TermMind: Alibaba's WMT21 Machine Translation using Terminologies Task Submission

Ke Wang, Shuqin Gu, Boxing Chen, Yu Zhao, Weihua Luo, Yuqi Zhang
Machine Intelligence Technology Lab
Alibaba Group
Beijing, China
{wk258730,shuqin.gsq,boxing.cbx}@alibaba-inc.com,
kongyu@taobao.com,{weihua.luowh,chenwei.zyq}@alibaba-inc.com

Abstract

This paper describes our work in the WMT 2021 Machine Translation using Terminologies Shared Task. We participate in the shared translation terminologies task in English to Chinese language pair. To satisfy terminology constraints on translation, we use a terminology data augmentation strategy based on Transformer model. We used tags to mark and add the term translations into the matched sentences. We created synthetic terms using phrase tables extracted from bilingual corpus to increase the proportion of term translations in training data. Detailed pre-processing and filtering on data, in-domain finetuning and ensemble method are used in our system. Our submission obtains competitive results in the terminology-targeted evaluation.

1 Introduction

Terminology is important for domain-specific machine translation. Each domain has its own terminology, which represents the important and core concepts in the domain. In the workflow of human translation, terminology is an effective method to integrate the knowledge of human translator into machine translations (Wuebker et al., 2016; Cheng et al., 2016; Álvaro Peris et al., 2017).

One line of approach is "hard constraint". The terminology is ensured to appear in the translation by adding constraints in beam search decoding (Hokamp and Liu, 2017; Post and Vilar, 2018). However, the enforcement of terminology constraints tends to reduce the fluency of translation (Hasler et al., 2018), especially when there are multiple constraints or the constraint is noisy (Susanto et al., 2020). Another line of approach is "soft constraint". Training data is augmented with placeholders or additional terminology translations (Arthur et al., 2016; Song et al., 2019; Dinu et al., 2019; Chen et al., 2020; Ailem et al., 2021a).

The above methods assume that the terminology translations are good ones. However, in industry or real world the terminology translations may be noisy (Li et al., 2020). And in the human translation workflows the terminology constraints usually need to be applied hierarchically according to priority. In these scenarios one source term will have more than one translation. Therefore, we are happy to participate in this task and develop the method to deal with 1-to-many term translations in neural machine translation systems.

The structure of the paper is as follows. Section 2 describes the dataset, data pre-processing and selection. We introduce details of our system in Section 3. The experiment settings, terminologies used in training and main results are introduced in Section 4. Finally, we conclude our work in Section 5.

2 Data

2.1 Data Source

For this task, we utilize parallel data from English to Chinese language provided in WMT2021: ParaCrawl v7.1, News Commentary v16, Wiki Titles v3, UN Parallel Corpus V1.0, CCMT Corpus and WikiMatrix. In addition, we also require Chinese monolingual data from News crawl and News Commentary corpora for back translation.

2.2 Data Pre-processing

For all datasets, we tokenize English text with Moses[1] and the Chinese text with Jieba[2] tokenizer. We create a joint source and target BPE vocab (Sennrich et al., 2016) with 40k merge operations using filtered bilingual dataset as described in Section 2.3, resulting in a vocabulary with size of 63K words.

2.3 Data Selection

According to the previous works (Li et al., 2019; Sun et al., 2019), we selected data for training with

[1]https://github.com/moses-smt/mosesdecoder
[2]https://github.com/fxsjy/jieba

Proceedings of the Sixth Conference on Machine Translation (WMT), pages 851–856
November 10–11, 2021.

