Three Strategies to Improve One-to-Many Multilingual Translation

Yining Wang[2,3] **, Jiajun Zhang**[1,2,3]*, **Feifei Zhai**[5], **Jingfang Xu**[5] and **Chengqing Zong**[2,3,4]
[1]Beijing Advanced Innovation Center for Language Resources, Beijing, China
[2]National Laboratory of Pattern Recognition, CASIA, Beijing, China
[3]University of Chinese Academy of Sciences, Beijing, China
[4]CAS Center for Excellence in Brain Science and Intelligence Technology, Beijing, China
[5]Sogou Inc., Beijing, China
{yining.wang, jjzhang, cqzong}@nlpr.ia.ac.cn
{zhaifeifei,xujingfang}@sogou-inc.com

Abstract

Due to the benefits of model compactness, multilingual translation (including many-to-one, many-to-many and one-to-many) based on a universal encoder-decoder architecture attracts more and more attention. However, previous studies show that one-to-many translation based on this framework cannot perform on par with the individually trained models. In this work, we introduce three strategies to improve one-to-many multilingual translation by balancing the shared and unique features. Within the architecture of one decoder for all target languages, we first exploit the use of unique initial states for different target languages. Then, we employ language-dependent positional embeddings. Finally and especially, we propose to divide the hidden cells of the decoder into shared and language-dependent ones. The extensive experiments demonstrate that our proposed methods can obtain remarkable improvements over the strong baselines. Moreover, our strategies can achieve comparable or even better performance than the individually trained translation models.

1 Introduction

Encoder-decoder based neural machine translation (NMT) has achieved the new state-of-the-art due to powerful end-to-end modeling (Sutskever et al., 2014; Bahdanau et al., 2015; Wu et al., 2016; Hassan et al., 2018). Under this end-to-end framework, many researchers attempt to improve the translation quality between two languages by exploiting monolingual data (Sennrich et al., 2016; Zhang and Zong, 2016), taking advantage of both NMT and statistical machine translation (Wang et al., 2017a; Tang et al., 2016; Zhao et al., 2018; Zhou et al., 2017) and so on.

Another research direction about how to perform multilingual translation within this encoder-decoder architecture has recently drawn more and more attention (Zoph and Knight, 2016; Dong et al., 2015; Luong et al., 2016; Johnson et al., 2017; Firat et al., 2016b).

In multilingual translation scenarios, one can employ multi-task learning framework to perform many-to-one or one-to-many translation using multiple encoders or multiple decoders (Luong et al., 2016; Dong et al., 2015). Firat et al. (2016a) and Lu et al. (2018) further propose to share a universal attention mechanism for many-to-many translations. In these methods, encoder or decoder is language dependent and network parameters increase linearly with the number of languages.

Johnson et al. (2017) and Ha et al. (2016) present an appealing approach in which a universal encoder-decoder framework is designed for many-to-one, many-to-many and one-to-many multilingual translation tasks. The network model is compact and the model size does not grow as the number of languages increases. However, Johnson et al. (2017) observe that only the many-to-one paradigm can achieve better translation results than the individually trained models. For the other two paradigms, there are various degrees of quality degradation. In this work, we focus on one-to-many multilingual translation under the universal encoder-decoder framework and attempt to boost its performance while maintaining the model compactness.

To this end, we propose three strategies which exploit the unique features of each target language and keep as many parameters shared as possible. First, we design two special labels at the tail of encoder and the head of decoder to mark the target language and guide the generation of different target languages. Then, we introduce language-dependent positional embeddings into the bottom

*Jiajun Zhang is the corresponding author and the work is done while Yining Wang is doing research intern at Sogou Inc.

Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 2955–2960
Brussels, Belgium, October 31 - November 4, 2018.

layer of the decoder network and correspondingly the structural difference between target languages can be well captured. Finally and especially, we propose a new parameter-sharing mechanism in which we divide the hidden units of each decoder layer into shared and language-dependent ones.

We verify the effectiveness of our proposed methods on two one-to-many tasks: Chinese-to-English/Japanese translation and English-to-German/French translation. The experimental results demonstrate that the three strategies can significantly outperform the baseline multilingual models and they can achieve comparable or even better performance than the individually trained translation models.

