i,t})} \tag{14}$$

Where

$$ew_{i,j} = v_{wa} \tanh(W_w hy_{i-1} + U_w hw_j) \tag{15}$$

$$C_{ctxi} = \sum_{j=1}^{T_x} ac_{i,j} h_{cj} \tag{16}$$

Where $ac_{i,j}$ is calculated as:

$$ac_{i,j} = \frac{exp(ec_{i,j})}{\sum_{t=1}^{T_x} exp(ec_{i,t})} \tag{17}$$

Where

$$ec_{i,j} = v_{ca} \tanh(W_c hy_{i-1} + U_c hc_j) \tag{18}$$

The probability of next word y_i is:

$$p(y_i | y_1, \ldots, y_{i-1}, x) = g(y_{i-1}, hy_i, H_{ctxi}) \tag{19}$$

Where hy_i is a decoder hidden state for time step i, which is computed as:

$$hy_i = f(hy_{i-1}, y_{i-1}, H_{ctxi}) \tag{20}$$

Here f and g are nonlinear transform functions, which are implemented as GRU in the proposed model.

The third difference is the design of loss function. Apart from the cross entropy loss used in the traditional NMT model, the error rate of the word segmentation network is also considered to train the proposed model.

$$J = J_m + \gamma * J_s \tag{21}$$

here J_m refers to the loss of the NMT, J_s denotes the loss of the word segmentation and γ is the hyper-parameter which can be set by the user beforehand. We use the γ to balance the loss from the word segmentation network.

4. Experiments

We evaluate the effectiveness of our proposed model on the Chinese to English translation tasks. The open-source NMT system, DL4MT[1], is used as the baseline system, which has been used to build top-performing submission to shared translation at WMT.

4.1. Dataset and task

For the Chinese to English translation tasks, the training data consists of 1.5M pairs of sentences from LDC, which includes 32.8M Chinese words and 38.2M English words respectively. On average, each Chinese sentence contains 22 words and each English sentence contains 25 words, respectively. As the traditional RNN search model relies on the vector representations of words, we build a fixed vocabulary for each language, respectively, by choosing 30k of the most frequent words for the source language and the target language separately. Words not included in the vocabulary are replaced with “unk”. For the proposed NMT model, we build a fixed character vocabulary with the size 9000 for Chinese, which covers all of the Chinese characters in the training data. The vocabulary for English is the same with the traditional RNN search model. The English sentences are tokenized by the tokenizer script from the Moses decoder[2]. For the Chinese word segmentation task, an in-house segmenter is utilized to segment the Chinese sentences. The test data derives from the second International Chinese Word Segmentation Bakeoff. To speed up the training speed, we clean the training data by removing the sentence pairs whose source sentence contains more than 100 characters or the length of the target-side portion is over 50 words. We use the BLEU metric to evaluate the translation quality and test the performance of the proposed model on NIST03, NIST04 and NIST05. NIST02 is used as the development set. We detail the configuration of our models in Table 1.

4.2. Training detail

In our implementation, we use parallel corpus to train the RNN search baseline model on a cluster with 4 Tesla K80 GPUs and it takes about 2 days to train the model for a total of 24 epochs. Our methods are implemented with TensorFlow[3][41]. Training ends when the BLEU score on the validation set stops improving. The final models are chosen by their performance on the development set. We also use the model detailed by Luong et al. [42] as the

[1] https://github.com/nyu-dl/dl4mt-tutorial

[2] http://www.statmt.org/moses/

[3] https://www.tensorflow.org/

Table 1
The comparable model architectures.

Parameter	Char	Word	Hybrid	Hybrid+BPE	Word+BPE
Source vocab	9000	30,000	9000	9000	30,000
Target vocab	30,000	30,000	30,000	30,000	30,000
Source emb	512	512	512	512	512
Target emb	512	512	512	512	512
Source BPE num-operations				20,000	20,000
Target BPE num-operations				20,000	20,000
Encoder	1-layer 512 GRUs				
Deocder	2-layer 1024 GRUs				

Table 2
The results on the Chinese to English translation tasks.

Model	NIST02	NIST03	NIST04	NIST05	Ave.
RNNSearch-char	30.25	28.45	29.85	27.98	29.13
RNNSearch-word	35.88	33.85	34.55	32.75	34.25
Hybrid attention	37.73	36.92	37.21	34.22	36.52
Hybrid attention+sourceBPE	38.01	36.34	37.06	34.42	36.45
RNNSearch-word+bothBPE	36.98	34.75	35.83	33.45	35.25
Hybrid attention+bothBPE	38.23	37.25	38.01	35.22	37.17

strong base model, which achieved the state of the art in 2017 and called the “GNMT” model.