Source	Those most at risk of COVID-19 **infection** and serious complications are the elderly and those with weakened immune systems or underlying health conditions like cardiovascular disease, **diabetes**, hypertension, **chronic respiratory disease**, and cancer.
Constraint	diabetes 糖尿病 infection 传染病 | 感染 chronic respiratory disease 慢性呼吸道疾病 | 慢性呼吸系统疾病
Match	Those most at risk of COVID-19 <term tgt=" 传染病 | 感染"> **infection** </term> and serious complications are the elderly and those with weakened immune systems or underlying health conditions like cardiovascular disease , <term tgt=" 糖尿病"> **diabetes** </term> , hypertension , <term tgt=" 慢性呼吸道疾病 | 慢性呼吸系统疾病"> **chronic respiratory disease** </term>, and cancer .
Tag & Mask	Those most at risk of COVID-19 <S> **[MASK]** <C> **传染病 [SEP] 感染** </C> and serious complications are the elderly and those with weakened immune systems or underlying health conditions like cardiovascular disease , <S> **[MASK]** <C> **糖尿病** </C> , hypertension , <S> **[MASK] [MASK] [MASK]** <C> **慢性呼吸道疾病 [SEP] 慢性呼吸系统疾病** </C> , and cancer .
Target	COVID - 19 **感染** 和严重并发症风险最高的是老年人、免疫力低下者或患有心血管疾病、**糖尿病** 、高血压、**慢性呼吸道疾病** 和癌症等基础性疾病的人群。

Table 1: Illustration of the terminology data augmentation.

the following schemes:

- Remove the texts of over 120 tokens.
- Remove bitexts with length ratios greater than 3.
- Remove texts with special HTML tags.
- Remove duplicate bitexts.
- Remove texts with fastText-langid (Joulin et al., 2016b,a), which is an open-source tool for text-based language identification.
- Remove Chinese sentences when the proportion of Chinese tokens is less than 0.8.

3 System Overview

In this section, we will describe the details about the model and techniques of our work. First, we will introduce the terminology data augmentation strategy to improve terminology translation accuracy. Then, different transformer model architectures we adopted in the paper will be depicted. Finally, we will introduce several strategies to train our models for performance improvement.

3.1 Terminology Learning

We use a terminology data augmentation strategy to encourage neural machine translation (NMT) to satisfy terminology constraints. The key point of term translation idea is that when multiple possible terms are encountered, the NMT model is preferred copying the correct terms, and the terms are correctly placed in the output sentence. Encouraged by the work (Chen et al., 2020; Ailem et al., 2021b), we use tags to specify the term constraints in the source sentence. We have given an example in the Table 1. A **Source** sentence could have more than one terms. Each term could have multiple **Constraint**. The source term is indicated as tag <S>, and the pair <C> </C> is used to label target term. Tag [SEP] is used to separate multiple possible target terminologies, when there are 1-m term constraints. Following the work (Ailem et al., 2021b) we mask the source tokens of a term to strengthen the learning of target term tokens. In table 1, term source tokens are marked in red, and the term target tokens are in blue. **Tag & Mask** shows an example. <S> indicates term constraint "infection", but the token "infection" is masked with [MASK]. "infection" 's translations " 传染病" and " 感染" are enclosed by <C> and </C>, separated by [SEP].

The official term table is small. We extract a phrase table from the bilingual training data and filter it as synthetic terms. More details are described in Section 4.2.

3.2 Model Architecture

In our systems, we adopt three different model architectures with Transformer (Vaswani et al., 2017):

- **BIG** Transformer is the Transformer-Base model (Vaswani et al., 2017) with 4096 feed-forward network (FFN) width and 16 attention heads.

- **DEEP** Transformer (Sun et al., 2019) is Transformer-Base model with 20 encoder layers.

- **LARGE** Transformer (Ng et al., 2019) is Transformer-Base model with 8192 FNN inner width.

We use 6 decoder layers for all models. Our models are implemented with open-source toolkit Fairseq (Ott et al., 2019).

3.3 Optimization Strategies

To further improve the translation performance, several common strategies are used to train our models such as Back Translation, Finetuning and Ensemble. The strategies are performed basically sequentially. We use the terminology data augmentation on back translation and fine-tuning datasets to train models.

3.3.1 Back Translation

Back translation is a data augmentation technique to incorporate monolingual data into NMT model. Similar to previous work (Edunov et al., 2018), we use back translation to improve the model performance. We first train a Chinese-to-English Transformer-Deep NMT model based on bilingual training dataset. The NMT model is applied to translate Chinese monolingual corpus to English. The pseudo parallel corpus is used to train models together with the bilingual training dataset.

