University of Rochester WMT 2017 NMT System Submission

Chester Holtz, Chuyang Ke, and **Daniel Gildea**
University of Rochester
choltz2@u.rochester.edu

Abstract

We describe the neural machine translation system submitted by the University of Rochester to the Chinese-English language pair for the WMT 2017 news translation task. We applied unsupervised word and subword segmentation techniques and deep learning in order to address (i) the word segmentation problem caused by the lack of delimiters between words and phrases in Chinese and (ii) the morphological and syntactic differences between Chinese and English. We integrated promising recent developments in NMT, including back-translations, language model reranking, subword splitting and minimum risk tuning.

1 Introduction

This paper presents the machine translation (MT) systems submitted by University of Rochester to the WMT 2017 news translation task. We participated in the Chinese-to-English and Latvian-to-English news translation tasks, but will focus on describing the system submitted for the Chinese-to-English task.

Chinese-to-English is a particularly challenging language pair for corpus-based MT systems due to the task of finding an optimal word segmentation for Chinese sentences as well as other linguistic differences between Chinese and English sentences. For example the fact that there may exist multiple possible meanings for characters depending on their context and that individual characters can be joined together to build compound words exacerbate the aforementioned segmentation problem. Additionally, translation performance is also affected by the frequent dropping of subjects and infrequent use of function words in Chinese sentences.

We used both word-level and morphological feature-based representations of Chinese to deal with data sparsity and reduce the size of the Chinese vocabulary. We experimented with both subphrase-based and character-based systems. Both RNN-based and 5-gram language models were trained with data extracted from the English news corpora provided and are used to rerank hypotheses proposed by the decoder.

The paper is organized as follows: in Section 2 we introduce our system and preprocessing methods for the Chinese language. Our main learning framework training settings are explained in Section 3. Our NMT, SMT, and submission results are presented in Section 4. The paper ends with some concluding remarks.

2 System Description

In this section we briefly introduce our preprocessing methods and the general encoder-decoder framework with attention (Sutskever et al., 2014; Cho et al., 2014; Bahdanau et al., 2014) used in our system. We closely followed the neural machine translation model proposed by Chorowski et al. (2015).

A neural machine translation model (Kalchbrenner and Blunsom, 2013; Cho et al., 2014; Sutskever et al., 2014) aims at building an end-to-end neural network framework, which takes as input a source sentence $X = (\mathbf{x}_1, ..., \mathbf{x}_{T_X})$ with length of T_X, and outputs its translation $Y = (\mathbf{y}_1, ..., \mathbf{y}_{T_Y})$ with length of T_Y, where $\mathbf{x}_t$ and $\mathbf{y}_t$ are the source and target language tokens, respectively. The framework is constructed as a composite of an encoder network and a decoder network.

Proceedings of the Conference on Machine Translation (WMT), Volume 2: Shared Task Papers, pages 310–314
Copenhagen, Denmark, September 711, 2017.

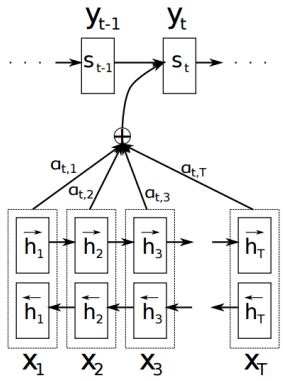

Figure 1: Illustration of the encoder-decoder framework from Bahdanau et al. (2014).

2.1 Morphological Analyzer

Word segmentation is considered an important first step for Chinese natural language processing tasks since individual Chinese words can be composed of multiple characters with no space appearing between words.

We employed the Jieba morphological analyzer (Junyi, 2013) to segment the source Chinese sentences into words. Jieba decomposes Chinese sentences into sequences of words by constructing a graph for all possible word combinations and finds the most probable sequence based on statistics derived from training data. For unknown words, an HMM-based model is used with the Viterbi algorithm.

2.2 Rare-Morpheme (BPE) Algorithm

If we simply apply the Chinese morphological analyzer to segment Chinese sentences into individual words and feed the words into our encoder, overfitting will occur; some words are so rare, that they only appear altogether with others. Thus, we enforced a thresholded on frequent words and applied the byte-pair-encoding (BPE) algorithm proposed by Gage (1994) and applied by Sennrich et al. (2016b) to NMT to further reduce the sparsity of our language data and to reduce the number of rare and out-of-vocabulary tokens.