GNMT GNMT is implemented with TensorFlow[4]. Our training hyper-parameters are similar to the [42] experiments except for the following details. We train 4-layer GRUs of 1024 units with bidirectional encoder, embedding dim is 512.

Transformer The self-attention based Transformer model is implemented with TensorFlow[5]. Our training hyper-parameters are same to the base Transformer model. [43].

CWS For the Chinese word segmentation task, we take the in-house[6] segmenter as origin-CWS and the hybrid attention model segmenter as hybrid-CWS. We evaluate the performance of our model with a standard scoring script[7].

BPE Byte Pair Encoding is a simple data compression technique, which is highly efficient to reduce the OOV quantity by iteratively replacing the most frequent pair of bytes in a sequence with a single unused byte. In experiment, we set the target and source number operations to 20,000 and the vocabulary size to 30,000.

Optimization AdaDelta algorithm [44] is applied to optimize our models. After each updating, we clip the norm of the gradient within the block of [-1,1]. In the proposed model, the word embedding of each word in the vocabulary is regarded as part of the model’s parameters, which is initialized by Gaussian distribution or uniform distribution, same as other parameters of the model.

Dropout The dropout [45] is an extremely efficient and simple approach to prevent overfit, especially when the data set is small. We apply dropout to our model on all GRU layers and embedding layers with a fixed dropout rate of 0.1. As a result, we observe remarkable improvement when dropout is employed.

Hyper – parameter The configuration of the hyper-parameters is critical to the performance of the proposed hybrid attention NMT model, since it directly affects the generalization and regression of the model. To balance the whole information from the input characters and the composed words, we introduce two parameters, α and β, which adhere to the following equation:

$$\alpha + \beta = 1 \tag{22}$$

Different from most existing work, which simply adds two kinds of attention directly, we find that it is rather sensible to keep the sum of the hybrid attention and the original word-level attention of the traditional RNN search model in the same order of magnitude. That’s also the reason why we bind the sum of α and β to 1. Besides, experimental results also show that stronger word information can lead to better performance. In our practice, it is a good choice to set α as 0.35 and β as 0.65. The hyper-parameter γ in Eq. (21) is designed to balance the loss from the word segmentation and the machine translation. In order to highlight the translation loss, we set the γ value to 0.3 to weaken the influence of word segmentation.

4.3. Results

We compare our system with the RNNSearch models, whose inputs are fed as word-level and char-level respectively. Table 4 shows the BLEU scores on Chinese–English test sets. The RNNSearch-word is the traditional word-level RNNSearch model, which is taken as a baseline. To show the ability of our proposed character-level NMT model, we also test the performance of the RNNSearch-char model. The only difference between the RNNSearch-char and the RNNSearch-word is that the former regards the input sentence as a sequence of characters and the latter regards it as a sequence of words. From Table 4, we find that the performance of the RNNSearch-char model is rather poor. Through the comparison between the RNNSearch-char and RNNSearch-word, we notice that the traditional NMT model obtains terrible performance when the input sentence is viewed as a sequence of characters. However, compared to the RNNSearch-word, the proposed hybrid attention model can lead to +2.27 BLEU points improvement on average. Furthermore, the experimental results show that the hybrid attention model achieves more significant improvement over the RNNSearch-char model, i.e., +7.39 BLEU points on average.

To exclude the influence of OOV, we use BPE to encode the training data. It is well acknowledged that BPE can improve the translation performance especially for the rare and unseen words by representing words via subword units. From Table 4, as expected, the BPE improves the translation performance of the RNNSearch model with 1.0 BLEU point (see line 2 and line 5). With BPE applied in the source-side, the proposed hybrid attention model achieves almost the same translation performance with the hybrid attention model which is trained without BPE (see line 3 and line 4). This indicates that the hybrid attention model can deal with the OOV problem well when the BPE is not adopted in the source-side. When we apply BPE in the both-side (line 6 in Table 4), the proposed model achieves 0.6 BLEU point improvement than the hybrid attention model without BPE applied. Combined with the result of line 3 in Table 4, we can draw the following conclusion that the character-level based side, namely source-side in our hybrid attention model, does not need the help of BPE to ease the OOV problem, while the target-side with BPE applied contributes to more robust ability to handle the OOV problem,

[4] https://github.com/tensorflow/nmt
[5] https://github.com/tensorflow/tensor2tensor
[6] https://github.com/chqiwang/convseg
[7] http://www.sighan.org/bakeoff2003/score

Table 3
Comparison of results between GNMT, GNMT-hybrid and transformer.

