Machine Translation with Pre-specified Target-side Words Using a Semi-autoregressive Model

Seiichiro Kondo **Aomi Koyama** **Tomoshige Kiyuna**
Tosho Hirasawa **Mamoru Komachi**
Tokyo Metropolitan University
kondo-seiichiro@ed.tmu.ac.jp, koyama-aomi@ed.tmu.ac.jp
kiyuna-tomoshige@ed.tmu.ac.jp, hirasawa-tosho@ed.tmu.ac.jp
komachi@tmu.ac.jp

Abstract

We introduce our TMU Japanese-to-English system, which employs a semi-autoregressive model, to tackle the WAT 2021 (Nakazawa et al., 2021) restricted translation task. In this task, we translate an input sentence with the constraint that some words, called restricted target vocabularies (RTVs), must be contained in the output sentence. To satisfy this constraint, we use a semi-autoregressive model, namely, RecoverSAT (Ran et al., 2020), due to its ability (known as "forced translation") to insert specified words into the output sentence. When using "forced translation," the order of inserting RTVs is a critical problem. In our system, we obtain word alignment between a source sentence and the corresponding RTVs and then sort the RTVs in the order of their corresponding words or phrases in the source sentence. Using the model with sorted order RTVs, we succeeded in inserting all the RTVs into output sentences in more than 96% of the test sentences. Moreover, we confirmed that sorting RTVs improved the BLEU score compared with random order RTVs.

1 Introduction

In this study, we tackle a machine translation task called "restricted translation." This task requires the output sentence to contain all the pre-specified restricted target vocabularies (RTVs)[1]. In other words, we are given a source sentence and a set of RTVs, and we are supposed to generate an output sentence that contains all the RTVs in the set[2].

Since the emergence of neural machine translation (NMT) models (Sutskever et al., 2014; Bahdanau et al., 2015; Vaswani et al., 2017), several studies have been conducted to explore NMT systems capable of decoding translations under terminological constraints (Hasler et al., 2018; Dinu et al., 2019; Chen et al., 2020; Song et al., 2020). However, these previous studies were conducted under the condition that a bilingual dictionary is given. Moreover, these challenges are limited to autoregressive NMT systems, and scant research has been conducted on non-autoregressive or semi-autoregressive NMT systems, which have received more attention recently.

To accomplish restricted translation, where only target terminologies are given, we used a semi-autoregressive model called RecoverSAT (Ran et al., 2020), which generates a sentence as a sequence of segments. In this model, the segments are generated simultaneously, and each segment is predicted token-by-token. Ran et al. (2020) also attempted to force the model to generate a certain token at the beginning of a segment and showed that the model could generate valid sentences under the constraint. Then, we considered whether this model could be applied to generate sentences containing RTVs.

When tackling this task using this model, the insertion order of RTVs is a critical issue. To address this issue, we used GIZA++ (Och and Ney, 2003) to obtain word alignments and then identify the source position corresponding to the RTVs. Subsequently, we inserted them in the order in which their corresponding source tokens appear. We confirmed that sorting RTVs with GIZA++ improved the BLEU (Papineni et al., 2002) score. Finally, by using this model, we achieved all the RTVs outputs in more than 96% of the test sentences.

2 System Overview

2.1 Corpus Refinement

Morishita et al. (2019) reported that the synthetic

[1]Each RTV is either a word or a phrase.

[2]For details of the task description, see https://sites.google.com/view/restricted-translation-task/.

Proceedings of the 8th Workshop on Asian Translation, pages 68–73
Bangkok, Thailand (online), August 5-6, 2021.

data generated by back-translation (Sennrich et al., 2016) degraded the performance in the Japanese-to-English translation setting. The reason for this phenomenon was that the ASPEC (Nakazawa et al., 2016) training sentences are ordered by sentence alignment scores, and so the sentences with lower scores are considered relatively noisy data. Therefore, Morishita et al. (2019) attempted to generate synthetic data using forward-translation instead of standard back-translation and confirmed that forward-translation improved the performance of the Japanese-to-English translation setting.

Following Morishita et al. (2019), we used forward-translation to refine the latter half of the ASPEC training data. In the same manner as their method, we first trained a Japanese-to-English translation model on the first 1.5M sentences of the ASPEC training data. Subsequently, we used the trained model to translate the latter 1.5M Japanese sentences of the ASPEC training data and obtained refined English sentences. Finally, we combined the first 1.5M training data and the refined 1.5M training data and trained a Japanese-to-English translation model.

