Non-literal Neural Machine Translation by Exploiting Non-literal Bitext

Lianhao Yu[1] Naoki Yoshinaga[2] Masato Neishi[1] Yuma Tsuta[1]
[1]The University of Tokyo [2]Institute of Industrial Science, The University of Tokyo
{yu-l, ynaga, neishi, tsuta}@tkl.iis.u-tokyo.ac.jp

Abstract

To avoid unnatural-sounding translations produced by existing neural machine translation (NMT) systems, we propose training an NMT model for non-literal translations by exploiting possibly non-literal translations in training data. Specifically, we split the training bitext into two sets in terms of non-literalness and applied domain adaptation techniques to acquire an NMT model adapted to non-literal translations. Our best-performing model achieved a BLEU score of 25.20 and a COMET score of 0.7932 in producing non-literal translations.

1 Introduction

Neural machine translation (NMT) utilizes a single large neural network to directly transform the source sentence into the target sentence [1, 2, 3], which brought great improvements in terms of the translation quality. NMT systems have already been developed and deployed for various practical uses. For example, Google Translate is a service that enables to translate sentences, documents, and websites among over 100 languages, and helps its users to overcome the obstacle of unknown languages and access a wide range of information and resources.

Although the translations produced by NMT systems are mostly correct, they are sometimes unnatural for native speakers and thus not quite understandable due to the existing differences between languages and cultures [4]. For comparison, human translations of the same source sentences are sometimes non-literal but more natural for native speakers, and thus more appropriate, as shown in Table 1. Considering the relatively common occurrences of non-literal translations, it is natural and promising to exploit such translations in existing parallel corpora to improve the performances of NMT models in terms of fidelity, adequacy, and fluency of translations [5, 6, 4].

Table 1 Examples of English-Chinese non-literal translations.

EN: In his hand, it **became buoyant**, **in sync with the motion of** the waves that he **made** with his arms. ZH: 它在他的手中浮 动，随着他用手臂模 仿的波浪一同 沉浮。 （“It **floated** in his hand, **rising and falling with** the waves he **mimicked** with his arm.”）
EN: Success is in **the doing**, and failures **are celebrated and analyzed.** ZH: 成功就在 过程中。我 们庆祝和分析失败。 （“Success is in **the process**. **We celebrate and analyze** failure.”）

In this paper, we propose to train an NMT model for non-literal translations from existing bitext. We obtain training data specialized in non-literal translations by ranking parallel sentences in the bitext in terms of non-literalness and splitting them into two sets in which one set is likely to be more non-literal than the other. Regarding the non-literalness as a kind of domain, we then apply domain adaptation techniques to train an NMT model specialized for non-literal translations using the training data. Specifically, we utilize multi-domain learning [7] and curriculum learning [8] for training. The former is to add domain signals for non-literal and literal sets of the training data and mix them to fine-tune a pre-trained language model. The latter is to fine-tune a pre-trained model first on the literal set and next on the non-literal set.

We evaluate our method on English-to-Chinese translation using OpenSubtitles and TED2020 datasets. To evaluate our method, we build a specialized evaluation dataset by manually extracting non-literal translations to find that our methods achieved even better results when we forced the models to generate non-literal translations. Experimental results on this specialized dataset show that the models trained by our methods perform better than the baseline in generating translations that are natural for native speakers.

2 Related Work

Non-literal translations have been long studied by translation theorists and linguists [9, 6]. Several studies have been done with different motivations to detect non-literal translations in existing corpora. Chen et al. [5] aims to mine lexical and phrasal non-literal translations for human translators' reference and to inspire improvements in machine translation. The authors designed an algorithm based on attention scores for searching possible non-literal translations in a bilingual corpus of Chinese and English. Several studies have been done for automatically detecting non-literal translations in existing corpora [10, 6, 4] for non-literal translations can bring difficulties for automatic word alignment, and the training of NMT systems [11]. For example, Zhai et al. [4] first annotated parallel corpora with categories of non-literal translations and then conducted experiments on the detection of phrasal non-literal translations. These studies focus on detecting non-literal translations, whereas our study focuses on generating non-literal translations.