Model	NIST02	NIST03	NIST04	NIST05	Ave.
GNMT-word	39.45	38.72	37.21	37.22	38.15
Transformer-word	40.70	39.54	39.86	38.51	39.65
GNMT-hybrid-char	41.01	40.21	39.75	38.71	39.92
Transformer-char	41.35	40.15	40.85	39.25	40.40

Table 4
The results on the WMT 14 English–German translation.

System	Architecture	BLEU
Minimum risk training NMT [46]	Gated RNN with 1 layer	20.45
Attention-based NMT [47]	LSTM with 4 layers	20.90
Linear time NMT [48]	ByteNet with 30 layers	23.75
Linearassociative unit NMT [49]	DeepLau with 4 layers	23.80
Google NMT[8]	LSTM with 8 layers	24.60
Convolutional sequence NMT [50]	CNN with 15 layers	25.16
Robust NMT [50]	Gated RNN with 1 layer	25.26
Transform NMT [43]	Self-attention with 6 layers	28.40
This work	GRU with 1 layers	20.83

since it is still based on word-level. Compared to the RNNSearch-word with BPE, our model achieves 1.9 BLEU points improvement.

5. Analysis

This section gives more detailed analysis about the proposed hybrid attention model and the experimental results.

5.1. Comparison with deep model

Our model also can be transplanted to GNMT model called "GNMT-hybrid-char", which achieves competitive results to state-of-the-art on the WMT'14 English-to-French and English-to-German benchmarks. With a sufficient number of layers, NMT has achieved comparable translation performance with the Transformer machine translation system. In our experiment, the Transformer models are exploited at word level and char level, respectively. Due to the poor performance of the character-level model, the input sentence of the GNMT model would be viewed as a sequence of word. By comparing GNMT-word and Transformer-word in Table 3, we can find that the Transformer model gains +1.5 BLEU points improvement over the GNMT-word. Although the Transformer-word model shows its superiority than the GNMT-word, our proposed GNMT-hybrid-char model, which is a character-level based hybrid attention model, leads to +1.77 BLEU points improvement on average based on the GNMT-word model. Furthermore, we find that fusing the character-level information and word-level information is helpful. The character-level information is supplementary to the word-level information. In neural machine translation, OOV words can be resorted to the character-level information, which has been shown the great effectiveness in GNMT-hybrid-char model. Moreover, Transformer-char model, which takes char sequence as input, achieves better performance than Transformer-word model. One possible hypothesis explaining this observation result is that the Transformer model can get more context information from the target and source side. So character-level information can be more effectively represented, which achieves the best performance in Table 3. And the results show that our hybrid attention can be effectively exploited in the GNMT system. Since the architecture of Transformer system is significantly different, we will consider fusing the word and character context information into the Transformer model in the next stage of research.

In Table 4, we list all the excellent systems on the WMT'14 English-to-German. The performance of these systems exceeds 20 BLEU, which can serve as excellent baselines. To use the subword unit information, we also use BPE to encode the training data. But the segmentation of the European language led to the sentence length being too long, we could only use GRU with one layers for training, which may cause performance degradation of our model. Although our system does not achieve the state-of-the-art on the WMT'14 English-German, we attempt to expand the model into hybrid word and char scheme horizontally and achieve the 20.83 BLEU, while most of other researchers are trying to deepen the model.

Table 5
The comparable results on Chinese word segmentation task.

Model	PKU			MSR		
	P	R	F	P	R	F
origin-CWS	96.1	95.2	95.7	97.4	97.3	97.3
hybrid-CWS	93.2	93.3	93.3	93.6	93.4	93.4