2.3 Encoder

The encoder reads a sequence of source language tokens $X = (\mathbf{x}_1, \ldots, \mathbf{x}_{T_X})$, and outputs a sequence of hidden states $H = (h_1, \ldots, h_{T_X})$. A bidirectional recurrent neural network (BiRNN) (Bahdanau et al., 2014) consisting of a forward recurrent neural network (RNN) and a backward RNN, is used to give additional positional representational power to the encoder. The lower part of Figure 1 illustrates the BiRNN structure.

The forward network reads the input sentence in a forward direction

$$\overrightarrow{h_t} = \overrightarrow{\phi_x}(i_x(\mathbf{x}_t), \overrightarrow{h_{t-1}}) \quad (1)$$

where for each input token $\mathbf{x}_t$, $i_x(\cdot) : X \to \mathbb{R}^n$ is a continuous embedding, that maps the t-th input token to a vector $i_x(\mathbf{x}_t)$ in a high dimensional space $\mathbb{R}^n$. A forward recurrent activation function $\overrightarrow{\phi_x}$ updates each forward hidden state $\overrightarrow{h_t}$, using the embedded token $i_x(\mathbf{x}_t)$ and the information of the previous hidden state $\overrightarrow{h_{t-1}}$.

Similarly, the reverse network reads the sentence in a reverse direction (right to left)

$$\overleftarrow{h_t} = \overleftarrow{\phi_x}(i_x(\mathbf{x}_t), \overleftarrow{h_{t+1}}) \quad (2)$$

and generates a sequence of backward hidden states.

The encoder utilizes information from both the forward RNN and the backward RNN to generate the hidden states $H = (h_1, \ldots, h_{T_X})$. For every input token $\mathbf{x}_t$, we concatenate its corresponding forward hidden state vector and the backward hidden state vector, such that $h_t = \begin{bmatrix} \overrightarrow{h_t} \\ \overleftarrow{h_t} \end{bmatrix}$.

2.4 Decoder

The upper part of Figure 1 illustrates the decoder. The decoder computes the conditional distribution over all possible translations based on the context information provided by the encoder (Bahdanau et al., 2014). More specifically, the decoder RNN tries to find a sequence of tokens in the target language that maximizes the following probability:

$$\log p(Y|X) = \sum_{t=1}^{T_Y} \log p(\mathbf{y}_t|\mathbf{y}_1, \ldots, \mathbf{y}_{t-1}, X) \quad (3)$$

Each hidden state s_t in the decoder is updated by

$$s_t = \phi_y(i_y(\mathbf{y}_{t-1}), s_{t-1}, c_t), \quad (4)$$

where i_y is the continuous embedding of a token in the target language. c_t is a context vector related to the t-th output token, such that

$$c_t = \sum_{l=1}^{T_X} h_l \cdot a_{tl} \quad (5)$$

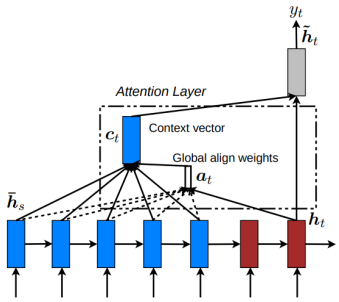

Figure 2: Illustration of Attention Mechanism from Luong et al. (2015).

and

$$a_{tl} = \frac{\exp(e_{tl})}{\sum_{k=1}^{T_X} \exp(e_{tk})} \quad (6)$$

Here, a_{tl} indicates the importance of the hidden state annotation h_l regarding to the previous hidden state s_{t-1} in the decoder RNN. e_{tk} measures how "matching" the input at position k and the output at position t are (Bahdanau et al., 2014; Chorowski et al., 2015); it is defined by a soft alignment model f_{align}, such that

$$e_{tk} = f_{\text{align}}(s_{t-1}, h_k). \quad (7)$$

Finally, each conditional probability in Equation 3 is generated by

$$p(\mathbf{y}_t|\mathbf{y}_1, \ldots, \mathbf{y}_{t-1}, X) = g(\mathbf{y}_{t-1}, s_t, c_t) \quad (8)$$

for some nonlinear function g.

2.5 Attention Mechanism

The soft-alignment mechanism f_{align} weighs each vector in the context set $C = (c_1, \ldots, c_{T_Y})$ according to its relevance given what has been translated (Bahdanau et al., 2014; Cho et al., 2014; Sutskever et al., 2014). It is commonly implemented as a feedforward neural network with a single hidden layer. This procedure can be understood as computing the alignment probability between the t-th target symbol and k-th source symbol.