3 Proposed Methods

In this section, we first describe our methods of finding possibly non-literal translations from existing bitext. We then use these possibly non-literal translations as the target-domain training data to perform multi-domain learning [7] or curriculum learning [8].

3.1 Ranking Translation Pairs

To begin with, we propose unsupervised and semi-supervised methods of ranking translation pairs of existing parallel corpora. We use the resulting rankings to obtain possibly non-literal and literal translations while changing their proportion as a hyperparameter.

Alignment-based score Since non-literal translations can bring difficulties for automatic sentential alignment [11, 4], we assume that parallel sentences with low alignment scores are more non-literal translations. Given a translation pair $< e, f >$, we use Vecalign [12] to compute the alignment cost $vecalign(e, f)$ to extract possibly non-literal translations with high cost values. However, such translations can also be noises or low-quality translations. To reduce such cases, we design an additional score to give higher ranks to parallel sentences with low alignment scores surrounded by ones with high alignment scores. Specifically, given a translation pair $\langle e, f \rangle$ and its previous and next translations $\langle e_{\text{prev}}, f_{\text{prev}} \rangle$, and $\langle e_{\text{next}}, f_{\text{next}} \rangle$, we used the value of

$$\frac{\mathsf{vecalign}(e, f)}{\mathsf{vecalign}(e_{\text{prev}}, f_{\text{prev}}) + \mathsf{vecalign}(e_{\text{next}}, f_{\text{next}})}$$

for ranking.

NMT-based score As NMT models are likely to generate literal translations, the outputs of an off-the-shelf NMT model [1)] can be used to find possibly non-literal translations. We assume that non-literal translations are semantically similar but superficially dissimilar to literal translations. Therefore, we computed and combined different evaluation metrics as scores to rank translation pairs. Specifically, given a parallel sentence, a pair of source sentence e and target sentence f, and a translation obtained by the off-the-shelf NMT model, $\hat{f}$, we computed BERTScore [13] and chrF [14] and used the value of

$$-\frac{\text{chrF}(\text{f}, \hat{\text{f}})}{\text{BERTScore}(\text{f}, \hat{\text{f}})}$$

for ranking.

3.2 Training NMT Models

We design two strategies to train a non-literal NMT model using the obtained possibly non-literal and literal translations. We adopt a pre-trained *mT5* [15, 16] as a backbone of our NMT models.

Multi-domain Learning (MDL) We regard non-literal and literal translations as data from two domains and perform multi-domain learning (**MDL**) using all the data. Specifically, we adopt domain-token mixing [7], which predicts a domain token (here, <NLT> and <LT> for non-literal and literal translations), before decoding a target sentence. This enables us to train the model to predict the type of translations (non-literal or literal) and generate a translation conditioned on the prediction.

Curriculum Learning Another idea is to regard non-literal translations as translations that are more difficult to produce than literal ones. Then, following the idea of curriculum learning [8, 17, 18] to mimic the human education process, we first finetune the model to generate literal translations using the possibly literal translations and then to generate non-literal translations using the possibly non-literal translations.

1) https://huggingface.co/K024/mt5-zh-ja-en-trimmed

4 Experiments

We apply our methods to English-to-Chinese translation using OpenSubtitles and TED2020 datasets. We evaluate the ranking methods for non-literal translations and the proposed non-literal NMT models obtained by our ranking and training methods.

4.1 Settings

Data We train and evaluate our methods on the whole dataset of TED2020 on the English-Chinese language pair. We prepare the data by first doing some simple noise filtering and then splitting the dataset by the ratio of 90 : 5 : 5 for training, validation, and test, respectively. As a result, we obtained about $246k$, $14k$, and $14k$ sentence pairs for the three parts, respectively.

As for the training data, we utilize the ranking methods introduced in § 3.1 to sort the translation pairs and take the top X% of data as the non-literal part and the rest as the literal part where X is a hyperparameter.

As for the test data, we additionally manually annotated part of the translation pairs and picked out actual non-literal ones to form a specialized test set to evaluate the ability to generate non-literal translations of the NMT models trained by the proposed methods. Currently, this specialized test set contains 222 non-literal translation pairs, which will be further expanded in the future.