Specifically, our contributions in this paper are two-fold:

- The proposed three strategies can take advantage of unique features of each target language while sharing the network parameters as many as possible.
- The extensive experiments on multiple translation tasks show that the three proposed strategies improve the translation quality. Moreover, the effects of the strategies are complementary and the combined one can perform on par with or better than the individually optimized translation models.

2 Background

Our proposed approach can be applied to any encoder-decoder architecture. Considering the excellent translation performance of Transformer network (Vaswani et al., 2017), we implement our method entirely based on it in this work. Transformer consists of stacked encoder and decoder layers. The encoder maps an input sequence $\mathbf{x} = (x_1, x_2, \cdots, x_n)$ to a sequence of continuous representations $\mathbf{z} = (z_1, z_2, \cdots, z_n)$ whose size varies with respect to the source sentence length. The decoder generates an output sequence $\mathbf{y} = (y_1, y_2, \cdots, y_m)$ from the continuous representations $\mathbf{z}$. Since the Transformer network contains no recurrence, positional embeddings are used in model to make use of sequence order. The encoder and decoder are trained to maximize the conditional probability of target sequence given a source sequence:

$$L(\theta) = \sum_{t=1}^{N} \log P(y_t | y_{<t}, x; \theta) \qquad (1)$$

For the sake of brevity, we refer the reader to Vaswani et al. (2017) for more details regarding the architecture.

3 Method Description

In this section, we introduce our general strategies for extending the transformer network to one-to-many translation task. We decompose the probability of the target sequences into the products of per token probabilities in all translation forms:

$$L(\theta) = \sum_{t=1}^{M} \sum_{l=1}^{N_l} \log(P(y_t^l | x, y_{<t}^l; \theta)) \qquad (2)$$

where M is number of target languages, and $P(y_t^l | x, y_{<t}^l; \theta)$ denotes the translation probability of t-th word of the l-th target language. Note that the translation process for all target languages uses the same parameter set θ.

Our methods mainly concentrate on improving one-to-many multilingual translation by designing new decoder structure under the universal encoder-decoder framework. The idea is to exploit the shared and unique features of different target languages, and we respectively propose three strategies including special label initialization, language-dependent positional embedding and a new parameter-sharing mechanism.

3.1 Special Label Initialization

In the universal encoder-decoder network for one-to-many multilingual translation (Johnson et al., 2017), a special token (e.g. **en2fr**) is added at the end of the source sentence to indicate the translation direction. Although it is an effective mechanism, we find that the initial states of the decoder are very important to guide the generation process for different target languages. In order to enhance the model, we utilize another special language-dependent label at the beginning of the decoder and we regard it as the first generated token of the target language (e.g. **2fr**).

3.2 Language-dependent Positional Embedding

Positional embeddings give the model the sense of which part of the sequence is currently being dealt with. Intuitively, different target languages should have different positional embeddings to distinguish the structural difference between multiple target languages. Therefore, we design language-dependent positional embeddings in the universal

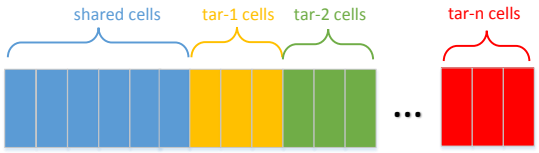

Figure 1: The hidden units of decoder network. Blue part represents the shared units, and yellow, green and red parts denote different language-dependent units respectively.

encoder-decoder multilingual translation. For the fixed embedding method (Vaswani et al., 2017), $sine(x)$ and $cosine(x)$ functions are used to generate positional embeddings. In this case, we introduce trigonometric functions with different orders or offsets on the decoder to distinguish different target languages. For the dynamic embedding method (Gehring et al., 2017), we equip the target inputs by embedding the absolute position of different languages separately.

3.3 Shared and Language-dependent Hidden Units per Layer

In the universal encoder-decoder multilingual translation, the hidden layers of the decoder are responsible for generating different target language sentences. As a result, the hidden layers should embody some language-dependent information.

In this work, we propose to divide the hidden units of each decoder layer into shared units and language-dependent ones. On the one hand, shared units can learn the commonality of languages and enable one-to-many translation to share the network parameters as many as possible. On the other hand, language-dependent units are capable of capturing the characteristic of each specific language.

Figure 1 gives a brief description of our proposed strategy. For instance, in training step for one target language (**tar-1**), we tune the shared units and the language-dependent units of **tar-1**, and mask out other parts. In decoding step, we only use the shared and language-dependent hidden units of target language **tar-1** to predict translation results.

