The Thirty-Second AAAI Conference
on Artificial Intelligence (AAAI-18)

Attention-via-Attention Neural Machine Translation

Shenjian Zhao
Department of Computer Science and Engineering
Shanghai Jiao Tong University
sword.york@gmail.com

Zhihua Zhang
Peking University
Beijing Institute of Big Data Research
zhzhang@math.pku.edu.cn

Abstract

Since many languages originated from a common ancestral language and influence each other, there would inevitably exist similarities between these languages such as lexical similarity and named entity similarity. In this paper, we leverage these similarities to improve the translation performance in neural machine translation. Specifically, we introduce an *attention-via-attention* mechanism that allows the information of source-side characters flowing to the target side directly. With this mechanism, the target-side characters will be generated based on the representation of source-side characters when the words are similar. For instance, our proposed neural machine translation system learns to transfer the character-level information of the English word '`system`' through the attention-via-attention mechanism to generate the Czech word '`systém`'. Consequently, our approach is able to not only achieve a competitive translation performance, but also reduce the model size significantly.

1 Introduction

A language family is a group of related languages that developed from a common ancestral language, such as the Indo-European family, the Niger-Congo family and the Austronesian family. The languages in the same family are more or less similar to each other. One of the measurements is lexical similarity (Simons and Fennig 2017), which approximately measures the similarity between the lexicons of two languages. Simons and Fennig (2017) calculated it by comparing a standardized set of wordlists and counting those forms that show similarity in both form and meaning. Based on such a method, English is evaluated to have a lexical similarity of 60% with German and 27% with French. Moreover, language itself is an evolving system and the evolution of lexicons in different languages never stops. Guestwords, foreignisms and loanwords from a language may be added to the lexicon of another language. Although the languages are different, many of the words (e.g., named entities) are usually represented by similar characters.

Currently, many state-of-the-art neural machine translation (NMT) systems (Bahdanau, Cho, and Bengio 2015; Sutskever, Vinyals, and Le 2014; Jean et al. 2015; Luong and Manning 2016) are built on words. There are various considerations behind the wide adoption of word-level modeling (Chung, Cho, and Bengio 2016). The vanishing gradient problem of character-level models and lower calculation cost of word-level models may be the major causes. However, the word-level NMT systems are unable to utilize the lexical similarity and named entity similarity between language pairs. Some explorations have been performed to incorporate the similarity of vocabularies. For instance, Gulcehre et al. (2016) introduced a pointer network to copy words from the source. However, they assumed that the target out-of-vocabulary (OOV) words are the same as the corresponding source words. Obviously, this assumption is not always satisfied.

The character-level information is critical in neural machine translation. Suppose there are two languages that differ only in the alphabets, e.g., Russian written in Cyrillic and Russian written in Latin script. It would not be easy for a purely word-level NMT to translate between such a language pair, because the word-level model needs to establish the mapping between words. In contrast, the character-level model only needs to establish the mapping between characters. Although there are no such language pairs in reality, we can still make use of the similarity of languages from the character level. In particular, we provide the following two sentences to clarify what we are focusing on:

1) Aby legenda byla věrohodná , psalo se o filmovém projektu ve specializovaných magazínech , pořádaly se tiskové konference , fiktivní produkční společnost měla reálnou kancelář . (Czech)

2) For the story to be believed , the film project was reported on in specialist magazines , press conferences were organised , and the fictitious production company had a real office . (English)

There are many similar words between two sentences. One may speculate the meaning of some Czech words based on the English words such as '`projektu - project`', '`magazínech - magazines`' and '`konference - conferences`'. In this paper, we leverage these similarities from character level in NMT to improve the translation performance, and reduce the model size simultaneously. In accordance with expectation, our model is able to detect and handle named entities, as shown in Section 6.

To sum up the above statements, the character-level lexicon

and the word-level grammar are both important for neural machine translation. Luong and Manning (2016) proposed a hybrid model on the English-to-Czech translation task which encodes the OOV words using a character-level RNN. However, this hybrid model is restricted to achieving open vocabularies, and the character-level information has not been exploited. Instead, we propose a model that takes advantage of word-level modeling and bridges lexicons with an *attention-via-attention* mechanism, without even involving any vocabularies. Specifically, we encode the source sentence from character level using an unidirectional recurrent neural network (RNN), then extract the word information to learn word-level grammar by a bidirectional RNN (BiRNN) (Schuster and Paliwal 1997). To predict at character level, the attention is paid to the word level first. Subsequently, the attention is turned to the character level with the help of the word-level attention. Finally, the word-level representation and character-level representation are combined together to predict the target character.

We illustrate the architecture of our model in Figure 1, from which we could find that the information of source-side characters flows to the target-side characters directly. There are many models employing multiple attention components, such as attention-over-attention neural networks (Cui et al. 2016), hierarchical attention networks (Yang et al. 2016) and multi-step attention (Gehring et al. 2017). The key difference is that our attention-via-attention mechanism is a top-down approach (from words to characters), while the others use a down-top approach, that is, to build the attention from a lower-level representation to a higher-level one. The hierarchical attention could not connect the source side and the target side directly, thus it is not applicable to this scenario.