To evaluate ranking methods of translation pairs, due to limited time and labor for manual annotation for evaluation, we only took the first 10% data of both datasets, which results in about $1.2M$ sentence pairs of OpenSubtitles and $46k$ of TED2020.

Models We finetune *mT5* [15, 16] to obtain different NMT models for non-literal translations using combinations of ranking and training methods described in § 3.1 and § 3.2. For each ranking method, we take top-X% of data as the non-literal part and the rest as the literal part and use the data for different training methods. For mixed fine-tuning, we set the value of X to 1, 5, 10, 20, and 40 to train different models, while for curriculum learning we only set the value to 30 at the moment. As for baselines, we finetuned a pre-trained mT5 model on the training data we prepared.

Evaluation metrics As for ranking translation pairs, we take the top-200 pairs of ranked translations from the

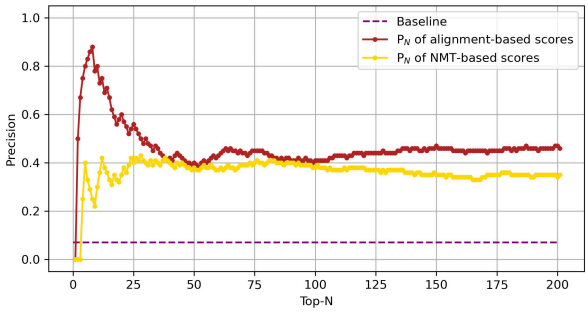

Figure 1 Top-N Precision of non-literal translations on OpenSubtitles.

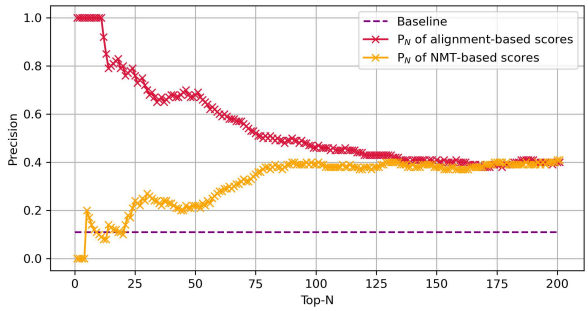

Figure 2 Top-N precision of non-literal translations on TED2020.

10% of entire data and manually annotate non-literal ones among them. We then compute top-N precision P_N of non-literal translations as below.

$$P_N = \frac{\#\text{non-literal at rank } N}{N}$$

As for non-literal NMT models, we adopt BLEU [19] and COMET-22-DA [20, 21] scores. We utilize SacreBLEU [22] to compute BLEU scores with reference translations and predicted translations as inputs. As for COMET scores, we use "*Unbabel/wmt22-comet-da*"[2)] model with inputs of source sentences, reference translations, and predicted translations.

Training details We choose the *mT5-small* model as the base model and implement different models by the Transformers library of Huggingface [23]. We set the training epoch to 8 and save the best model checkpoints for final evaluation, which have the lowest training loss. As for models trained by curriculum learning, we additionally choose the best model trained on the literal part of data before training on the non-literal part. As for other hyperparameters and settings, we keep them by default.

4.2 Results and Analysis

Results of Ranking Translations Figure 1 and 2 show the results of evaluating ranking methods for non-

2) https://huggingface.co/Unbabel/wmt22-comet-da

Table 2 Evaluation results of NMT models on the specialized test set of non-literal translations. "X%" indicates that the top-X% of translation pairs are used as the non-literal translations and the rest as the literal ones in training.

Model	BLEU	COMET
Baseline	24.99	0.7868
MDL, 1% non-literal by alignment	25.10	0.7899
MDL, 5%	24.68	0.7800
MDL, 10%	24.90	0.7875
MDL, 20%	25.07	**0.7907**
MDL, 40%	24.90	0.7829
MDL, 1% non-literal by NMT	24.48	0.7795
MDL, 5%	24.45	0.7851
MDL, 10%	**25.15**	0.7858
MDL, 20%	24.77	0.7887
MDL, 40%	24.60	0.7857
Curriculum, 30% non-literal by alignment	24.24	0.7758
Curriculum, 30% non-literal by NMT	24.10	0.7785

literal translations. As for the baseline, we randomly sample 200 sentence pairs, manually annotate non-literal ones, and compute the ratio of non-literal translations as the overall precision, which resulted in 6.5% and 10.5% (shown as a horizontal dotted line in the figures) for OpenSubtitles and TED2020, respectively. We can see from the results that the proposed methods outperform the baseline on both datasets, which proves our hypothesis that top-ranked translation pairs are more likely to be non-literal.