4 Experiments Settings

In this section, we test the proposed methods on two one-to-many translation tasks, including (i) Chinese→English/Japanese in general domain, and (ii) English→French/German in WMT14 task.

Chinese→English/Japanese For this translation task, the training sets of Chinese-to-English (briefly, Zh→En) and Chinese-to-Japanese (briefly, Zh→Ja) both contain about 10 million parallel corpora. We evaluate our methods on NIST03-06 (MT03-06) for Zh→En translation and 400 sentences extracted from our general corpus for Zh→Ja translation.

English→French/German The training set consists of about 4.5 million bilingual sentence pairs in WMT14 English-German (briefly, En→De) task and about 36 million sentence pairs in WMT14 English-French (briefly, En→Fr) task[1]. We use the combination of *newstest2012* and *newstest2013* as our validation set, and we use *newstest2014* as our test set on En→De and En→Fr tasks.

We adopt the tensor2tensor[2] library for training and evaluating our basic Transformer translation model. We use wordpiece method (Wu et al., 2016; Schuster and Nakajima, 2012) to encode source side sentences and the combination of target side sentences. The vocabulary size is 37,000 for both sides. We train our models using configuration *transformer_big* adopted by Vaswani et al. (2017), which contains a 6-layer encoder and a 6-layer decoder with 1024-dimensional hidden representations. During training, each mini-batch on one GPU contains a set of sentence pairs with roughly 3,072 source and 3,072 target tokens. We use Adam optimizer (Kingma and Ba, 2014) with β_1=0.9, β_2=0.98, and ϵ=10^{-9}. For our model, we train for 400,000 steps on one machine with 8 NVIDIA Tesla M40 GPUs.

5 Results and Analysis

We show the results of one-to-many translation experiments using our proposed strategies. The translation performance is evaluated by case-insensitive BLEU4 for Zh→En translation, character-level BLEU5 for Zh→Ja translation, and case-sensitive BLEU4 (Papineni et al., 2002) for En→De/Fr translation task.

5.1 Our Strategies vs. Baseline

Table 1 reports the main translation results of Zh→En/Ja and En→De/Fr translation tasks. We conduct universal one-to-many translation using

[1]http://www.statmt.org/wmt14/translation-task.html
[2]https://github.com/tensorflow/tensor2tensor

Methods	Zh→En					Zh→Ja	En→De	En→Fr
	MT03	MT04	MT05	MT06	**Ave**	**test**	**test**	**test**
Indiv	43.59	43.95	45.34	44.05	44.23	40.71	**27.84**	41.50
O2M	43.20	43.55	44.68	43.93	43.84	42.09	26.42	41.32
O2M + ①	43.91	44.01	45.12	44.14	44.30	42.54	26.78	41.56
O2M + ① + ② (Dyn)	44.24	44.45	45.43	44.51	44.66	42.77	26.98	41.78
O2M + ① + ② (Fixed)	44.13	44.57	45.22	44.68	44.65	42.70	26.90	41.75
O2M + ① + ③	44.78	45.23	45.78	45.22	45.25	42.97	27.11	**41.98**
O2M + ① + ② (Dyn)+ ③	**44.85**	**45.51**	**45.91**	**45.38**	**45.41**	**43.03**	27.23	41.92

Table 1: Translation performance of our methods on Zh→En/Ja and En→De/Fr tasks. Indiv means translation model of individual pair. O2M is the our baseline system. ① , ② and ③ denote our proposed three strategies of special label initialization, language-dependent positional embedding and the new parameter-sharing mechanism separately. ② (Dyn) and ② (Fixed) represent the two ways of language-dependent positional embedding method. For shared and language-dependent method, we set one-half of hidden units as shared units, and for another half, we use a quarter hidden units to denote two output languages respectively.

Johnson et al. (2017) method on Transformer framework as our baseline system (briefly, O2M method). From the first two lines, we can see that the O2M method cannot perform on par with the individually trained systems in most cases.

We mentioned before that our goal is to improve the universal one-to-many multilingual translation framework while maintaining the parameter sharing property. We can observe from the table that all our proposed strategies (last part in Table 1) improve the translation performance compared to the baseline (O2M). Specifically, the combined use of three strategies performs best and it can achieve the improvements up to 1.96 BLEU points (45.51 vs. 43.55 on Zh→En MT04). As for language-dependent positional embedding, we find that both fixed and dynamic styles perform similarly.