With a hierarchical encoder and an attention-via-attention mechanism, our method is capable of addressing several essential issues in neural machine translation community. That is,

- We avoid the use of large vocabularies. Instead, we employ a character-level RNN to encode the entire source sentence which also handles the rare words. The character-level RNN makes use of distributed representation, which generally yields better generalization. It is one of the key ingredients for the attention-via-attention mechanism.
- We alleviate the vanishing gradient problem of purely character-level models by introducing a hierarchical encoder.
- We detect named entities and similar lexemes automatically, then transfer them to the target language through the attention-via-attention mechanism.

These issues impact not only on translation tasks but also on many other natural language processing tasks, such as text summarization (Gulcehre et al. 2016) and conversational models (Vinyals and Le 2015). Thus these tasks may benefit from our approach in principle.

2 Neural Machine Translation

Neural machine translation systems are typically implemented as an encoder-decoder architecture (Bahdanau, Cho, and Bengio 2015; Sutskever, Vinyals, and Le 2014). The encoder could be a recurrent neural network or a bidirectional recurrent neural network that encodes a source language sentence $x = \{x_1, \ldots, x_{T_c}\}$ into a sequence of hidden states $\mathbf{h} = \{\mathbf{h}_1, \ldots, \mathbf{h}_{T_c}\}$:

$$\mathbf{h}_t = f_{\text{enc}}(e(x_t), \mathbf{h}_{t-1}),$$

in which $\mathbf{h}_t$ is the hidden state at time step t, $e(x_t)$ is the continuous embedding of x_t, T_c is the number of symbols in the source sequence, and the function f_{enc} is the recurrent unit such as the gated recurrent unit (GRU) (Chung et al. 2014) or the long short-term memory (LSTM) unit (Hochreiter and Schmidhuber 1997). The decoder, another RNN, is trained to predict the conditional probability of each target symbol y_t given its preceding symbols $y_{<t}$ and the context vector $\mathbf{c}_t$:

$$\begin{aligned} P(y_t|y_{<t}) &= g(e(y_{t-1}), \mathbf{r}_{t-1}, \mathbf{c}_t), \\ \mathbf{r}_t &= f_{\text{dec}}(e(y_t), \mathbf{r}_{t-1}, \mathbf{c}_t), \end{aligned}$$

where $\mathbf{r}_t$ is the hidden state of the decoder RNN at time step t and updated by f_{dec}, $e(y_t)$ is the continuous embedding of target symbol y_t, and g is a nonlinear function that computes the probability of y_t. The context vector $\mathbf{c}_t$ at each decoding time step is computed as a weighted sum of source hidden states (Bahdanau, Cho, and Bengio 2015), e.g.,

$$\begin{aligned} \mathbf{c}_t &= \sum_{i=1}^{T_c} \alpha_i \mathbf{h}_i, \\ \alpha_i &= \frac{\exp(f_{\text{energy}}(\mathbf{r}_{t-1}, \mathbf{h}_i))}{\sum_{j=1}^{T_c} \exp(f_{\text{energy}}(\mathbf{r}_{t-1}, \mathbf{h}_j))}, \end{aligned}$$

where f_{energy} is a feed-forward network, computing how well the representation $\mathbf{h}_i$ of source symbol matches the hidden state $\mathbf{r}_{t-1}$ of the decoder RNN. Specifically, we use the following function,

$$f_{\text{energy}}(\mathbf{r}_{t-1}, \mathbf{h}_i) = \mathbf{v}_e^T \tanh(\mathbf{W}_r \mathbf{r}_{t-1} + \mathbf{W}_h \mathbf{h}_j), \quad (1)$$

where $\mathbf{v}_e$, $\mathbf{W}_r$ and $\mathbf{W}_h$ are trainable parameters. Luong, Pham, and Manning (2015) have used several alternatives to compute the energy.

The end-to-end model is then jointly trained to maximize the conditional log-likelihood:

$$\mathbf{\Theta}^* = \operatorname*{argmax}_{\mathbf{\Theta}} \sum_{(x,y) \in D} \sum_{t=1}^{T_y} \log P(y_t|y_{<t}, x; \mathbf{\Theta}), \quad (2)$$

where $\mathbf{\Theta}$ is the parameters of the model and (x, y) corresponds to a sentence pair in the dataset D.

3 Attention-via-Attention Neural Machine Translation

We could either use the character symbols or the word symbols as the inputs of NMT systems. Both character-level models and word-level models have their own merits and demerits (Chung, Cho, and Bengio 2016). We devise two novel components which utilize the advantages of both modeling methods: the hierarchical encoder and the attention-via-attention mechanism. Accordingly, we propose an attention-via-attention

neural machine translation (AvA NMT) model. Figure 1 illustrates the general architecture of our NMT system. The proposed model will be described formally in the following sections.