3.3.2 Finetuning

Previous study (Sun et al., 2019) demonstrate that fine-tuning a model on in-domain data effectively improve the model performance. For the term translation task, two fine-tuning datasets are used in our works. We use two kinds of finetuning datasets to train the model sequentially.

Base FT We use all the previous English $\rightarrow$ Chinese development and test dataset as fine tuning corpus, including WMT2017 development data, WMT2017 test data, WMT2018 test data, WMT2019 test data and WMT2020 test data.

In-domain FT To use in-domain dataset to fine tune the model, we perform data selection on out-of-domain corpus based on in-domain n-gram match. The key idea is to select sentence pairs from the large out-of-domain corpus that are similar to the in-domain data. We use the bilingual training data as the out-of-domain corpus and WMT2021 term development dataset as the in-domain corpus. We extract 1-3grams from the in-domain and out-of-domain dataset. After exclude the ngrams from the out-of-domain data, the left in-domain ngrams are applied to match relevant sentence from the bilingual training.

In our work, we use source and target to select in-domain dataset respectively and finally the two sets are combined to train the model.

3.3.3 Ensemble

Model ensemble is an effective strategy widely used in real-world tasks. At each step of translation prediction, it combines the predicted probabilities of different models. We use the log-avg strategy to ensemble the different NMT models. The model diversity is an important factor for ensemble. We have trained three Transformer models with different architectures including the variants of Transformer-BIG, Transformer-DEEP and Transformer-LARGE.

4 Experiments

4.1 Setups

Our models are implemented in Fairseq Library[3]. All the single models are trained based on 4 NVIDIA P100-PCIe GPUs, each with 16 GB memory. The models are optimized with Adam algorithm (Kingma and Ba, 2015) with $\beta_1 = 0.9$ and $\beta_2 = 0.98$. We set max learning rate to 0.001 when training a single model from scratch and 0.0007 when fine-tuning the model. The batch size is set to 2048 tokens per GPU. The 'update-freq' parameter in Fairseq is set to 16 when training a single model from scratch and 4 when fine-tuning the model. The dropout (Gal and Ghahramani, 2016) probabilities are set to 0.1 in all experiments. We select the checkpoint with the best BLEU score on development set as the final checkpoint in each training. Evaluation of results focus on translation accuracy and term translation consistency. We evaluate translation accuracy with SacreBLEU (Post, 2018), which is a

[3] https://github.com/pytorch/fairseq

System	Model	BLEU	Exact-Match Accuracy	Window Overlap Accuracy (2/3)	1-TERm Score
Baseline NMT	LARGE	37.8	65.89	16.52/16.30	36.48
Data Selection	BIG	36.09	71.28	15.82/16.57	30.08
	DEEP	35.85	74.76	17.01/17.62	29.74
	LARGE	36.17	69.23	14.94/15.33	30.91
	+Ensemble	**38.22**	**74.52**	**17.47/17.56**	**33.00**
+Back Translation	BIG	37.72	73.92	17.28/17.71	33.33
	DEEP	37.74	73.92	17.55/18.05	33.85
	LARGE	37.50	72.36	15.97/16.53	32.68
	+Ensemble	**39.39**	**75.60**	**17.87/18.62**	**33.90**
+Base FT	BIG	38.12	71.86	17.57/18.14	34.68
	DEEP	38.17	72.72	17.32/18.18	33.74
	LARGE	40.97	72.95	17.26/18.40	38.03
	+Ensemble	**41.43**	**75.72**	**18.91/19.89**	**38.17**
+In-domain FT	BIG	39.12	71.63	17.09/17.71	36.25
	DEEP	38.33	73.08	17.48/18.25	34.60
	LARGE	41.11	72.72	17.04/18.24	38.48
	+Ensemble	**41.71**	**76.68**	**18.88/19.88**	**39.05**

Table 2: Evaluation results on the WMT2021 English → Chinese development set.

case-sensitive detokenized BLEU. Terminology-targeted metrics (Anastasopoulos et al., 2021) is used to term translation consistency, including exact-match accuracy, window overlap metric and terminology-biased Translation Edit Rate (TERm)[4]. The exact-match accuracy is defined as the ratio between the number of matched source terms and the total number of source terms. The window overlap metric is to evaluate the position accuracy of each target term in translation. The TERm, a metric based on TER (Snover et al., 2006), focuses on penalizing errors related to terminology tokens.