Results of Non-literal NMT Models As for training non-literal translators, we get results by evaluating models on the specialized non-literal test set and the overall test set. The latter is shown in Table 2 while the former can be found in the Appendix. In addition, we force models trained by multi-domain learning to generate non-literal translations by using <NLT> at the start of decoding when evaluated on the specialized test set. We report this part of the results in Table 3. We can see from the results in Table 2 and 3 that the **MDL, 20%** and **MDL, 10%** models perform better than other models. In addition, when forced to generate non-literal translations, they perform even better, which proves the effectiveness of our methods for training NMT models for non-literal translations.

Examples of Generated Translations We present some generated samples of non-literal translators as shown in Table 4. We can see from the examples that the generated translations have common features of non-literal translations in that they are fluent and natural but have some semantic divergences with the source sentences.

Table 3 Evaluation results of NMT models on the specialized test set when forced to generate non-literal translations.

Model	BLEU	COMET
MDL, 1% non-literal by alignment	24.90	0.7867
MDL, 5%	24.59	0.7799
MDL, 10%	24.82	0.7906
MDL, 20%	25.20	**0.7932**
MDL, 40%	24.86	0.7818
MDL, 1% non-literal by NMT%	24.41	0.7788
MDL, 5%	24.52	0.7859
MDL, 10%	**25.25**	0.7885
MDL, 20%	24.76	0.7888
MDL, 40%	24.16	0.7868

Table 4 Generated non-literal translations by the **MDL, 20%** and the **MDL, 10%** model. The English sentences are the source sentences while the Chinese sentences are the forcedly generated non-literal translations.

From MDL, 20%, non-literal by alignment-based scores
EN: **It's very expensive to drive** that much, and as we've seen, the middle class is **struggling to hold on**. ZH: 开车非常昂 贵, 正如我 们所看到的, 中 产阶级挣扎不堪。 (“**Driving is very expensive**, and as we've seen, the middle class is **struggling unbearably**.”)
From MDL, 10%, non-literal by NMT-based scores
EN: But **the suggestion I want to put to you** today is that **there's something** fundamentally wrong **with this model**. ZH: 但是我今天想告 诉你们的是, 这个模型根本上是错误的。 (“But **what I want to tell you** today is that **this model is** fundamentally wrong.”)

5 Conclusions

In this paper, we propose to train an NMT model for producing non-literal translations by exploiting translations that are possibly non-literal in existing corpora. We first rank translation pairs by different scores and extract translation pairs that are possibly non-literal. We then adopt two domain adaptation techniques. One is to add domain signals to the non-literal and literal parts of the data and train the model by multi-domain learning. The other is to first train the model on the literal part of data and second on the non-literal part by curriculum learning. The results show that our methods effectively improve the NMT model for non-literal translations.

As for future work, we plan to conduct further experiments with improved methods and on other parallel corpora, such as OpenSubtitles.

Acknowledgement

This work was partially supported by the special fund of Institute of Industrial Science, The University of Tokyo, by JSPS KAKENHI Grant Number JP21H03494.

References

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

[2] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. **Advances in neural information processing systems**, Vol. 27, , 2014.

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

[4] Yuming Zhai, Gabriel Illouz, and Anne Vilnat. Detecting non-literal translations by fine-tuning cross-lingual pretrained language models. In **Proceedings of the 28th International Conference on Computational Linguistics**, pp. 5944–5956, 2020.

[5] Qi Chen, Olivia OY Kwong, and Jingbo Zhu. Detecting free translation in parallel corpora from attention scores. In **Proceedings of the 32nd Pacific Asia Conference on Language, Information and Computation**, 2018.