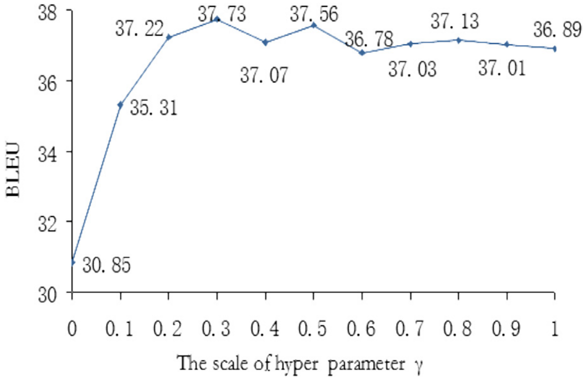
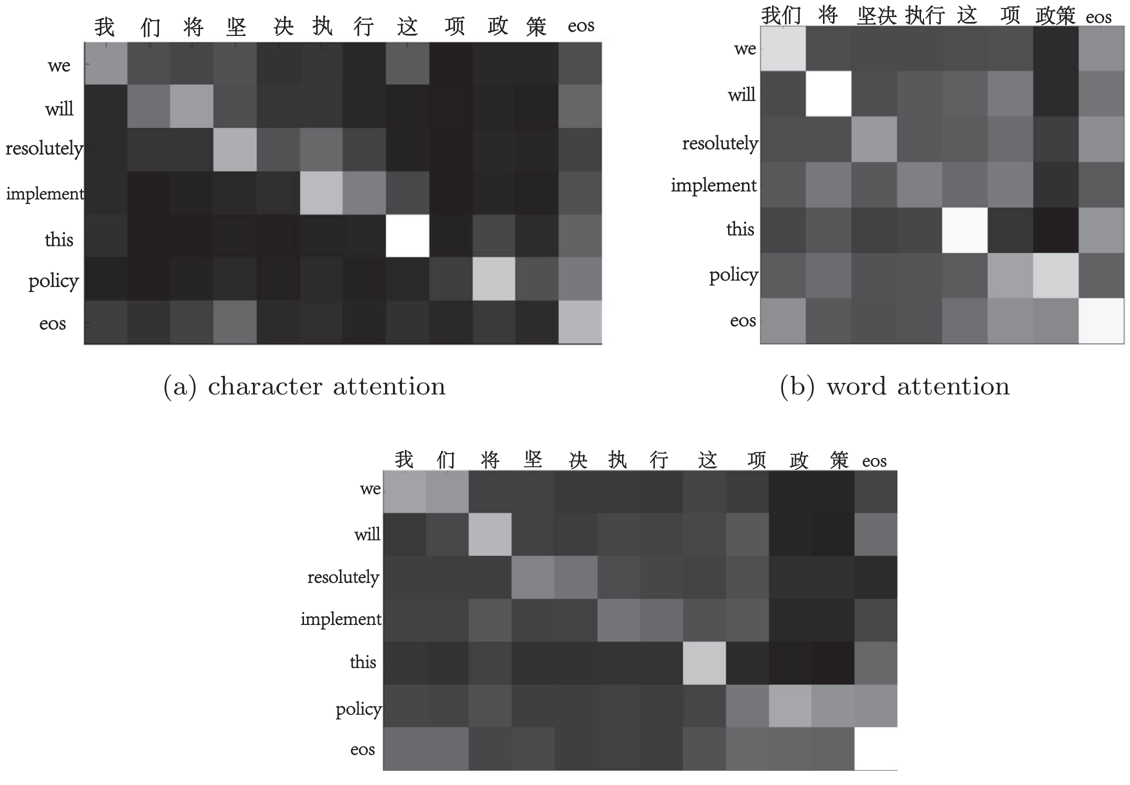


Fig. 2. The BLEU influence of the hyper parameter.

5.2. Word segmentation

To investigate the ability of the segmentation module in our hybrid attention model, we test its performance on the Chinese word segmentation tasks. We compare our segmentation module, called hybrid-CWS, with the origin-CWS. As shown in Table 3, the hybrid-CWS achieves 93.3 F-score, which is lower the F-score of origin-CWS. Although the hybrid-CWS does not achieve the F-score as high as the origin-CWS, its performance on word segmentation is acceptable. The reason can be explained as follows: the segmentation module in our hybrid attention model is repurposed to generate adapted segmentation embeddings for the neural machine translation rather than being optimized to achieve high segmentation performance. Our built-in adaptive segmentation module makes our translation model a true sense of end-to-end NMT, since our hybrid attention model no longer needs the third-party segmentation tool but achieves excellent translation results in Table 4. This is mainly because that the criterion of the state-of-the-art outside segmentation module may not be the most appropriate for the NMT.

In this experiment, we compare the effects of different γ value on the NIST02 development set. Fig. 2 shows the BLEU results. When we set the value of γ parameter to 0, without any loss back-propagation, we get pool BLEU value. The main reason is that the model can not learn any segmentation knowledge from the training set. We gradually increase the parameter value from 0 to 1. The BLEU curve ascends rapidly when the parameter value is between 0 and 0.3, and then converges to a relatively stable BLEU scope, with slightly descending trend. We achieve best BLEU performance when the parameter value equals 0.3. And the experimental results



(a) character attention

(b) word attention

(c) hybrid attention

我	们	将	坚	决	执	行	这	项	政	策
wo	men	jiang	jian	jue	zhi	xing	zhe	xiang	zheng	ce

(d) Chinese to Pin-Yin lookup table

Fig. 3. Example alignments of the character (a), word (b) and hybrid (c) attention model on a Chinese–English sentence pair. And (d) is Chinese to Pin-Yin lookup table.

show that the word segmentation task is relatively easier to converge. We finally choose 0.3 as the γ parameter value to weaken the influence of word segmentation, which highlights the translation loss due to the multi-task learning.