Hierarchical Encoder

As described by Chung, Cho, and Bengio (2016), many issues of word-level translation could be elegantly addressed by using a parametric approach based on neural networks instead of a non-parametric count-based approach. For instance, Lee, Cho, and Hofmann (2016) proposed a fully character-level model which is comparable to the subword-based models (Sennrich, Haddow, and Birch 2016). However, the convolutional neural network (CNN) encoder (Kim et al. 2016) used in their model is not suitable for utilizing the similarity of lexicons as the character-level information is filtered. Chung, Ahn, and Bengio (2016) proposed a hierarchical multiscale RNN which outperforms the standard RNN in character-level language modeling. Unfortunately, it suffers from inefficiency and is much slower than the word-level encoder. To utilize the similarity between languages and encode efficiently, we devise a hierarchical RNN encoder which consists of a small character-level RNN and a large word-level BiRNN.

Character-Level RNN. We encode the source sentence $x = \{x_1, \ldots, x_{T_c}\}$ with a character-level RNN as

$$\mathbf{h}_t^c = f_{\text{char\_rnn}}(e(x_t), \mathbf{h}_{t-1}^c),$$

where $e(x_t)$ is the continuous embedding of character x_t and the function $f_{\text{char\_rnn}}$ is the recurrent unit. The hidden state $\mathbf{h}_t^c$ should be able to summarize the preceding character sequence. Since the primary function of this character-level RNN is to generate a continuous representation of words, we employ a recurrent neural network containing fewer hidden units for efficiency. In contrast to the CNN, it is much easier for a character-level RNN to generate a reasonable representation of the substrings such as '`Exp`', '`Expe`' and '`Exper`'. As explained in Section 3, this feature is essential for bridging lexicons between the source language and the target language. We utilize all these hidden states to form the context set $\mathcal{C}^{\text{c}} = \{\mathbf{h}_1^{\text{c}}, \ldots, \mathbf{h}_{T_c}^{\text{c}}\}$ of the character-level sequence.

Word-Level BiRNN. After encoding the entire sentence with the character-level RNN, we are able to obtain the representation of each word. We extract the hidden states according to the spaces in the sentences. For instance, $\mathbf{h}_7^c$, $\mathbf{h}_{14}^c$ and $\mathbf{h}_{16}^c$ are extracted, representing 'Expert', 'system' and '</s>' respectively. Obviously, the word-level sequence is much shorter than the character-level sequence. Therefore we could employ a large BiRNN to capture the semantic information, whose overhead is similar to that of purely word-level models. It consists of two RNNs: the forward network $f_{\text{word\_forw}}$ and the backward network $f_{\text{word\_back}}$. The hidden states from both networks are concatenated at each time step. Formally, the hidden state $\mathbf{h}_t^c$ is encoded by the following steps:

$$\begin{aligned}
\mathbf{h}_t^{\text{fw}} &= f_{\text{word_forw}}(\mathbf{h}_t^c, \mathbf{h}_{t-1}^{\text{fw}}), \\
\mathbf{h}_t^{\text{bw}} &= f_{\text{word_back}}(\mathbf{h}_t^c, \mathbf{h}_{t+1}^{\text{bw}}), \\
\mathbf{h}_t^{\text{w}} &= [\mathbf{h}_t^{\text{fw}}; \mathbf{h}_t^{\text{bw}}].
\end{aligned}$$

All these hidden states form the context set $\mathcal{C}^{\text{w}} = \{\mathbf{h}_1^{\text{w}}, \ldots, \mathbf{h}_{T_w}^{\text{w}}\}$ of the word-level sequence containing T_w words. Since the word-level sequence is much shorter, it is possible to utilize the deep multi-layer architecture such as in (Luong and Manning 2016).

Attention via Attention

There are many models employing a hierarchical attention, such as attention-over-attention neural networks (Cui et al. 2016), hierarchical attention networks (Yang et al. 2016) and multi-step attention (Gehring et al. 2017). It is worth mentioning that the hierarchical attention mechanism in the previous work is built from a lower-level representation to a higher-level one. However, in our work, we build it reversely. We would first attend to the higher level and then attend to the lower level guided by the higher-level attention, so-called *attention-via-attention* mechanism. In our model, the higher-level representation is the word-level representation while the lower-level representation is the character-level representation. First, we obtain the context vector of the word level which is similar to RNNsearch (Bahdanau, Cho, and Bengio 2015), that is,

$$\begin{aligned}
\mathbf{c}_t^w &= \sum_{i=1}^{T_w} \alpha_i^w \mathbf{h}_i^w, \\
\alpha_i^w &= \frac{\exp(f_{\text{energy}}^w(\mathbf{r}_{t-1}, \mathbf{h}_i^w))}{\sum_{j=1}^{T_w} \exp(f_{\text{energy}}^w(\mathbf{r}_{t-1}, \mathbf{h}_j^w))},
\end{aligned}$$

where f_{energy}^w is a feed-forward network described by Eqn. (1) and $\mathbf{r}_t$ is the hidden state of a character-level decoder RNN, which will be described in Section 3.