2.2 RecoverSAT

RecoverSAT (Ran et al., 2020) is a semi-autoregressive model that performs generation autoregressively in local and non-autoregressively in global. At each decoding step, the model generates a token in each segment, with paying attention to not only all the previous tokens in the segment but also those in all the other segments. The model continues decoding in each segment until either a special token, EOS or DEL, is generated, or the length of the generated token reaches the maximum token number. The final translation is a concatenation of all the segments except those that end with DEL.

RecoverSAT is also known for its capability to generate a translation under a word constraint (Ran et al., 2020), which is called the "forced translation" approach. In this approach, the model generates the constraint word (or phrases) at the beginning of an arbitrary segment. Once the constraint word (or phrase) has been generated, the model predicts the remainder of the segment in a semi-autoregressive manner.

In contrast to the original "forced translation," which only takes one constrained word (or phrase), we are required to place multiple RTVs in a translation. To compensate for this gap, we place the i-th RTV at the P_i-th segment as follows[3]:

$$P_i = \lfloor \frac{N_S}{N_V} \rfloor \cdot i \quad (1)$$

where N_S is the number of segments and N_V is the number of RTVs. When the RTVs have more phrases than segments during inference, we cut off phrases in the RTVs from the tail to fit the placeholder.

2.3 Sorting RTVs Using Source Alignment

RecoverSAT outputs RTVs in the order where they are inserted, so the order of inserting RTVs is important for accurate translation. We determined the order of the RTVs under the assumption that it correlated with the order of the aligned words in the input sentence.

We used GIZA++ to align each RTV with a word in the input sentence and sorted the RTVs in the order of their corresponding input words. When the RTV was a phrase, we first obtained a source word that was most aligned with each word in the RTV and then selected the source word with the highest alignment score as the aligned word for the entire RTV. If there was a tie, the first aligned word in the input sentence was selected as the corresponding word.

3 Experimental Setup

3.1 Dataset

We used the ASPEC (Nakazawa et al., 2016) dataset for Japanese-to-English translation. This dataset contains 3M sentences as training data, 1,790 sentences as validation data, and 1,812 sentences as test data. As explained in Section 2.1, we refined the latter half of the training data using forward-translation.

We used SentencePiece (Kudo and Richardson, 2018) to tokenize the training data for both the source and target sentences, where the vocabulary size was set to 4K. Note that we used SentencePiece models obtained from the first 1.5M training data through all the experiments. When determining the insertion order of RTVs using GIZA++, we used MeCab[4] with IPADIC to tokenize Japanese sentences before computing the alignment.

[3] Note that both P_i and i start from 0.

[4] https://taku910.github.io/mecab/

3.2 Evaluation

We evaluated system outputs using the following two distinct metrics.

BLEU score. The BLEU score is a metric evaluated by the n-gram matching rate with the reference. We calculated it using `multi-bleu.perl` in the Moses toolkit (Koehn et al., 2007).

Consistency score. The consistency score is the ratio of translations that satisfy the exact match of all the given constraints over the entire test corpus. The exact match is determined as follows. We simply lowercased hypotheses and constraints and then judged character-level sequence matching (including whitespaces) for each constraint.

For the final score, we calculated the BLEU score using only the translations that exactly matched their RTVs. In other words, first, we calculated the exact match, and then, we replaced the translations that did not satisfy the constraint with an empty string. Subsequently, we calculated the BLEU score with the modified translations.

3.3 Model

Transformer. We used "Transformer (base)" (Vaswani et al., 2017) for forward-translation and a baseline model. The hyperparameter settings were the same as described in Vaswani et al. (2017).

In the baseline model, we inserted the RTVs at the tail of the output sentence without sorting.

RecoverSAT. We use the encoder of the Transformer to initialize the encoder of RecoverSAT, and share the parameters of the embedding layers and the pre-softmax linear layer in the same way as Ran et al. (2020). We adopted the same model and hyperparameters that were used in the previous study (Ran et al., 2020)[5], where $d_{\text{model}} = 512$, $d_{\text{hidden}} = 512$, $n_{\text{layer}} = 6$, and $n_{\text{head}} = 8$. However, we did not share the source and target vocabularies.

Moreover, we changed the number of segments from the original paper (i.e., 10) because some examples had more than 10 (up to 14) RTVs in the test data. We also expanded the length of a segment to be able to insert all the tokens of the RTV if the RTV has more tokens than allowed by default. We examined four RecoverSAT models with different numbers of segments: 10 is the default value in Ran et al. (2020) and 14 is the maximum number of RTVs among the development data. The models with 21 and 29 segments have more free segments than the previous models, which are supposed to be lubricating segments to improve the overall output.

[5]We used the implementation at https://github.com/ranqiu92/RecoverSAT and minimally modified it for inserting RTVs.

	BLEU	RIBES	AMFM
RecoverSAT	25.29	0.653597	0.612290

Table 1: Results of the official score using RecoverSAT with 14 segments and forced translation with sorted order.