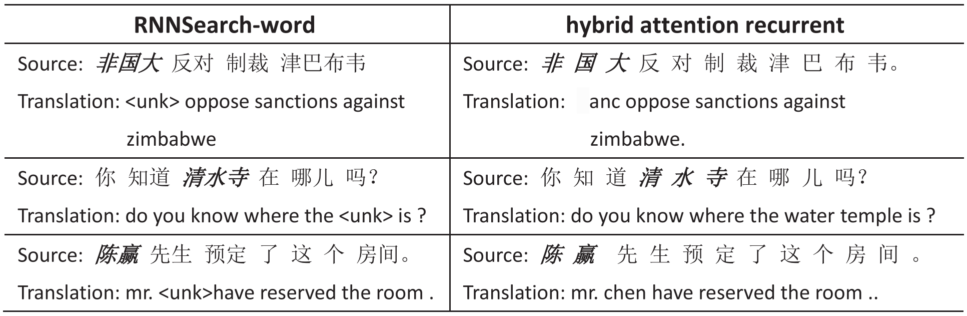
5.3. Character alignments

Experimental results in Section 4.3 show that the proposed hybrid attention model achieves significant improvement than the baseline models, which demonstrates that the composed word-level information and the character-level information is compatible or even complementary to some extent. We are curious about how these two different kinds of information coordinate with each other. To that end, we investigate how the character alignments are influenced by the word-level information and character-level information simultaneously. In our hybrid attention model, we can extract both the character-level and the word-level attention to indicate the correlation of the character and the word context. And we have:

$$C_{ctx} + W_{ctx} \propto \sum_{i=1}^{T_x} C_{att_i} + \sum_{j=1}^{T_w} W_{att_j} \tag{23}$$

where the C_{att_i} is the corresponding character-level attention weight, W_{att_j} is the corresponding word-level attention weight. To show how the word-level attention and character-level attention boost the character alignments, we calculate the whole alignment matrix by adding each element of the word-level attention matrix into the corresponding element of the character-level attention matrix. Specifically, for the target word y, the word-level attention pays its attention on the source word x with the weight of a_w. Given x includes characters $x_1, \ldots, x_l$ and the weight attention for these characters in the character-level attention is $a_{c1}, \ldots, a_{cl}$. The whole attention weight for $x_1, \ldots, x_l$ is calculated by adding a_w into $a_{c1}, \ldots, a_{cl}$, respectively.

In experiment, we extract only one-to-one alignments by selecting the source word with the highest alignment for each target word. Fig. 3 shows the alignment matrices of decoders based on different attentions, which are character-level attention, word-level attention and hybrid-level attention, respectively. Each element in the matrices indicates the alignment probability. The bigger the probability, the brighter the color. We observe that in character-level attention based alignment matrix, a target word tends to align to single character while the correct source object is a multi-character word. For example, in Fig. 3a, the target English word “we” is aligned to the first source Chinese character “wo”, while the correct source word should be the first two characters “wo men”. Similarly, the English word “policy” is aligned to the Chinese character “zheng”, while the correct source word should be two characters “zheng ce”. In other words, character-level attention cannot deal well with the source word boundary. This problem can be alleviated in word-level attention based alignment matrix. As it shown in Fig. 3b, the target English word “we” is aligned to the first source Chinese word “wo men” correctly. However, we



RNNSearch-word	hybrid attention recurrent
Source: *非国大* 反对 制裁 津巴布韦 Translation: <unk> oppose sanctions against zimbabwe	Source: *非 国 大* 反 对 制 裁 津 巴 布 韦。 Translation: anc oppose sanctions against zimbabwe.
Source: 你 知道 *清水寺* 在 哪儿 吗? Translation: do you know where the <unk> is ?	Source: 你 知 道 *清 水 寺* 在 哪 儿 吗? Translation: do you know where the water temple is ?
Source: *陈赢* 先生 预定 了 这 个 房间。 Translation: mr. <unk>have reserved the room .	Source: *陈 赢* 先 生 预 定 了 这 个 房 间 。 Translation: mr. chen have reserved the room ..