Next we pay attention to the character-level context. As the sequence of characters is much longer than the sequence of words, it would be much harder to obtain the corresponding representation. We would utilize the context vector $\mathbf{c}_t^w$ from the word level. Suppose we are translating the word '`Expert`' as illustrated in Figure 1. The context $\mathbf{c}_t^w$ would roughly point out the source word '`Expert`' as it shares more similarities with the substrings in '`Expert`' than '`system`'. We also utilize the hidden states $\mathbf{r}_5$ of the decoder recurrent network. In this case, we could find that $\mathbf{r}_5$ summarizes '`Expe`' in the target side and $\mathbf{h}_4^c$ summarizes '`Expe`' in the source side. $\mathbf{r}_5$ and $\mathbf{h}_4^c$ are similar in some sense, thus it would be helpful for the content-based addressing. Formally, the context vector of character level is computed by

$$\begin{aligned}
\mathbf{c}_t^c &= \sum_{i=1}^{T_c} \alpha_i^c \mathbf{h}_i^c, \\
\alpha_i^c &= \frac{\exp(f_{\text{energy}}^c(\mathbf{r}_{t-1}, \mathbf{c}_t^w, \mathbf{h}_i^c))}{\sum_{j=1}^{T_c} \exp(f_{\text{energy}}^c(\mathbf{r}_{t-1}, \mathbf{c}_t^w, \mathbf{h}_j^c))}.
\end{aligned}$$

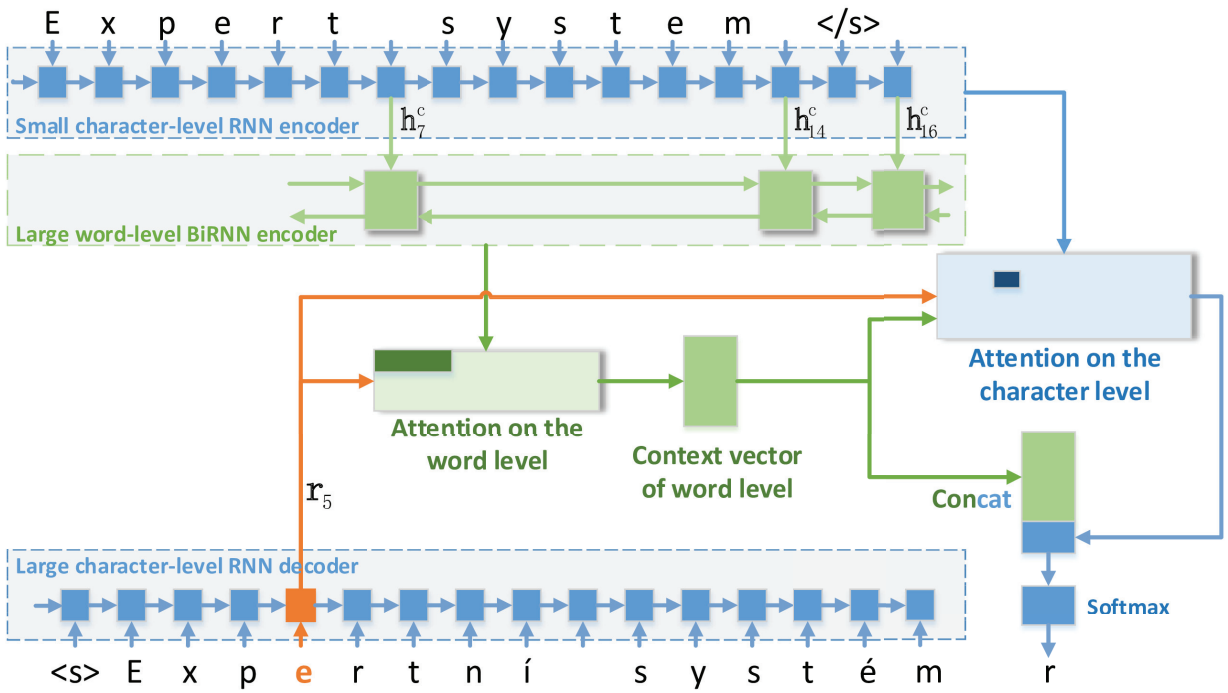


Figure 1: Attention-via-attention neural machine translation. As 'Expe' has already been generated, next target character 'r' will be generated based mainly on the representation of 'r' in the source side. For brevity, we omit certain connections in the graphical illustration.

In this character-level attention step, $f^c_{\text{energy}}(\mathbf{r}_{t-1}, \mathbf{c}^w_t, \mathbf{h}^c_i)$ is calculated by,

$$f_{\text{energy}}(\mathbf{r}_{t-1}, \mathbf{c}^w_t, \mathbf{h}^c_i) = \mathbf{v}^T_{ce} \tanh(\mathbf{W}_{cr}\mathbf{r}_{t-1} + \mathbf{W}_{cc}\mathbf{c}^w_t + \mathbf{W}_{ch}\mathbf{h}^c_j),$$

where $\mathbf{v}_{ce}$, $\mathbf{W}_{cr}$, $\mathbf{W}_{cc}$ and $\mathbf{W}_{ch}$ are trainable parameters.

Finally, the character-level context vector $\mathbf{c}^c_t$ and the word-level context vector $\mathbf{c}^w_t$ are concatenated into $\mathbf{c}_t = [\mathbf{c}^w_t; \mathbf{c}^c_t]$, which is incorporated to predict the next target character.

The idea of the attention-via-attention mechanism should be applicable to the purely word-level model in which different levels of abstraction correspond to different layers of RNNs in the multi-layer BiRNN encoder. It can be regarded as adding attentive shortcuts between layers.