[6] Yuming Zhai, Lufei Liu, Xinyi Zhong, Gbariel Illouz, and Anne Vilnat. Building an English-Chinese parallel corpus annotated with sub-sentential translation techniques. In **Proceedings of the Twelfth Language Resources and Evaluation Conference**, pp. 4024–4033, 2020.

[7] Denny Britz, Quoc Le, and Reid Pryzant. Effective domain mixing for neural machine translation. In **Proceedings of the Second Conference on Machine Translation**, pp. 118–126, 2017.

[8] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In **Proceedings of the 26th annual international conference on machine learning**, pp. 41–48, 2009.

[9] Lucía Molina and Amparo Hurtado Albir. Translation techniques revisited: A dynamic and functionalist approach. **Meta**, Vol. 47, No. 4, pp. 498–512, 2002.

[10] Yuming Zhai, Pooyan Safari, Gabriel Illouz, Alexandre Allauzen, and Anne Vilnat. Towards recognizing phrase translation processes: Experiments on english-french. **arXiv preprint arXiv:1904.12213**, 2019.

[11] Dun Deng and Nianwen Xue. Translation divergences in chinese–english machine translation: An empirical investigation. **Computational Linguistics**, Vol. 43, No. 3, pp. 521–565, 2017.

[12] Brian Thompson and Philipp Koehn. Vecalign: Improved sentence alignment in linear time and space. In **Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)**, pp. 1342–1348, 2019.

[13] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with bert. **arXiv preprint arXiv:1904.09675**, 2019.

[14] Maja Popović. chrf: character n-gram f-score for automatic mt evaluation. In **Proceedings of the tenth workshop on statistical machine translation**, pp. 392–395, 2015.

[15] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. **The Journal of Machine Learning Research**, Vol. 21, No. 1, pp. 5485–5551, 2020.

[16] Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. mt5: A massively multilingual pre-trained text-to-text transformer. **arXiv preprint arXiv:2010.11934**, 2020.

[17] Xin Wang, Yudong Chen, and Wenwu Zhu. A survey on curriculum learning. **arXiv preprint arXiv:2010.13166**, 2020.

[18] Petru Soviany, Radu Tudor Ionescu, Paolo Rota, and Nicu Sebe. Curriculum learning: A survey. **International Journal of Computer Vision**, Vol. 130, No. 6, pp. 1526–1565, 2022.

[19] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In **Proceedings of the 40th annual meeting of the Association for Computational Linguistics**, pp. 311–318, 2002.

[20] Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. Comet: A neural framework for mt evaluation. **arXiv preprint arXiv:2009.09025**, 2020.

[21] Ricardo Rei, José GC De Souza, Duarte Alves, Chrysoula Zerva, Ana C Farinha, Taisiya Glushkova, Alon Lavie, Luisa Coheur, and André FT Martins. Comet-22: Unbabel-ist 2022 submission for the metrics shared task. In **Proceedings of the Seventh Conference on Machine Translation (WMT)**, pp. 578–585, 2022.

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

[23] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. Huggingface's transformers: State-of-the-art natural language processing. **arXiv preprint arXiv:1910.03771**, 2019.

Table 5 Evaluation results of NMT models on the overall test set.

Model	BLEU	COMET
Baseline	23.97	0.7809
MDL, 1% non-literal by alignment	24.06	0.7803
MDL, 5%	23.96	0.7799
MDL, 10%	23.92	0.7816
MDL, 20%	24.15	0.7812
MDL, 40%	24.06	0.7812
MDL, 1% non-literal by NMT	24.02	0.7813
MDL, 5%	23.82	0.7806
MDL, 10%	23.92	0.7799
MDL, 20%	24.15	0.7816
MDL, 40%	**24.23**	**0.7823**
Curriculum, 30% non-literal by alignment	23.45	0.7724
Curriculum, 30% non-literal by NMT	23.41	0.7742

A Appendix

A.1 Results on Overall Test Set

The results of evaluating non-literal translators on the overall test set are shown in Table 5. As for proposed models, the **MDL, 40%** model performs relatively the best, which is different from the results on the specialized test set. This difference may indicate that the non-literal neural machine translation can be considered as an individual and important direction besides other sub-tasks of neural machine translation.