Fig. 4. The translation performance on name entity.

also observe that a target word is prone to align to several source words in Fig. 3b. For example, the English word “implement” is aligned to three Chinese words: “jiang”, “zhi xing” and “xiang”. In addition, all the three alignment probabilities are relatively low. In contrast, the hybrid-level attention based alignment matrix shows much more exact source boundary than the character-level one, which owes to the word-level attention. As it shown in 3 c, it corrects the source Chinese word of target English word “we” from “wo” in Fig. 3a to “wo men”. Meanwhile, it leads to more concentrated alignment distributions than the word-level one, which inherits the merit of character-level attention.

5.4. Rare or unknown words

This subsection gives some qualitative analysis about the proposed model. We compare the translation performance on sentences with rare or unknown words between the proposed model and the traditional word-based NMT model. In Fig. 4, we show some translation samples between the two systems. In the first example, we show the sample where the input sentence contains the rare word. We can find that the proposed model can handle this rare word very well rather than outputs the unknown token “ < unk > ” as the traditional word-based NMT model does. This is partly because that the name entity rarely occurs in the training corpus, which will usually be mapped to an unknown word by RNN search model. However, in the proposed model, the name entity is split into a sequence of characters and each character can be found in the vocabulary. In the second example, we show the example about the rare word whose meaning can be simply composed of the meaning of its individual characters. The traditional word-based NMT model generates “ < unk > ” since the rare word is not included in the vocabulary. On the contrary, our model achieves great translation performance by translating the rare word character by character. Lastly, especially for the Chinese person name, the word-segmenter often segments the person name as a word which is rarely in the vocabulary. Hence, the traditional word-based NMT model is incapable of yielding good translation. However, our model has the ability to omit the first name and translate the last name as a compromise.

6. Conclusion and future work

In this paper, we propose a novel hybrid attention NMT model which views the input sentence as a sequence of characters. With the hybrid attention, we can incorporate the composed word-level information and the character-level information into the NMT model at the same time. In addition to being applicable to languages with clear boundary between words, the proposed model also applies to languages without explicit word segmentation. Experimental results show that the proposed model achieves significant improvement than the strong baseline models. We give deep analysis about the proposed model and we find that the composed word-level information is compatible and complementary to the original input character-level information.

One limitation of our model is that the decoder is still word based. In the next stage of this research, we will investigate the character-level decoding in the target side and apply the hybrid character-level and word-level attention to Transform model. Another interesting direction is to rule out the outside segmentation information which is used as additional supervision for the word segmentation module in the proposed model.

Declaration of interests

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work is also supported by the National High Technology Research and Development Program of China (863 Program) (Grant No. 2015AA015402), the Hundred Talents Program of Chinese Academy of Sciences (No. Y3S4011D31) and National Natural Science Foundation (Grant no. 71402178).

References

[1] Kalchbrenner, Blunsom, Recurrent continuous translation models, Proceedings of the Conference on Empirical Methods in Natural Language Processing, 2013.
[2] I. Sutskever, O. Vinyals, Q.V. Le, Sequence to sequence learning with neural networks, Proceedings of the Conference on Neural Information Processing Systems, 2014.
[3] D. Bahdanau, K. Cho, Y. Bengio, Neural machine translation by jointly learning to align and translate, Proceedings of the International Conference on Learning Representations, 2015.
[4] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Comput. 9 (8) (1997) 1735–1780.
[5] K. Cho, B.V. Merriënboer, D. Bahdanau, Y. Bengio, On the properties of neural machine translation: encoder-decoder approaches, Proceedings of the Syntax, Semantics and Structure in Statistical Translation, 2014.
[6] Y. Wu, M. Schuster, Z. Chen, Q.V. Le, a.W.M. Mohammad Norouzi, M. Krikun, Y. Cao, Q. Gao, e.a. Klaus Macherey, in: Googles neural machine translation system: bridging the gap between human and machine translation, 2016. arXiv preprint arXiv: 1609.08144.
[7] J. Zhou, Y. Cao, X. Wang, P. Li, W. Xu, in: Deep recurrent models with fastforward connections for neural machine translation, 2016. arXiv preprint arXiv: 1606.04199.
[8] Y. Wu, M. Schuster, Z. Chen, Q.V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey, et al., in: Google’s neural machine translation system: Bridging the gap between human and machine translation, 2016. arXiv preprint arXiv: 1609.08144.
[9] M. Johnson, M. Schuster, Q.V. Le, M. Krikun, Y. Wu, Z. Chen, N. Thorat, F. Viégas, M. Wattenberg, G. Corrado, et al., in: Google’s multilingual neural machine translation system: enabling zero-shot translation, 2016. arXiv preprint arXiv: 1611.04558.
[10] I. Sutskever, O. Vinyals, Q.V. Le, Sequence to sequence learning with neural networks, Advances in neural information processing systems, 2014, pp. 3104–3112.