4.2 Terminologies

In order to increase the proportion of term translations in training data, we extract phrase tables from bilingual training corpus to create synthetic term translations. First, we use FastAlign (Dyer et al., 2013) to generate word alignments. Second, based on the word alignments we extract a phrase table by using moses (Koehn et al., 2007) with default settings. We use count-based pruning (Zens et al., 2012) and fastText-langid (Joulin et al., 2016b,a) to filter the phrase table. The count threshold is set to 200. Finally, the term table for the terminology data augmentation is obtained by combining the English → Chinese term table from WMT2021 and the filtered phrase table. The target terms corresponding to the same source term are separated by '|'. The term table contains 1-to-1 and 1-to-many term pairs. The term information with tags will be added into source sentences when they match, as shown in Table 1. 15.4% of the training sentences with the term information. We have used only the official terms from WMT 2021 for the test and dev datasets.

4.3 Results

Table 2 shows the English → Chinese translation results on WMT2021 terminologies development dataset, including BLEU, exact-match accuracy, window overlap accuracy (2/3) and 1-TERm Score. We train multiple single models in each settings and report the best BLEU scores in Table 2. The baseline is the LARGE transformer model using the bilingual training data. Our models using terminology data augmentation are called Term model. Ensemble models of each step consist of 3 single models: BIG, DEEP and LARGE models. As shown in Table 2, the LARGE Term model using the bilingual dataset boosts the exact-match

[4]https://github.com/mahfuzibnalam/terminology_evaluation

accuracy from 65.89 to 69.23. Under each setting, the performance of the ensemble Term models is higher than that of the best single Term model by a BLEU score of 0.46 to 2.05. After adding back translation, we improved the BLEU score to 39.39 and the exact-match accuracy to 75.6 on ensemble models. The base FT can achieve 2 BLEU and 4.3 1-TERm score improvements on ensemble models. After applying In-domain FT, We achieve 0.96 exact-match accuracy and 0.88 1-TERm score improvements on ensemble models.

Considering the effectiveness of fine-tuning, we use WMT2021 development data to fine tune the model after completing 100 steps. In our final submission, we selected sentences with the higher probability from the translations of the ensemble Term model and the ensemble NMT model.

5 Conclusion

This paper presents the submissions by Alibaba for WMT 2021 English to Chinese translation terminologies task. We have applied a terminology data augmentation method to integrate term translations into NMT systems. We also used a series of data filtering strategies, fine-tuning and ensemble methods to improve the system performance. Experimental results show the method can improve terminologies translation performance.

6 Acknowledgments

This work is supported by National Key R&D Program of China (2018YFB1403202).

References

Melissa Ailem, Jinghsu Liu, and Raheel Qader. 2021a. Encouraging neural machine translation to satisfy terminology constraints. *CoRR*, abs/2106.03730.

Melissa Ailem, Jinghsu Liu, and Raheel Qader. 2021b. Encouraging neural machine translation to satisfy terminology constraints. *arXiv preprint arXiv:2106.03730*.

Antonios Anastasopoulos, Laurent Besacier, James Cross, Matthias Gallé, Philipp Koehn, Vassilina Nikoulina, et al. 2021. On the evaluation of machine translation for terminology consistency. *arXiv preprint arXiv:2106.11891*.

Philip Arthur, Graham Neubig, and Satoshi Nakamura. 2016. Incorporating discrete translation lexicons into neural machine translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1557–1567, Austin, Texas. Association for Computational Linguistics.