The hidden state annotation $\mathbf{h}_t$, together with the previous target symbol $\mathbf{y}_{t-1}$ and the context vector c_t, is fed into a feedforward neural network to result in the conditional distribution and the whole network, consisting of the encoder, decoder and soft-alignment mechanism, is then tuned end-to-end to minimize the negative log-likelihood using stochastic gradient descent. In our system, the source sentence X is a sequence of sub-phrase and sub-word tokens extracted by the morphological analyzer and BPE algorithms, and the target sentence Y is represented as a sequence of sub-words.

2.6 Minimum Risk Tuning

We applied minimum risk training (Shen et al., 2016) to tune the model parameters post convergence of the cross-entropy loss by minimizing the expected risk for sentence-level BLEU scores where the risk is defined to be

$$R(\boldsymbol{\theta}) = \sum_{s=1}^{S} \mathbb{E}_{\mathbf{y}|\mathbf{x}^{(s)};\boldsymbol{\theta}}[\Delta(\mathbf{y}, \mathbf{y}^{(s)})] \quad (9)$$

$$= \sum_{s=1}^{S} \sum_{\mathbf{y} \in Y(\mathbf{x}^{(s)})} P(\mathbf{y}|\mathbf{x}^{(s)}; \boldsymbol{\theta}) \Delta(\mathbf{y}, \mathbf{y}^{(s)}) \quad (10)$$

for candidate translations $Y(\mathbf{x}^{(s)})$ for $\mathbf{x}^{(s)}$. Details regarding methods to solve this problem can be found in Shen et al. (2016).

3 Experimental Settings

In this section, we describe the details of the experimental settings for our system.

3.1 Corpora and Preprocessing

Our model was trained on all available training parallel corpora for the ZH-EN language pair. The training data consists of approximately $2,000,000$ sentence pairs. We removed sentence pairs from our data when the source or target side is more than 50 tokens long. A set of $50,000,000$ sentences was sampled from the News Crawl 2007-15 data and was used to train our target side (English) language model. Additionally, we backtranslated a subset of these sentences and used the resulting source-target sentences to augment our training data.

Our training and development data were lowercased and preprocessed using the Moses tokenizer script (Koehn et al., 2007), Jieba, and BPE. We set the upper bound on the target vocabulary to $30,000$ sub-words and two additional tokens reserved for $\langle EOS \rangle$ and $\langle UNK \rangle$. For the source vocabulary, we constrained the size of BPE symbol vocabulary to $30,000$ tokens.

3.2 Synthetic Training Data

Sennrich et al. (2016a) introduced the augmentation of a parallel corpus by leveraging target-side monolingual data and empirically showed

that treating back-translations as additional training data reduced overfitting and increased fluency of the translation model. We sampled monolingual sentences from the same news data used to construct our language models. Due to computation and time constraints, we were only able to augment our training data by an additional 190,000 sentence pairs. We hypothesize that increasing the number of back-translated sentences in our training set will further improve our system's performance.

3.3 Neural Baseline

Our NMT baseline is an encoder-decoder model with attention and dropout implemented with Nematus (Sennrich et al., 2017) and AmuNMT (Junczys-Dowmunt et al., 2016). This baseline system without pre-tokenization or language model scoring achieves 17.32 uncased BLEU on news-test2017 and 19.78 after source-segmentation with the BPE algorithm.

We used beam search with a beam width of 8 to approximately find the most likely translations given a source sentence before introducing features proposed by our language models and reranking with the default Moses (Koehn et al., 2007) implementation of K-best MIRA (Cherry and Foster, 2012). Both language models were trained on the English news data. Our unigram-pruned 5-gram language model was trained with KenLM (Heafield, 2011), and our RNN-based language model was trained with RNNLM (Mikolov et al., 2011) with a hidden layer size of 300.

3.4 Statistical Baseline

For our SMT baseline, we trained a standard phrase-based system on input segmented with Jieba: Berkeley Aligner (IBM Model 1 and HMM, both for 5 iterations); phrase table with up to 5 tokens per phrase, 40-best translation options per source phrase, and Good-Turing smoothing; 4-gram language model and pruning of singleton n-grams; and the default K-best MIRA reordering.

This baseline system achieves an uncased BLEU score of 7.46 on news-test2017.

4 Experimental Results

We compared the performance of our system to several state-of-the-art algorithms. Our systems (Character-level BiRNN, Morphological Subword BiRNN) are marked in a bold font. It can be seen that our system outperformed the baselines, whether using words or subwords as the input tokens. The experiments also showed that the rare-morpheme algorithm significantly reduced some potential overfitting, compared to the character-level BiRNN.