Character-Level Decoder

A character-level decoder is essential for bridging lexicons of the source language and the target language. The trivial character-level RNNs are able to achieve competitive translation performance (Luong and Manning 2016; Chung, Cho, and Bengio 2016). We predict the conditional probability of the target character y_t based on its preceding characters $y_{<t}$, the context vector $\mathbf{c}_t$, and the hidden state $\mathbf{r}_{t-1}$ of decoder RNN; that is

$$P(y_t|y_{<t}) = g(e(y_{t-1}), \mathbf{r}_{t-1}, \mathbf{c}_t),$$
$$\mathbf{r}_t = f_{\text{dec}}(e(y_t), \mathbf{r}_{t-1}, \mathbf{c}_t).$$

where the function f_{dec} is the recurrent unit such as GRU or LSTM, and g is a nonlinear function evaluating the probability of y_t. Finally, the whole NMT system could be jointly trained in terms of the problem in (2).

Complexity Analysis

Our model is a fully character-based model that depends only on the character vocabulary. Thus both source sequences and target sequences become much longer than the word-based models, and the model becomes more complex. We would analyze the complexity of our attention-via-attention neural machine translation model briefly.

Suppose the character-level RNN encoder contains $N_{\text{char\_rnn}}$ hidden units, then the complexity to encode a character-level sequence is $\mathcal{O}(T_c N^2_{\text{char\_rnn}})$. Similarly, the complexity of word-level encoder is $\mathcal{O}(T_w N^2_{\text{word\_forw}} + T_w N^2_{\text{word\_back}})$. The complexity of word-level attention and character-level attention are $O(T_w T_y N_{\text{word\_context}}(N_{\text{word\_forw}} + N_{\text{word\_back}}))$ and $O(T_c T_y N_{\text{char\_context}} N_{\text{char\_rnn}})$, respectively. Finally, the complexity of the character-level decoder mainly consists of complexity of the RNN and complexity of evaluating probability via the nonlinear function g, i.e., the softmax function. Therefore, the total complexity of decoder is $O(T_y N^2_{\text{dec\_char}}) + O(T_y N_{\text{dec\_char}} N_{\text{char\_vocab\_size}})$. We summarize the complexity of various models in Table 1 for comparison, we assume that $N_{\text{word\_forw}} = N_{\text{word\_back}}$.

Based on the statistics that sentences are on average 6 longer (i.e., $T_c = 6T_w$) when represented in characters (for English, French and Czech) and our experiments

Model	Complexity	
Word-based	$O(2T_w N^2_{\text{word\_forw}})$	+
	$O(2T_w T_y N_{\text{word\_context}} N_{\text{word\_forw}})$	+
	$O(T_y N^2_{\text{dec\_word}})$	+
	$O(T_y N_{\text{dec\_word}} N_{\text{word\_vocab\_size}})$	
bpe2char	$O(2T_w N^2_{\text{word\_forw}})$	+
	$O(2T_w T_y N_{\text{word\_context}} N_{\text{word\_forw}})$	+
	$O(T_y N^2_{\text{dec\_char}})$	+
	$O(T_y N_{\text{dec\_char}} N_{\text{char\_vocab\_size}})$	
AvA NMT	$\mathcal{O}(T_c N^2_{\text{char\_rnn}}) + O(2T_w N^2_{\text{word\_forw}})$	+
	$O(T_c T_y N_{\text{char\_context}} N_{\text{char\_rnn}})$	+
	$O(2T_w T_y N_{\text{word\_context}} N_{\text{word\_forw}})$	+
	$O(T_y N^2_{\text{dec\_char}})$	+
	$O(T_y N_{\text{dec\_char}} N_{\text{char\_vocab\_size}})$	

Table 1: Complexity of various models.

setting that $N_{\text{word\_forw}} = 2N_{\text{char\_rnn}}$ and $N_{\text{word\_context}} = 4N_{\text{char\_context}}$, the extra overhead of AvA NMT model comparing to bpe2char model is $O(1.5T_w N^2_{\text{word\_forw}}) + O(0.75T_w T_y N_{\text{word\_context}} N_{\text{word\_forw}})$. This overhead is insignificant compared with the whole complexity.

4 Experiments

We evaluate the effectiveness and the efficiency of the proposed attention-via-attention model on the WMT'15 En-Fr and En-Cs translation tasks. [1] We conduct comparison with various strong baselines including RNNsearch (Bahdanau, Cho, and Bengio 2015), GNMT (Wu et al. 2016), bpe2char models (Chung, Cho, and Bengio 2016), char2char models (Lee, Cho, and Hofmann 2016) and hybrid models (Luong and Manning 2016). For fair comparison, two metrics are used: BLEU (Papineni et al. 2002) and chrF$_3$ (Popovic 2015)[2].