[11] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, Y. Bengio, in: Learning phrase representations using RNN encoder-decoder for statistical machine translation, 2014. arXiv preprint arXiv: 1406.1078.

[12] D. Bahdanau, K. Cho, Y. Bengio, in: Neural machine translation by jointly learning to align and translate, 2014. arXiv preprint arXiv: 1409.0473.

[13] T. Cohn, C.D.V. Hoang, E. Vymolova, K. Yao, C. Dyer, G. Haffari, in: Incorporating structural alignment biases into an attentional neural translation model, 2016. arXiv preprint arXiv: 1601.01085.

[14] M.-T. Luong, I. Sutskever, Q.V. Le, O. Vinyals, W. Zaremba, in: Addressing the rare word problem in neural machine translation, 2014. arXiv preprint arXiv: 1410.8206.

[15] A. Mnih, K. Kavukcuoglu, Learning word embeddings efficiently with noise-contrastive estimation, in: Proceedings of the Advances in Neural Information Processing Systems, 2013, pp. 2265–2273.

[16] S. Jean, K. Cho, R. Memisevic, et al., On using very large target vocabulary for neural machine translation (2014) arXiv preprint arXiv:1412.2007.

[17] Z. Yang, W. Chen, F. Wang, B. Xu, A character-aware encoder for neural machine translation, Proceedings of the COLING, 2016.

[18] R. Sennrich, B. Haddow, A. Birch, in: Neural machine translation of rare words with subword units, 2015. arXiv preprint arXiv: 1508.07909.

[19] W. Ling, I. Trancoso, C. Dyer, A.W. Black, in: Character-based neural machine translation, 2015. arXiv preprint arXiv: 1511.04586.

[20] J. Chung, K. Cho, Y. Bengio, in: A character-level decoder without explicit segmentation for neural machine translation, 2016. arXiv preprint arXiv: 1603.06147.

[21] W. Ling, T. Luís, L. Marujo, R.F. Astudillo, S. Amir, C. Dyer, A.W. Black, I. Trancoso, in: Finding function in form: compositional character models for open vocabulary word representation, 2015. arXiv preprint arXiv: 1508.02096.

[22] X. Zhang, J. Zhao, Y. LeCun, Character-level convolutional networks for text classification, in: Proceedings of the Advances in Neural Information Processing Systems, 2015, pp. 649–657.

[23] D. Golub, X. He, in: Character-level question answering with attention, 2016. arXiv preprint arXiv: 1604.00727.

[24] M.-T. Luong, C.D. Manning, in: Achieving open vocabulary neural machine translation with hybrid word-character models, 2016. arXiv preprint arXiv: 1604.00788.

[25] J. Lee, K. Cho, T. Hofmann, in: Fully character-level neural machine translation without explicit segmentation, 2016. arXiv preprint arXiv: 1610.03017.

[26] N. Xue, et al., Chinese word segmentation as character tagging, Comput. Linguist. Chin. Lang. Process. 8 (1) (2003) 29–48.

[27] F. Peng, F. Feng, A. McCallum, Chinese segmentation and new word detection using conditional random fields, in: Proceedings of the 20th International Conference on Computational Linguistics, Association for Computational Linguistics, 2004, p. 562.

[28] A.L. Berger, V.J.D. Pietra, S.A.D. Pietra, A maximum entropy approach to natural language processing, Comput. Linguist. 22 (1) (1996) 39–71.

[29] J. Lafferty, A. McCallum, F.C.N. Pereira, Conditional random fields: Probabilistic models for segmenting and labeling sequence data (2001).

[30] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, P. Kuksa, Natural language processing (almost) from scratch, J. Mach. Learn. Res. 12 (1) (2011) 2493–2537.

[31] X. Zheng, H. Chen, T. Xu, Deep learning for Chinese word segmentation and pos tagging, in: Proceedings of the Conference on Empirical Methods in Natural Language Processing, 2013.

[32] X. Chen, X. Qiu, C. Zhu, et al., Gated recursive neural network for Chinese word segmentation., in: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), 1, 2015, pp. 1744–1753.

[33] X. Chen, X. Qiu, C. Zhu, P. Liu, X. Huang, Long short-term memory neural networks for Chinese word segmentation, in: Proceedings of the Conference on Empirical Methods in Natural Language Processing, 2015, pp. 1197–1206.

[34] G. Andrew, A hybrid Markov/semi-Markov conditional random field for sequence segmentation, in: Proceedings of the Conference on Empirical Methods in Natural Language Processing, 2006, pp. 465–472.