Our ultimate goal is to make the universal one-to-many framework as good as or better than the individually trained systems. Table 1 demonstrates some encouraging results. It is shown in the table that the universal one-to-many architecture enhanced with our strategies can outperform the individually trained models on three out of four language translations (Zh→En, Zh→Ja, En→Fr). The results verify the effectiveness of our proposed methods.

5.2 Comparison of Shared Unit Size

For the new parameter-sharing mechanism, it is an open question to decide how many hidden units should be shared and how many ones should be language dependent. To figure out this question, we further conduct an experiment to investigate

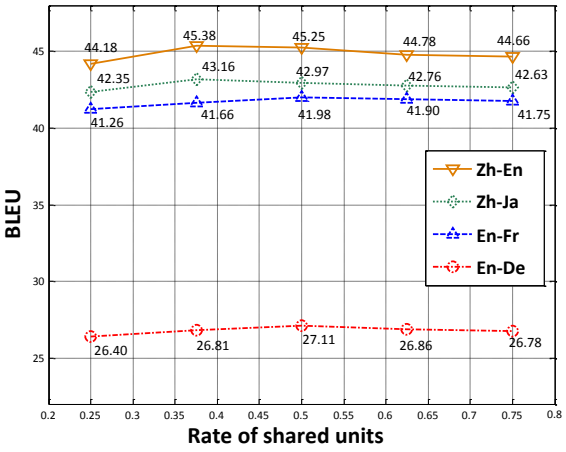

Figure 2: The comparison of different shared units.

different settings. For example, we keep a quarter of the hidden units of each decoder layer as shared and make the left three quarters evenly distributed to different target languages.

Figure 2 reports the results. We can observe different trends for different language pairs. On the En→De/Fr translation task, the performance is best when we share one-half of the hidden units. In contrast, it obtains the best results when we share only 37.5% of hidden units on Zh→En/Ja translation. It indicates that similar languages (De/Fr) can share more hidden units and languages with a great difference (En/Ja) may share less hidden units.

6 Related Work

In this work, we explore the balancing problem of shared and unique parameters, and attempt to

incorporate the language-dependent presentation features to distinguish different target languages under the scenario of one-to-many multilingual translation.

Multilingual translation has been extensively studied in Dong et al. (2015), Firat et al. (2016a), Luong et al. (2016) and Johnson et al. (2017). Owing to excellent translation performance and ease of use, many researchers (Blackwood et al., 2018; Lakew et al., 2018) have conduct translation of multiple languages based on the framework of Johnson et al. (2017) and Ha et al. (2016). As for low-resource translation scenario (Zoph et al., 2016; Chen et al., 2017; Wang et al., 2017b), similar to above method, Gu et al. (2018) enable sharing of lexical and sentence representation across multiple languages especially for low-resource multilingual NMT. Different from previous methods, our work mainly focuses on improving the one-to-many multilingual translation framework while sharing as many parameters as possible.

7 Conclusion

In this paper, we have proposed three effective strategies to improve the universal one-to-many multilingual translation, including special label initialization, language-dependent positional embedding and a new parameter-sharing mechanism. The empirical experiments on four language pairs demonstrate that our strategies can obtain significant improvement over the strong baseline, and can achieve comparable or even better results than the individually trained models.

For future work, we plan to extend our strategies on many-to-many multilingual translation scenarios, and explore other effective strategies to balance parameter sharing.

Acknowledgments

The research work described in this paper has been supported by the National Key Research and Development Program of China under Grant No. 2016QY02D0303 and the Natural Science Foundation of China under Grant No. 61673380. The research work in this paper has also been supported by Beijing Advanced Innovation Center for Language Resources and Sogou Inc.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. *In Proceedings of ICLR 2015*.

Graeme Blackwood, Miguel Ballesteros, and Todd Ward. 2018. Multilingual neural machine translation with task-specific attention. *In Proceedings of COLING 2018*, pages 3112–3122.

Yun Chen, Yong Cheng, Yang Liu, and Li Victor, O.K. 2017. A teacher-student framework for zero-resource neural machine translation. *In Proceedings of ACL 2017*, pages 1925–1935.