Datasets

We use the parallel corpora from WMT. When comparing with RNNsearch on En-Fr task, we reduce the size of the combined corpus to have 12.1M sentence pairs for fairness. When comparing with GNMT, we use the whole dataset which contains 36M parallel sentences. For En-Cs, we use all parallel corpora available for WMT'15. In terms of preprocessing, we only apply the usual tokenization in comparison with the other NMT systems. We choose a list of 300 most frequent characters for each language which covers nearly all of the training data. A great advantage over the hybrid model (Luong and Manning 2016) is that we do not use any word vocabularies in our model. We use *newstest2013* as the development set and evaluate the models on *newstest2014* and *newstest2015* for En-Fr and En-Cs task, respectively. We do not use any monolingual corpus.

Training and Decoding Details

Models. First we want to verify the effectiveness of the hierarchical encoder and the attention-via-attention mechanism, by comparing with RNNsearch (Bahdanau, Cho, and Bengio 2015) on En-Fr translation task. Concretely, we follow Bahdanau, Cho, and Bengio to use similar architectures. Both the BiRNN encoder and the RNN decoder consist of one-layer GRUs, each having 1024 hidden units. We only use those pairs in which the sentence is not longer than 300 characters. Because the capacity of an embedding matrix is much higher than the character-level RNN, we use one-layer 512 LSTMs in this RNN. In order to comparing with GNMT, we employ a much deeper model which consists of a four-layer encoder and two-layer decoder.

Czech is a Slavic language with not only rich and complex inflection but also fusional morphology, which is more challenging. To demonstrate the efficiency of the proposed model on such a challenging language, we have constrained our shallow AvA model to have similar capacity with the other character-level models. Besides, we also trained a deep AvA model in order to comparing with the deep hybrid model . We provide the detailed setting in Table 3.

Training Details. We use the ADAM optimizer (Kingma and Ba 2015) with minibatch of 100 sentences to train each model. The learning rate is first set to $5e^{-4}$ and then halved every epoch. The norm of the gradient is clipped with a threshold of 1. We train each shallow model for approximately 2 weeks on a single Titan X GPU. However, the deep AvA NMT model takes longer time to train. We list the rough training days on GPUs in Table 2 and Table 3, which are estimated by the number of GPUs multiply by training days. Note that, the GPU days may be inaccurate because of different hardwares and different implementation.

Decoding Details. In case of decoding, we use beam search with length-normalization to find a translation. The beam width is set to 12 for all models.

5 Quantitative Results

In this section, we conduct comparison of quantitative results on the En-Fr and En-Cs translation tasks, which is evaluated using BLEU and chrF$_3$. As our model is purely character-based, we focus on comparison of En-Cs translation tasks which is more appropriate by character-level modeling. Moreover, we approximately estimate the similarity between these languages based on the alignments on the character level.

Translation Performance

En-Fr task. We list the BLEU scores on the En-Fr task in Table 2. From the table, we easily reach the conclusion that the proposed AvA NMT model outperforms RNNsearch despite of much smaller model and fewer training epochs, confirming the effectiveness of our model. Further more, our deep AvA NMT model is comparable to the GNMT model

[1] http://www.statmt.org/wmt15/translation -task.html

[2] We use the scripts from Moses to compute the BLEU score. For chrF3, we use the implementation from github: https://github.com/rsennrich/subword-nmt .

	Vocabulary	Parameters	Layers	Epochs	BLEU	GPU Days
RNNsearch (Bahdanau, Cho, and Bengio 2015)	30K words	85 M	(1, 1)	5	28.5	10
AvA NMT (shallow model)	300 chars	30 M	(1, 1)	2.2	**33.2**	10
GNMT (Wu et al. 2016)	32K WPM	$>$ 100 M	(8, 8)	-	39.0	576
ConvS2S (Gehring et al. 2017)	40K BPE	$>$ 100 M	-	-	**40.5**	296
AvA NMT (deep model)	300 chars	120 M	(4, 2)	4.5	40.2	120

Table 2: Comparison with RNNsearch and GNMT on the En-Fr translation task.

	Vocabulary	Parameters	Layers	Epochs	BLEU	chrF_3	GPU Days
bpe2char (Chung, Cho, and Bengio 2016)	word, char	76 M	(1, 2)	-	17.0	-	-
hybrid (Luong and Manning 2016)		250 M	(4, 4)	6	19.6	46.5	25
character (Luong and Manning 2016)	char	100 M	(4, 4)	6	17.5	46.6	90
char2char (Lee, Cho, and Hofmann 2016)		69 M	(1, 2)	4.8	17.6	46.8	14
AvA NMT (shallow model)		66 M	(2, 1)	3.6	**19.8**	**48.3**	14
AvA NMT (deep model)		120 M	(4, 2)	5.6	**20.9**	**49.2**	30

Table 3: Comparison with various models on the En-Cs translation task.

though our deep AvA model is much shallower than the GNMT model. Although the complexity of our character-level model is higher than the word-level model (see, Table 1) , the convergence of training is much faster.