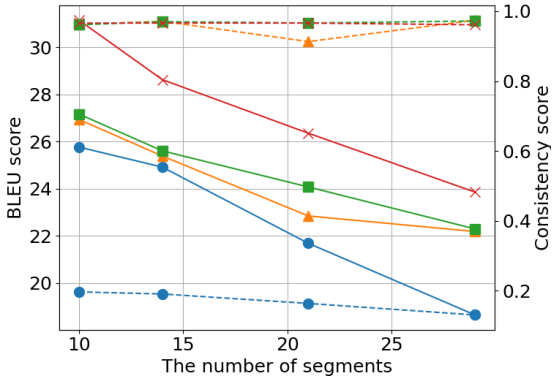

Figure 1: Results of our experiments using RecoverSAT. The solid line represents the BLEU score, and the dotted line represents the consistency score. The dot marker represents RecoverSAT without RTVs. The triangle marker represents forced translation without sorting RTVs. The square marker represents forced translation with sorted order. The cross marker represents forced translation with oracle order.

4 Results

4.1 Official Evaluation

Table 1 presents the official BLEU, RIBES (Isozaki et al., 2010), and AMFM (Banchs et al., 2015) scores, calculated in the evaluation server, for the model in which the number of segments is 14. As shown in Table 1, the BLEU, RIBES, and AMFM scores were 25.29, 0.653597, and 0.612290 points, respectively.

4.2 Our Evaluation

Table 2 presents the scores obtained in our evaluation. Moreover, Figure 1 shows the BLEU score and consistency scores for different numbers of segments {10, 14, 21, 29}.

BLEU score. Figure 1 shows that the translation accuracy decreases as the number of segments in-

Model	BLEU score	Consistency score	Final score
Transformer	27.78	0.220	0.27
+ Append RTVs	25.57	**1.000**	26.75
RecoverSAT	25.76	0.197	0.16
+ Forced translation with random order	26.93	0.962	26.98
+ Forced translation with sorted order	27.16	0.961	27.10
+ Forced translation with oracle order	**31.14**	0.966	**31.02**

Table 2: Results of the experiments in our evaluation. The number of segments of RecoverSAT is 10. The consistency score is the ratio of sentences satisfying the exact match of the given constraints. The final score is the constraint-aware BLEU score. "random order": we insert RTVs without sorting. "sorted order": we insert RTVs in the order of the corresponding source words. "oracle order": we insert RTVs in the same order as that in the reference.

creases, similar to the previous study (Ran et al., 2020). This may be because the model predicts the target tokens more independently as the number of segments increases. As the number of segments increases, the length of each segment becomes shorter, and the model becomes closer to the non-autoregressive model.

Table 2 shows that sorting the RTVs using GIZA++ improves the BLEU score. However, there is still a significant gap in the scores compared with those obtained using the oracle order. This is because the word order between Japanese and English is different.

Consistency score. Figure 1 shows that RecoverSAT with forced translation reliably outputs RTVs in almost all the cases. When the number of segments was 10, we could not insert all the RTVs in some test sentences with more than 10 RTVs[6]. On the other hand, when the number of segments was 14 or more, it was expected that all the RTVs could be inserted into all the test sentences. However, some output sentences did not contain all the RTVs, even if the number of segments was 14 or more. This result indicates that the model generates a special token, DEL, to delete segments beginning with the RTVs.

The final BLEU score of the model with 10 segments, which gives up to generate some RTVs on occasion, was the highest. This is because it is rare to have more than 10 RTVs for a single sentence[7]. Additionally, we confirmed that the insertion of RTVs was effective in improving not only the consistency score but also the BLEU score.

5 Related Work

Previously, some NMT with terminology constraints have been studied (Hasler et al., 2018; Alkhouli et al., 2018; Dinu et al., 2019; Chen et al., 2020; Song et al., 2020). For example, Song et al. (2020) proposed a dedicated head in a multi-head Transformer architecture to learn explicit word alignment and use it to guide the constrained decoding process. When the source-aligned word matches a dictionary, the model outputs the corresponding target word. However, these models are not available for the "restricted translation" task because we can only access the target-side vocabularies.

In this study, we used the semi-autoregressive model RecoverSAT (Ran et al., 2020). Originally, this model was not intended to output forcibly more than one constrained word. A non-autoregressive model can decode target tokens simultaneously, resulting in faster decoding. However, its output sentence suffers from the multi-modality problem causing token repetitions or missing by not using the dependency between the output words (Gu et al., 2018; Ran et al., 2020). Thus, Ran et al. (2020) proposed RecoverSAT to alleviate this problem. Their model could maintain the accuracy of the autoregressive model while achieving a faster processing speed. They also mentioned that, as the number of segments increases, the closer the model becomes to a non-autoregressive model. In other words, when the number of segments increases, the decoding process is faster, but the accuracy is lower. Moreover, they attempted to force the model to generate a pre-specified token at the beginning of

[6] As mentioned in Section 3.3, the maximum number of RTVs in the test set was 14.