[35] Y. Zhang, S. Clark, Chinese segmentation with a word-based perceptron algorithm., in: Proceedings of the Meeting of the Association for Computational Linguistics, June 23–30, 2007, Prague, Czech Republic, 2007.

[36] X. Sun, Y. Zhang, T. Matsuzaki, Y. Tsuruoka, J. Tsujii, A discriminative latent variable Chinese segmenter with hybrid word/character information, in: Proceedings of the Human Language Technologies: the 2009 Conference of the North American Chapter of the Association for Computational Linguistics, 2009, pp. 56–64.

[37] Y. Liu, W. Che, J. Guo, et al., Exploring segment representations for neural segmentation models (2016) arXiv preprint arXiv:1604.05499.

[38] M. Zhang, Y. Zhang, G. Fu, Transition-based neural word segmentation, in: Proceedings of the Meeting of the Association for Computational Linguistics, 2016, pp. 421–431.

[39] W. Sun, Word-based and character-based word segmentation models: comparison and combination, in: Proceedings of the 23rd International Conference on Computational Linguistics: Posters, Association for Computational Linguistics, 2010, pp. 1211–1219.

[40] M. Wang, R. Voigt, C.D. Manning, Two knives cut better than one: Chinese word segmentation with dual decomposition, in: Proceedings of the Meeting of the Association for Computational Linguistics, 2014, pp. 193–198.

[41] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, e.a. Sanjay Ghemawat, in: Tensorflow: A system for large-scale machine learning, 2015. arXiv preprint arXiv: 1605.08695.

[42] M. Luong, E. Brevdo, R. Zhao, in: Neural machine translation (seq2seq) tutorial, 2017. https://github.com/tensorflow/nmt

[43] A. Vaswani, N. Shazeer, N. Parmar, et al., Attention is all you need, Advances in neural information processing systems, 2017, pp. 5998–6008.

[44] M.D. Zeiler, in: Adadelta: an adaptive learning rate method, 2012. arXiv preprint arXiv: 1212.5701.

[45] N. Srivastava, G.E. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Dropout: a simple way to prevent neural networks from overfitting., J. Mach. Learn. Res. 15 (1) (2014) 1929–1958.

[46] S. Shen, Y. Cheng, Z. He, W. He, H. Wu, M. Sun, Y. Liu, in: Minimum risk training for neural machine translation, 2015. arXiv preprint arXiv: 1512.02433.

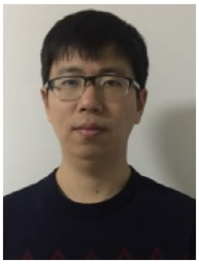
[47] M.T. Luong, H. Pham, C.D. Manning, Effective approaches to attention-based neural machine translation (2015) arXiv preprint arXiv:1508.04025.

[48] N. Kalchbrenner, L. Espeholt, K. Simonyan, et al., Neural machine translation in linear time (2016) arXiv preprint arXiv:1610.10099.

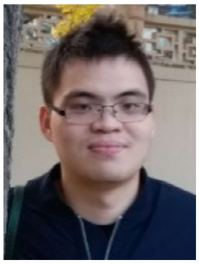
[49] M. Wang, Z. Lu, J. Zhou, et al., Deep neural machine translation with linear associative unit (2017) arXiv preprint arXiv:1705.00861.

[50] J. Gehring, M. Auli, D. Grangier, et al., Convolutional sequence to sequence learning, in: Proceedings of the 34th International Conference on Machine Learning-Volume 70. JMLR. org, 2017, pp. 1243–1252.

Feng Wang received Ph.D. degree in Institute of Automation, Chinese Academy of Science. Now he is an assistant professor at Institute of Automation of Chinese Academy Sciences. His interests include NLP and ASR.



Wei Chen received B.S. degree in Harbin Institute of Technology, Ph.D. degree in Institute of Automation, Chinese Academy of Science, and now is an assistant professor at Institute of Automation, Chinese Academy of Science. His research interests include NLP, ASR and MT.



Zhen Yang is a Ph.D. candidate in the Institute of Automation of Chinese Academy Sciences (CASIA), Beijing, China. His research interests include deep learning, machine translation and generative adversarial net.

Shuang Xu received Ph.D. degree in Institute of Automation, Chinese Academy of Science. Now She is an associated professor at Institute of Automation of Chinese Academy Sciences. Her interests include NLP and ASR.

Bo Xu is a Professor at the Institute of Automation of Chinese Academy of Sciences, Beijing, China. He received his B.S. degree in Zhejiang University and his Ph.D. degree in Institute of Automation of Chinese Academy of Sciences. His interests include NLP, ASR and human-like intelligence.