En-Cs task. There are many works for dealing with the morphologically rich languages, such as bpe2char models (Chung, Cho, and Bengio 2016), char2char models (Lee, Cho, and Hofmann 2016) and hybrid models (Luong and Manning 2016). We compare the performance of these systems[3] in Table 3. We could find that our shallow AvA NMT model outperforms all these models in terms of the BLEU score and chrF_3. Moreover, our shallow AvA NMT model achieves a competitive BLEU score and a substantial improvement on chrF_3 score comparing to the state-of-the-art hybrid model. However, the size of our shallow model is about a quarter of the hybrid model and the training time is halved. It is probably because our model builds a shortcut mapping between characters through the attention-via-attention mechanism instead of building a mapping between words directly. We will analyze this property in the following sections. Besides, it might be unfair to compare our shallow model to the deep hybrid model, thus we trained a deep AvA NMT model. As shown in Table 3, the deep AvA NMT model further results a substantial improvement both on BLEU and chrF_3.

Language Similarity

We would like to verify whether our model could detect the named entities and similar words quantitatively. Specifically, we hypothesize that the characters in these words would be aligned by our attention-via-attention mechanism (see Section 6 for a graphical illustration). Thus, we may estimate the similarity between the languages. We simply regard the words to be similar when more than three characters in the source words are attended. To eliminate the noise, we only the focus characters that account for more than half of the weight. In this way, we find 22% of the words in French are similar to the corresponding words in English based on *newstest2013*. However, the similarity between English and Czech is decreased to 13%. It shows a good accordance with the reality, which also matches the statistics in (Simons and Fennig 2017).

[3]The results are taken from the corresponding paper, except the result of char2char model which is evaluated by us based on their codes.

6 Qualitative Analysis

Apart from measuring translation quality, we analyze effects of the attention-via-attention mechanism in more details.

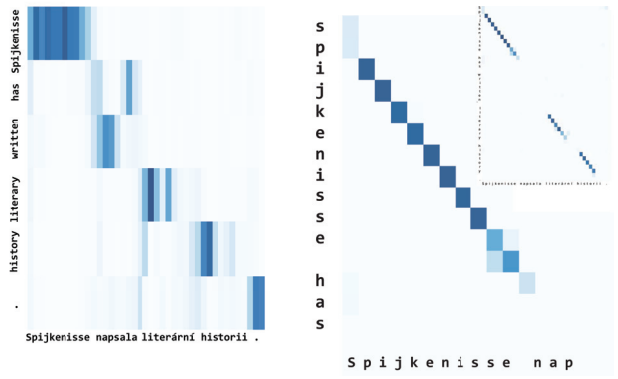
Alignments

In this section, we investigate whether our model could utilize the attention-via-attention mechanism graphically. We select a representative English sentence "`Spijkenisse has written literary history.`" from *newstest2013*, which would be translated into a Czech sentence "`Spijkenisse napsala literární historii.`".

We could find in Figure 2 that the alignment of '`Spijkenisse`' is well captured both on the word level and the character level. Besides, the source words '`literary history`' and the similar target words '`literární historii`' are also aligned respectively as shown in the thumbnail in Figure 2(b). However, the source word '`written`', significantly different from the target word '`napsala`', is attended only on the word level. Thus, we claim that our model bridges the lexicons of languages through attention-via-attention mechanism. We provide the

En src	The analogy with the **electromagnetic** field is again useful for explaining the relationship between the ***Higgs*** and mass.
Fr ref	L'analogie avec le champ **électromagnétique** est de nouveau utile pour expliquer le rapport entre le ***Higgs*** et la masse.
Fr gen	L'analogie avec le champ **électromagnétique** est une fois encore utile pour expliquer la relation entre les ***Higgs*** et la masse.
Cs ref	Analogie s **elektromagnetickým** polem se nám znovu hodí k objasnění vztahu mezi ***Higgsem*** a hmotou.
Cs gen	Analogie s **elektromagnetickým** polem je znovu užitečná pro vysvětlení vztahu mezi ***Higgs*** a hmotností.
En src	You can download the **document** (in English for the time being, a [French] translation will be available shortly) at this **address**: ***http://ca.movember.com/fr/mens-health/prostate-cancer-screening***
Fr ref	On peut télécharger ce **document** (en anglais pour l'instant, une traduction sera offerte sous peu) à cette **adresse**: ***http://ca.movember.com/fr/mens-health/prostate-cancer-screening***
Fr gen	Vous pouvez télécharger le **document** (en anglais pour le moment, une traduction française sera disponible prochainement) à l'**adresse**: ***http://ca.movember.com/fr/mens-health/prostate-cancer-screening***
Cs ref	Tento **dokument** si můžete stáhnout (momentálně v angličtině, překlad bude k dispozici později) na této **adrese**: ***http://ca.movember.com/fr/mens-health/prostate-cancer-screening***
Cs gen	**Dokument** můžete stáhnout (v angličtině pro čas , překlad [Francie] bude k dispozici krátce na této **adrese**: ***http://ca.movember.com/fr/mens-health/prostate-cancer-screening***

Table 4: Sample translations of *newstest2013*.



(a) Word-level attention (b) Character-level attention

Figure 2: Sample alignments found by the attention-via-attention mechanism. The translated Czech characters are listed below the figure. The character-level attention is zoomed in for clarity, and the thumbnail in the top right corner contains the full alignments of the character level.

more detailed figures in the supplemental material.