Daxiang Dong, Hua Wu, Wei He, Dianhai Yu, and Haifeng Wang. 2015. Multi-task learning for multiple language translation. *In Proceedings of ACL 2015*, pages 1723–1732.

Orhan Firat, Kyunghyun Cho, and Yoshua Bengio. 2016a. Multi-way, multilingual neural machine translation with a shared attention mechanism. *In Proceedings of NAACL-HLT 2016*, pages 866–875.

Orhan Firat, Baskaran Sankaran, Yaser Al-Onaizan, Fatos T. Yarman Vural, and Kyunghyun Cho. 2016b. Zero-resource translation with multi-lingual neural machine translation. *In Proceedings of EMNLP 2016*, pages 268–277.

Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N. Dauphin. 2017. Convolutional sequence to sequence learning. *arXiv preprint arXiv:1601.03317*.

Jiatao Gu, Hany Hassan, Jacob Devlin, and Victor OK Li. 2018. Universal neural machine translation for extremely low resource languages. *In Proceedings of NAACL-HLT 2018*, pages 344–354.

Thanh-Le Ha, Jan Niehues, and Alexander Waibel. 2016. Toward multilingual neural machine translation with universal encoder and decoder. *In Proceedings of IWSLT 2016*.

Hany Hassan, Anthony Aue, Chang Chen, Vishal Chowdhary, Jonathan Clark, Christian Federmann, Xuedong Huang, Marcin Junczys-Dowmunt, William Lewis, Mu Li, et al. 2018. Achieving human parity on automatic Chinese to English news translation. *arXiv preprint arXiv:1803.05567*.

Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2017. Google's multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, 5:339–351.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Surafel Melaku Lakew, Mauro Cettolo, and Marcello Federico. 2018. A comparison of transformer and recurrent neural networks on multilingual neural machine translation. *In Proceedings of COLING 2018*, pages 641–652.

Yichao Lu, Phillip Keung, Faisal Ladhak, Vikas Bhardwaj, Shaonan Zhang, and Jason Sun. 2018. A neural interlingua for multilingual machine translation. *arXiv preprint arXiv:1804.08198*.

Minh-Thang Luong, Quoc V Le, Ilya Sutskever, Oriol Vinyals, and Lukasz Kaiser. 2016. Multi-task sequence to sequence learning. *In Proceedings of ICLR 2016*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. *In Proceedings of ACL*, pages 311–318.

Mike Schuster and Kaisuke Nakajima. 2012. Japanese and korean voice search. *In Proceedings of ICASSP 2012*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Improving neural machine translation models with monolingual data. *In Proceedings of ACL 2016*, pages 86–96.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. *In Proceedings of NIPS*, pages 3104–3112.

Yaohua Tang, Fandong Meng, Zhengdong Lu, Hang Li, and Philip LH Yu. 2016. Neural machine translation with external phrase memory. *arXiv preprint arXiv:1606.01792*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, and Łukasz Kaiser. 2017. Attention is all you need. *In Proceedings of NIPS 2017*, pages 30–34.

Xing Wang, Zhengdong Lu, Zhaopeng Tu, Hang Li, Deyi Xiong, and Min Zhang. 2017a. Neural machine translation advised by statistical machine translation. *In Proceedings of AAAI 2017*.

Yining Wang, Yang Zhao, Jiajun Zhang, Chengqing Zong, and Zhengshan Xue. 2017b. Towards neural machine translation with partially aligned corpora. *In Proceedings of IJCNLP 2017*, pages 384–393.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

Jiajun Zhang and Chengqing Zong. 2016. Exploiting source-side monolingual data in neural machine translation. *In Proceedings of EMNLP 2016*, pages 1535–1545.

Yang Zhao, Yining Wang, Jiajun Zhang, and Chengqing Zong. 2018. Phrase table as recommendation memory for neural machine translation. *In Proceedings of IJCAI 2018*, pages 4609–4615.

Long Zhou, Wenpeng Hu, Jiajun Zhang, and Chengqing Zong. 2017. Neural system combination for machine translation. *In Proceedings of ACL 2017*, pages 378–384.

Barret Zoph and Kevin Knight. 2016. Multi-source neural translation. *In Proceedings of NAACL-HLT 2016*, pages 30–34.

Barret Zoph, Deniz Yuret, Jonathan May, and Kevin Knight. 2016. Transfer learning for low-resource neural machine translation. *In Proceedings of EMNLP 2016*, pages 1568–1575.