Guanhua Chen, Yun Chen, Yong Wang, and Victor O.K. Li. 2020. Lexical-constraint-aware neural machine translation via data augmentation. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pages 3587–3593. International Joint Conferences on Artificial Intelligence Organization. Main track.

Shanbo Cheng, Shujian Huang, Huadong Chen, Xinyu Dai, and Jiajun Chen. 2016. Primt: A pick-revise framework for interactive machine translation. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1240–1249.

Georgiana Dinu, Prashant Mathur, Marcello Federico, and Yaser Al-Onaizan. 2019. Training neural machine translation to apply terminology constraints. *CoRR*, abs/1906.01105.

Chris Dyer, Victor Chahuneau, and Noah A. Smith. 2013. A simple, fast, and effective reparameterization of IBM model 2. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 644–648, Atlanta, Georgia. Association for Computational Linguistics.

Sergey Edunov, Myle Ott, Michael Auli, and David Grangier. 2018. Understanding back-translation at scale. *arXiv preprint arXiv:1808.09381*.

Yarin Gal and Zoubin Ghahramani. 2016. A theoretically grounded application of dropout in recurrent neural networks. *Advances in neural information processing systems*, 29:1019–1027.

Eva Hasler, Adrià de Gispert, Gonzalo Iglesias, and Bill Byrne. 2018. Neural machine translation decoding with terminology constraints. In *NAACL-HLT (2)*, pages 506–512. Association for Computational Linguistics.

Chris Hokamp and Qun Liu. 2017. Lexically constrained decoding for sequence generation using grid beam search. *CoRR*, abs/1704.07138.

Armand Joulin, Edouard Grave, Piotr Bojanowski, Matthijs Douze, Hérve Jégou, and Tomas Mikolov. 2016a. Fasttext. zip: Compressing text classification models. *arXiv preprint arXiv:1612.03651*.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2016b. Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, et al. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th annual meeting of the association for computational linguistics companion volume proceedings of the demo and poster sessions*, pages 177–180.

Bei Li, Yinqiao Li, Chen Xu, Ye Lin, Jiqiang Liu, Hui Liu, Ziyang Wang, Yuhao Zhang, Nuo Xu, Zeyang Wang, et al. 2019. The niutrans machine translation systems for wmt19. In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 257–266.

Huayang Li, Guoping Huang, Deng Cai, and Lemao Liu. 2020. Neural machine translation with noisy lexical constraints. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:1864–1874.

Nathan Ng, Kyra Yee, Alexei Baevski, Myle Ott, Michael Auli, and Sergey Edunov. 2019. Facebook fair's wmt19 news translation task submission. *arXiv preprint arXiv:1907.06616*.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 48–53, Minneapolis, Minnesota. Association for Computational Linguistics.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.

Matt Post and David Vilar. 2018. Fast lexically constrained decoding with dynamic beam allocation for neural machine translation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1314–1324, New Orleans, Louisiana. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas: Technical Papers*, pages 223–231.

Kai Song, Yue Zhang, Heng Yu, Weihua Luo, Kun Wang, and Min Zhang. 2019. Code-switching for enhancing NMT with pre-specified translation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 449–459, Minneapolis, Minnesota. Association for Computational Linguistics.

Meng Sun, Bojian Jiang, Hao Xiong, Zhongjun He, Hua Wu, and Haifeng Wang. 2019. Baidu neural machine translation systems for wmt19. In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 374–381.

Raymond Hendy Susanto, Shamil Chollampatt, and Liling Tan. 2020. Lexically constrained neural machine translation with levenshtein transformer. *CoRR*, abs/2004.12681.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Joern Wuebker, Spence Green, John DeNero, Saša Hasan, and Minh-Thang Luong. 2016. Models and inference for prefix-constrained machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 66–75, Berlin, Germany. Association for Computational Linguistics.

Richard Zens, Daisy Stanton, and Peng Xu. 2012. A systematic comparison of phrase table pruning techniques. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 972–983.

Álvaro Peris, Miguel Domingo, and Francisco Casacuberta. 2017. Interactive neural machine translation. *Computer Speech & Language*, 45:201–220.