System	Score
Moses Baseline (word)	7.5
Neural Baseline (word)	17.3
Neural Baseline (subword)	19.8
BiRNN (character)	12.5
BiRNN (word + subword)	21.6

Table 1: Test Results. Uncased BLEU scores of the trained models computed over all sentences on the development and test sets.

4.1 Error Analysis

Error analysis on the validation set shows that the two main sources of errors produced by the baseline are missing and incorrect words. These issues are addressed in our model by applying morphological segmentation in combination with BPE and adding new backtranslated data to the training set. Our model's translation error rate (0.716) is strictly lower than that of our baseline's output (0.743). We attribute this reduction in error rate to our system being able to more robustly model multi-character words in Chinese.

5 Conclusion

We describe the University of Rochester neural machine translation system for WMT'17 Chinese-English news translation task, which employs recent developments in the machine translation field. Our results show that applying word and morpheme-aware tokenization, minimum risk tuning, and language model reranking to an existing MT framework help to improve the overall translation quality of the model.

Machine translation is a dynamic area, and there are many opportunities for further exploration.

- **Other objectives**: Modify the encoder-decoder trainer and add secondary tasks for multi-task training (e.g. source sentence tagging) for explicit use of linguistic features.
- **Sentence reordering**: Reorder the training data in various ways to encourage the model to learn a more robust translation model.

- **Source-side monolingual data**: Leverage source-side monolingual data to improve translation performance.

Acknowledgments

The authors would like to thank the developers of Nematus (Sennrich et al., 2017) and Amunmt (Junczys-Dowmunt et al., 2016) as well as Theano (Al-Rfou et al., 2016). We acknowledge the support of the University of Rochester and the Center for Integrated Research Computing at the University of Rochester for computing support. Finally, we are grateful to the University of Edinburgh for centralizing the Chinese-English parallel corpora.

References

Rami Al-Rfou, Guillaume Alain, Amjad Almahairi, Christof Angermueller, Dzmitry Bahdanau, Nicolas Ballas, et al. 2016. Theano: A Python framework for fast computation of mathematical expressions. *arXiv:1409.1259*.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv:1409.0473*.

Colin Cherry and George Foster. 2012. Batch tuning strategies for statistical machine translation. In *Proceedings of the 2012 Conference the North American Association for Computational Linguistics: Human Language Technologies (HLT-NAACL 2012)*.

Kyunghyun Cho, Bart Van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv:1409.1259*.

Jan Chorowski, Dzmitry Bahdanau, Dmitriy Serdyuk, Kyunghyun Cho, and Yoshua Bengio. 2015. Attention-based models for speech recognition. In *Advances in Neural Information Processing Systems (NIPS 2015)*, pages 577–585.

Philip Gage. 1994. A new algorithm for data compression. *The C Users Journal*, 12(2):23–38.

Kenneth Heafield. 2011. Kenlm: Faster and smaller language model queries. In *Proceedings of the 2011 Workshop on Statistical Machine Translation (WMT11)*, pages 187–197.

Marcin Junczys-Dowmunt, Tomasz Dwojak, and Hieu Hoang. 2016. Is neural machine translation ready for deployment? a case study on 30 translation directions. In *Proceedings of the 9th International Workshop on Spoken Language Translation (IWSLT 2016)*.

Sun Junyi. 2013. Jieba. http://github.com/fxsjy/jieba.

Nal Kalchbrenner and Phil Blunsom. 2013. Recurrent continuous translation models. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP 2013)*, pages 1700–1709.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. *Association for Computational Linguistics (ACL)*.

Minh-Thang Luong, Hieu Pham, and Christopher Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP 2013)*.

Tomas Mikolov, Stefan Kombrink, Anoop Deoras, Lukas Burget, and Jan "Honza" Cernock. 2011. Rnnlm - recurrent neural network language modeling toolkit. In *Proceedings of Interspeech*.

Rico Sennrich, Orhan Firat, Kyunghyun Cho, Alexandra Birch, Barry Haddow, Julian Hitschler, Marcin Junczys-Dowmunt, Samuel Läubli, Antonio Valerio Miceli Barone, Jozef Mokry, and Maria Nadejde. 2017. Nematus: a toolkit for neural machine translation. In *Proceedings of the 15th Conference of the Association for Computational Linguistics (ACL 2017)*, pages 65–68.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016a. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL 2016)*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016b. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL 2016)*.

Shiqi Shen, Yong Cheng, Zhongjun He, Wei He, Hua Wu, Maosong Sun, and Yang Liu. 2016. Minimum risk training for neural machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL 2016)*.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems (NIPS 2014)*, pages 3104–3112.