Sample Translations

To further demonstrate the characteristic of the attention-via-attention mechanism, we provide several sample translations generated by our models. The cognates and the named entities are marked as bold and bold italic type respectively in Table 4. For instance, the cognates '`electromagnetic`', '`électromagnétique`' and '`elektromagnetickým`' are well handled by our model. The named entity '`Higgs`' are copied to target languages. Although our models have never seen the url during training, the long URL is copied to target languages without any mistakes.

From above examples, we could see that the attention-via-attention mechanism not only copy words (Gulcehre et al. 2016; He et al. 2017), but also adaptively modify the words based on the target language. Thus, we have alleviated the burden of building the mapping between words by bridging the characters in NMT.

7 Conclusion

We have developed a hierarchical encoder and an attention-via-attention mechanism to bridge the lexicons of the languages in neural machine translation. Consequently, our AvA NMT model is able to deal with the cognates and named entities more elegantly. We have achieved the competitive performance with the proposed models whose size is much smaller. The promising empirical results strongly suggest that it is indeed beneficial for neural machine translation to exploit character-level information. More interestingly, English is roughly evaluated to have a similarity of 22% with French and 13% with Czech based on the alignments found by our attention-via-attention mechanism.

Furthermore, the attention-via-attention mechanism could be regarded as adding attentive shortcuts between layers, thus we would like to incorporate this mechanism into very deep word-level RNN models (Britz et al. 2017) to replace the skip connections in future work.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (No. 61572017 and 11771002).

References

Bahdanau, D.; Cho, K.; and Bengio, Y. 2015. Neural machine translation by jointly learning to align and translate. *International Conference on Learning Representation*.

Britz, D.; Goldie, A.; Luong, M.; and Le, Q. V. 2017. Massive exploration of neural machine translation architectures. *arXiv preprint arXiv:1703.03906*.

Chung, J.; Ahn, S.; and Bengio, Y. 2017. Hierarchical multiscale recurrent neural networks. *International Conference on Learning Representation*.

Chung, J.; Gulcehre, C.; Cho, K.; and Bengio, Y. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*.

Chung, J.; Cho, K.; and Bengio, Y. 2016. A character-level decoder without explicit segmentation for neural machine translation. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*.

Cui, Y.; Chen, Z.; Wei, S.; Wang, S.; Liu, T.; and Hu, G. 2016. Attention-over-attention neural networks for reading comprehension. *arXiv preprint arXiv:1607.04423*.

Gehring, J.; Auli, M.; Grangier, D.; Yarats, D.; and Dauphin, Y. N. 2017. Convolutional sequence to sequence learning. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, 1243–1252.

Gulcehre, C.; Ahn, S.; Nallapati, R.; Zhou, B.; and Bengio, Y. 2016. Pointing the unknown words. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*.

He, S.; Liu, C.; Liu, K.; and Zhao, J. 2017. Generating natural answers by incorporating copying and retrieving mechanisms in sequence-to-sequence learning. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, 199–208.

Hochreiter, S., and Schmidhuber, J. 1997. Long short-term memory. *Neural computation* 9(8):1735–1780.

Jean, S.; Cho, K.; Memisevic, R.; and Bengio, Y. 2015. On using very large target vocabulary for neural machine translation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics*, 1–10.

Kim, Y.; Jernite, Y.; Sontag, D.; and Rush, A. M. 2016. Character-aware neural language models. *Association for the Advancement of Artificial Intelligence, AAAI 2016*.

Kingma, D., and Ba, J. 2015. Adam: A method for stochastic optimization. *International Conference on Learning Representation*.

Lee, J.; Cho, K.; and Hofmann, T. 2016. Fully character-level neural machine translation without explicit segmentation. *arXiv preprint arXiv:1610.03017*.

Luong, M.-T., and Manning, C. D. 2016. Achieving open vocabulary neural machine translation with hybrid word-character models. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*.

Luong, M.-T.; Pham, H.; and Manning, C. D. 2015. Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*.

Papineni, K.; Roukos, S.; Ward, T.; and Zhu, W.-J. 2002. BLEU: a method for automatic evaluation of machine translation. *Association for Computational Linguistics* 311–318.

Popovic, M. 2015. chrF: character n-gram f-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation, WMT@EMNLP 2015*, 392–395.

Schuster, M., and Paliwal, K. K. 1997. Bidirectional recurrent neural networks. *Signal Processing, IEEE Transactions on* 45(11):2673–2681.

Sennrich, R.; Haddow, B.; and Birch, A. 2016. Neural machine translation of rare words with subword units. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*.

Simons, G. F., and Fennig, C. D. 2017. *Ethnologue: Languages of the world, twentieth edition*. SIL international Dallas, Texas.

Sutskever, I.; Vinyals, O.; and Le, Q. V. 2014. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems*, 3104–3112.

Vinyals, O., and Le, Q. V. 2015. A neural conversational model. *arXiv preprint arXiv:1506.05869*.

Wu, Y.; Schuster, M.; Chen, Z.; Le, Q. V.; Norouzi, M.; Macherey, W.; Krikun, M.; Cao, Y.; Gao, Q.; Macherey, K.; et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

Yang, Z.; Yang, D.; Dyer, C.; He, X.; Smola, A. J.; and Hovy, E. H. 2016. Hierarchical attention networks for document classification. In *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 1480–1489.