[7] Only 14 out of 1,812 (0.8%) sentences were given more than 10 RTVs in the test data.

a segment and showed that the model could avoid repetitive output and translate properly.

6 Conclusions

We introduced a semi-autoregressive approach to tackle the restricted translation task. In our experiments, we showed that RecoverSAT could output almost all the RTVs. Additionally, we used source sentence alignment to determine the insertion position and observed that it improved the BLEU score. Moreover, the importance of the order of the RTVs was confirmed by the fact that the score was considerably improved by inserting RTVs in the order in which they appear in the reference translations. However, there is still room for improvement in determining the insertion order. In future work, investigating how to determine the best order to insert RTVs will be necessary.

References

Tamer Alkhouli, Gabriel Bretschner, and Hermann Ney. 2018. On the alignment problem in multi-head attention-based neural machine translation. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 177–185, Brussels, Belgium. Association for Computational Linguistics.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *Proceedings of the 3rd International Conference on Learning Representations*, San Diego, California.

Rafael E. Banchs, Luis F. D'Haro, and Haizhou Li. 2015. Adequacy–fluency metrics: Evaluating MT in the continuous space model framework. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 23(3):472–482.

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*, pages 3587–3593. International Joint Conferences on Artificial Intelligence Organization.

Georgiana Dinu, Prashant Mathur, Marcello Federico, and Yaser Al-Onaizan. 2019. Training neural machine translation to apply terminology constraints. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3063–3068, Florence, Italy. Association for Computational Linguistics.

Jiatao Gu, James Bradbury, Caiming Xiong, Victor O.K. Li, and Richard Socher. 2018. Non-autoregressive neural machine translation. In *Proceedings of the 6th International Conference on Learning Representations*, Vancouver, Canada. OpenReview.net.

Eva Hasler, Adrià de Gispert, Gonzalo Iglesias, and Bill Byrne. 2018. Neural machine translation decoding with terminology constraints. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 506–512, New Orleans, Louisiana. Association for Computational Linguistics.

Hideki Isozaki, Tsutomu Hirao, Kevin Duh, Katsuhito Sudoh, and Hajime Tsukada. 2010. Automatic evaluation of translation quality for distant language pairs. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 944–952, Cambridge, MA. Association for Computational Linguistics.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 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, Prague, Czech Republic. Association for Computational Linguistics.

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

Makoto Morishita, Jun Suzuki, and Masaaki Nagata. 2019. NTT neural machine translation systems at WAT 2019. In *Proceedings of the 6th Workshop on Asian Translation*, pages 99–105, Hong Kong, China. Association for Computational Linguistics.

Toshiaki Nakazawa, Hideki Nakayama, Chenchen Ding, Raj Dabre, Shohei Higashiyama, Hideya Mino, Isao Goto, Win Pa Pa, Anoop Kunchukuttan, Shantipriya Parida, Ondřej Bojar, Chenhui Chu, Akiko Eriguchi, Kaori Abe, and Sadao Oda, Yusuke Kurohashi. 2021. Overview of the 8th workshop on Asian translation. In *Proceedings of the 8th Workshop on Asian Translation*, Bangkok, Thailand. Association for Computational Linguistics.

Toshiaki Nakazawa, Manabu Yaguchi, Kiyotaka Uchimoto, Masao Utiyama, Eiichiro Sumita, Sadao Kurohashi, and Hitoshi Isahara. 2016. ASPEC: Asian scientific paper excerpt corpus. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 2204–2208, Portorož, Slovenia. European Language Resources Association (ELRA).

Franz Josef Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Qiu Ran, Yankai Lin, Peng Li, and Jie Zhou. 2020. Learning to recover from multi-modality errors for non-autoregressive neural machine translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3059–3069, Online. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96, Berlin, Germany. Association for Computational Linguistics.

Kai Song, Kun Wang, Heng Yu, Yue Zhang, Zhongqiang Huang, Weihua Luo, Xiangyu Duan, and Min Zhang. 2020. Alignment-enhanced transformer for constraining NMT with pre-specified translations. In *Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence*, pages 8886–8893, New York City, New York. Association for the Advancement of Artificial Intelligence.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems 27*, pages 3104–3112, Montreal, Canada. Curran Associates, Inc.

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 30*, pages 5998–6008, Long Beach, California. Curran Associates, Inc.